Content Delivery Networks Architecture, Features, and Benefits

Technical Report · April 2016

DOI: 10.13140/RG.2.1.1762.0722

1 author:

Mustafa Sadiq Aljumaily
University of Tennessee

Content Delivery Networks Architecture, Features, and Benefits

Mustafa Aljumaily,

mlatief@vols.utk.edu

PhD student in Computer Engineering

University of Tennessee, Knoxville

Department of Electrical Engineering and Computer Science

ECE-553: Computer Networks

Abstract

Content Delivery Networks (CDN) are one of the most important components of the Internet today. They were proposed since the late 90's to maximize bandwidth, improve accessibility, and maintain correctness through content replication and bringing the content as close as possible to the clients [1]. The commercial success of the Internet and e-services, together with the exploding use of complex media content online has paved the way for the birth and growing interest in these networks. With CDNs, web content is distributed to cache servers located close to users, resulting in fast, reliable applications and Web services for the users. CDNs maintain multiple Points of Presence (PoP) with clusters of (the so-called surrogate) servers that store copies of identical content, such that users' requests are satisfied by the most appropriate site. There are two general approaches to building CDNs. One is overlay model, which replicate content to thousands of servers worldwide. Another approach is network model which deploys code to routers and switches so that they can recognize specific application types and make forward decisions on the basis of predefined policies [2]. Conventional CDNs can be mainly classified into two sub-categories, Commercial CDNs, and Academic CDNs [3].

General Terms: Taxonomy, Survey, CDNs, Design, Performance, Architecture, Content networks, Content distribution, peer-to-peer, replica management, request-routing

1 Introduction

The application of content delivery networking (CDN) in today's internet is one of the hot topics in computer networking research and development efforts and one of the biggest IP trends going on right now. CDNs leverage high-layer network intelligence to efficiently manage the delivery of data, which is becoming increasingly multimedia in nature. They were initially built on top of the public Internet to accelerate Web site performance. With the success of CDNs for this purpose, network engineers realized that the intelligent network tools could be applied in other beneficial and profitable ways [4]. A key challenge for Internet infrastructure -that is getting bigger and bigger everyday- has been delivering increasingly complex data to a greedy and growing user population. The need to scale has led to the development of thousand-node clusters, global-scale CDNs, and more recently, self-managing peer-to-peer (P2P) structures. These content delivery mechanisms are rapidly changing the nature of Internet content delivery and traffic; therefore, an understanding of the modern Internet requires a detailed understanding of these new mechanisms and the data they serve [5]. CDNs were born to distribute heavily requested contents from popular web servers, most of all image files. Nowadays, a CDN supports the delivery of any type of dynamic content, including various forms of interactive media streaming. CDN providers are companies devoted to hosting in their servers the content of third-party content providers, to mirroring or replicating such contents on several servers spread over the world, and to transparently redirecting the customers' requests to the 'best replica' (e.g. the closest replica, or the one from which the customer would access content at the lowest latency). Designing a complete solution for CDN therefore requires addressing a number of technical issues: which kind of content should be hosted (if any) at a given CDN server (replica placement), how the content must be kept updated, which is the 'best replica' for a given customer, which mechanisms must be in place to transparently redirect the user to such replica [3]. A proper placement of replica servers shortens the path from servers to clients thus lowering the risk of encountering bottlenecks in the non-dedicated environment of the Internet. Successfully implemented CDN can accelerate end user access to content, reduce network traffic, and reduce content provider hardware requirements.

As known, the Internet is a network of heterogeneous networks composed of thousands of different autonomous systems (AS) ranging from large backbone providers to small local ISPs. The autonomous systems connect to each other creating the global Internet. The communication between two networks is achieved through the connection of border routers in a peering session. Two peer routers periodically exchange routing information and forward the received packets to carry each packet to its correct destination. This structure of the Internet as an interconnection of individual networks is the key to its scalability but is not sufficient to guarantee that a quickly growing number of users, services and traffic do not create bottlenecks that, if left unaddressed, can slow down performance. Bottlenecks may occur at many points in the core Internet and most of all in correspondence to peering points and backbones [6]. Such problems are the key motivation to adopt the CDNs as a good solution to the possible bottlenecks of the networks and ensure fast response to uses requests.

The network capacity is determined by the capacity of its cables and routers and although cable capacity is not an issue, the strongest limit to the backbone capacity comes from the packet-forwarding hardware and software of the routers. Once a peering point has been installed, traffic may have grown beyond expectations, resulting in a saturated link, typically because a network provider purchases just enough capacity to handle current traffic levels, to maximize the link utilization. The practice of running links at full capacity is one of the major causes of traffic bottlenecks showing very high utilization but also high rates of packet loss and high latency. Further the capacity of long backbones cannot always be adapted to the sudden and fast increases of the Internet traffic [6].

All of the above mentioned problems are considered good motivations for the internet service providers to participate and support the CDN services as it will lead to better performance and increase the users satisfiability which will lead to reduce the number of users abandoning a specific ISP because of bad service or frustration with a long delays waiting for their data to be delivered to their personal machines.

The rest of this paper is organized as following: section two will be about the different aspects of CDN architecture, whereas section three will be about the taxonomy of CDNs. Section four is focusing on the benefits and features of today CDN. Then the paper will be concluded by section five.

2 Content Delivery Networks (CDN) Architecture

Before going in details about how to build the CDNs', we should give a brief overview about how they are distributed world-wide and how they are managed and utilized:

A typical content delivery environment has the replicated Web server clusters that are located at the edge of the network to which the end-users are connected. A content provider (i.e. customer) can sign up with a CDN provider for service and have its content placed on the content servers. The content is replicated either on-demand when users request for it, or it can be replicated in advanced, by pushing the content to the surrogate servers. A user is served with the content from the nearby replicated Web server. Thus, the user ends up unknowingly communicating with a replicated CDN server close to it and retrieves files from that server [7]. As explained in figure (1) below:

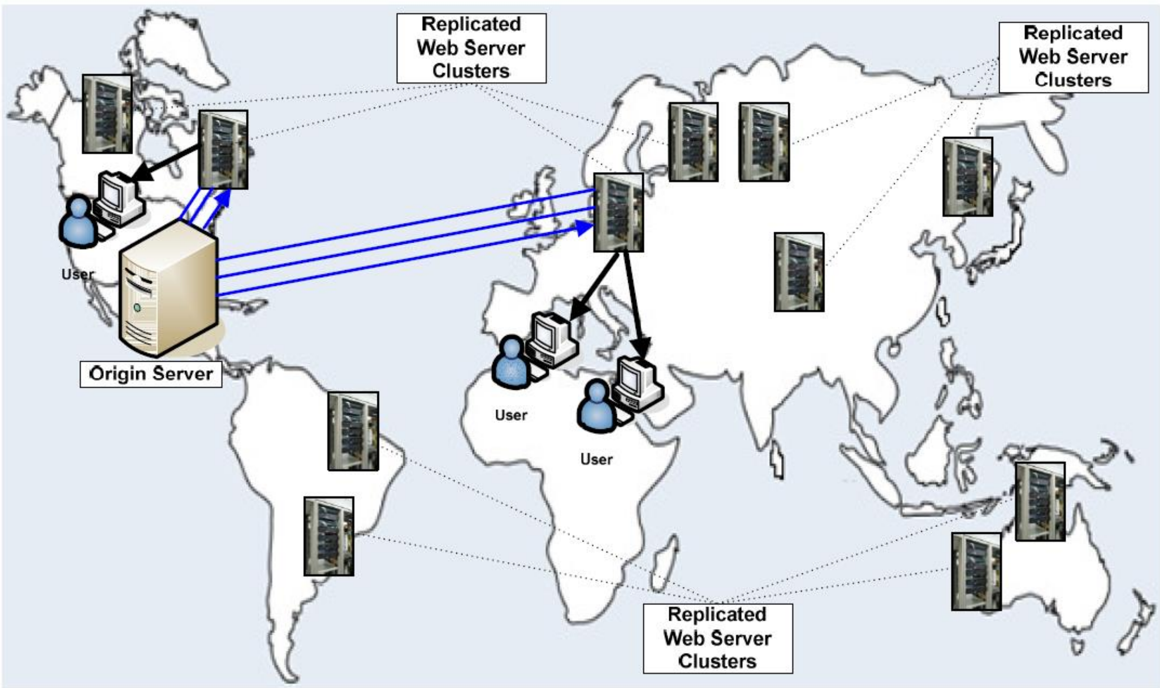

Figure (1): Abstract architecture of a Content Delivery Network (CDN)

The architecture of CDNs can be presented according to a **layered approach**. The layered architecture of CDNs consists of the following layers: Basic Fabric, Communication & Connectivity, CDN and End-user as shown in figure (2). The layers are defined in the following as a **bottom up approach** [7]:

- **Basic Fabric** is the lowest layer of a CDN. It provides the infrastructural resources for its formation. This layer consists of the distributed computational resources such as Symmetric Multiprocessing (SMP) servers, clusters, file servers, index servers, and basic network infrastructure connected by high-bandwidth network. Each of these resources runs system software such as operating system, distributed file management system, and content indexing and management systems.
- **Communication & Connectivity layer** provides the core internet protocols (e.g. TCP/UDP, FTP) as well as CDN specific internet protocols (e.g. Internet Cache Protocol (ICP), Hypertext Caching Protocol (HTCP), and Cache Array Routing Protocols (CARP), and authentication protocols such as Public Key Infrastructures (PKI), or Secure Sockets Layer (SSL) for communication, caching and delivery of content and/or services in an authenticated

manner. Application specific overlay structures provide efficient search and retrieval capabilities for replicated content by maintaining distributed indexes.

- **CDN layer** consists of the core functionalities of CDN. It can be divided into three sub-layers: CDN services, CDN types and content types. A CDN provides core services such as surrogate selection, request-routing, caching and geographic load balancing, and user specific services for Service Level Agreement (SLA) management, resource sharing and CDN brokering. A CDN can operate within an enterprise domain, it can be for academic and/or public purpose or it can simply be used as edge servers of content and services. A CDN can also be dedicated to file sharing based on a peer-to-peer (P2P) architecture. A CDN provides all types of content (e.g. text, audio, video … etc.) to its users.
- **End-users** are at the top of the CDN layered architecture. In this layer, we have the Web users who connect to the CDN by specifying the URL of content provider's Web site, in their Web browsers.

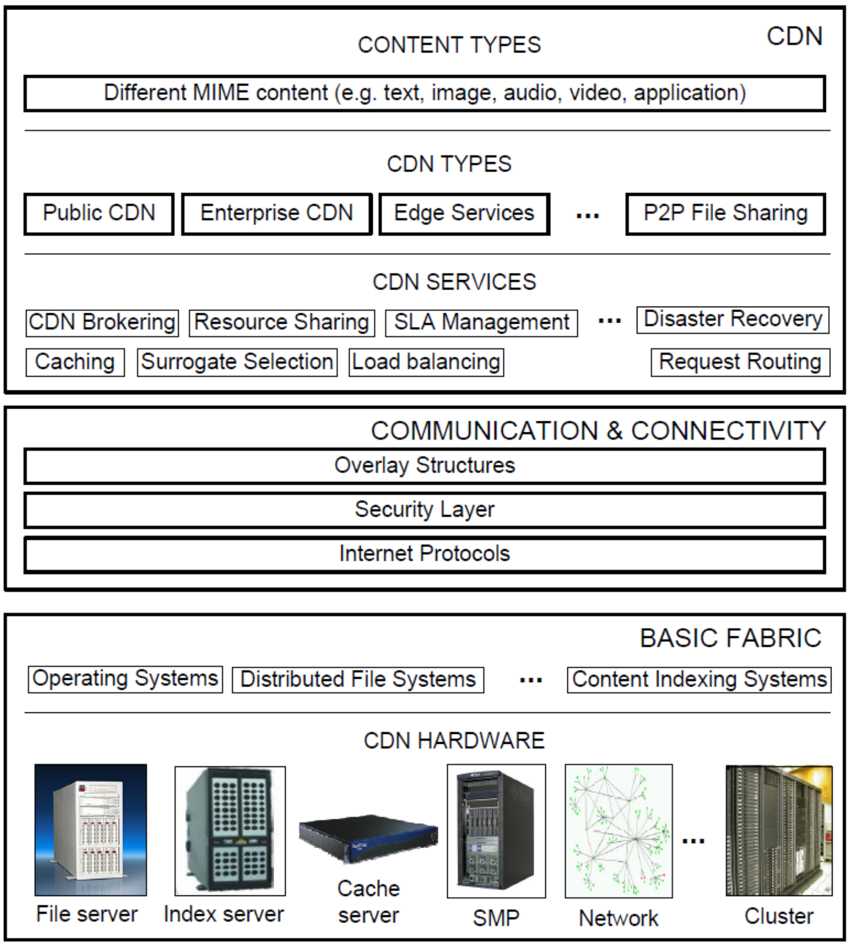

Figure (2): Layered Approach in building the CDNs'

3 Content Delivery Networks (CDN) Taxonomy

The main goal of server replication in a CDN is to avoid large amounts of data repeatedly traversing possibly congested links on the Internet. There are a variety of ways and scale (local area or wide area networks) in which content networks may be implemented. Local solutions are web clusters that typically hosts single site, and web farms, typically used to host multiple sites [6]. Wide area solutions include: distributed web server systems, used to host single or multiple sites; cooperative proxy cache networks (a service infrastructure to reduce latency in downloading web objects) and content delivery networks [8] as shown in the figure (3) below:

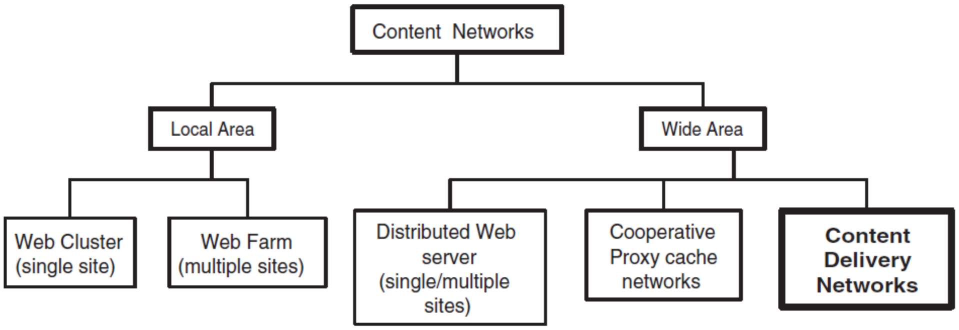

Figure (3): the taxonomy of content networks.

And because we are focusing on the CDNs in this paper, we will talk only about it in the next sections. Talking about the CDN taxonomy means basically explaining the different aspects and issues that participate in defining different CDN network types for different usages and applications and these issues are shown in the figure (4) below:

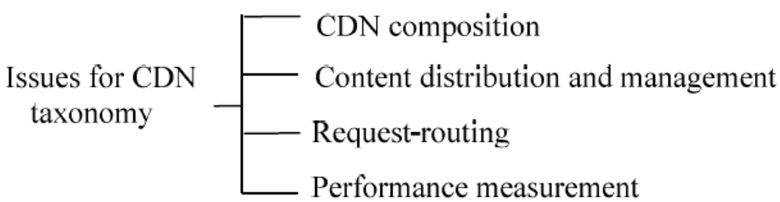

Figure (4): CDN issues taxonomy.

3-1 CDN Composition

A CDN typically incorporates dynamic information about network conditions and load on the cache servers, to redirect request and balance loads among surrogates [7]. The structure of a CDN varies depending on the content/services it provides to its users. Within the structure of a CDN, a set of surrogates is used to build the content-delivery infrastructure, some combinations of relationships and mechanisms are used for redirecting client requests to a surrogate and interaction protocols are used for communications among the CDN elements.

Figure (5) shows the taxonomy based on the various structural characteristics of CDNs. These characteristics are central to the composition of a CDN and they address the organization, types of servers used, relationships and interactions among CDN components, as well as the different content and services provided by the CDNs [7]. Each of these services has its benefits, features and limitations and there are many research efforts on developing all of them and propose better solutions [14].

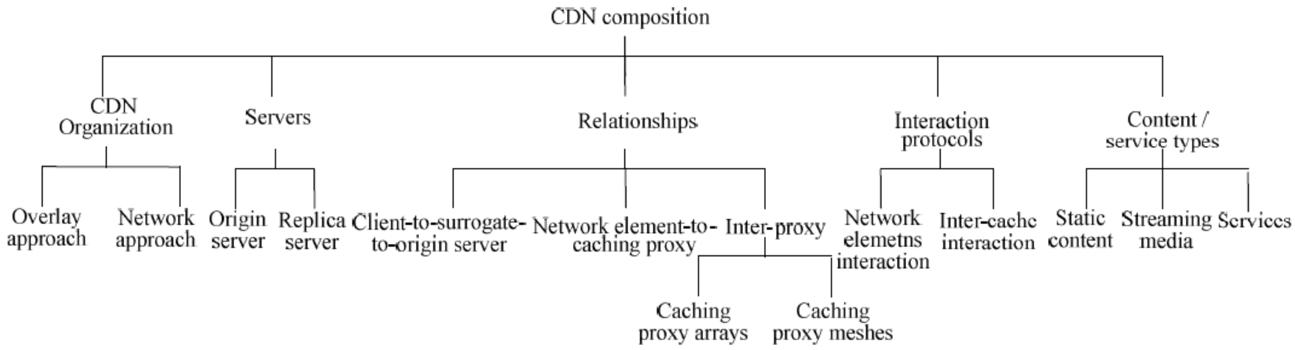

Figure (5): the taxonomy of the issues related to the composition of the CDN [7].

3-2 Content Distribution and Management

Content distribution and management is strategically vital in a CDN for efficient content delivery and for overall performance. Content distribution includes – the placement of surrogates to some strategic positions so that the edge servers are close to the clients; content selection and delivery based on the type and frequency of specific user requests and content outsourcing to decide which outsourcing methodology to follow. Content management is largely dependent on the techniques for cache organization (i.e. caching techniques, cache maintenance and cache update). The content distribution and management taxonomy is shown in Figure 6 below:

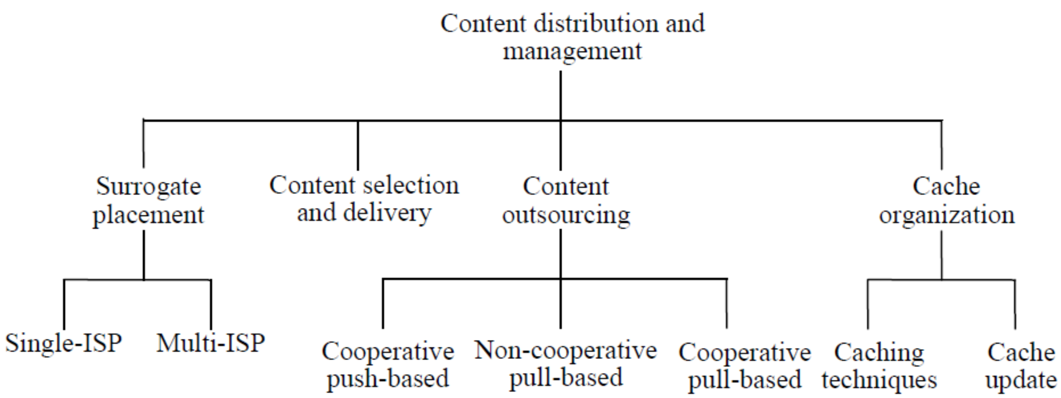

Figure (6): the issues related to the content distribution and management [7].

3-3 Request-Routing

A request-routing system is responsible for routing client requests to an appropriate surrogate server for the delivery of content. It consists of a collection of network elements to support request-routing for a single CDN. It directs client requests to the replica server that is the 'closest' to the client. However, the closest server may not be the best surrogate server for servicing the client request according to [9]. Hence, a request-routing system uses a set of metrics such as network proximity, client perceived latency, distance, and replica server load in an attempt to direct users to the closest surrogate that can best serve the request. The request-routing system in a CDN has two parts: deployment of a request-routing algorithm and use of a request-routing mechanism [10].

Figure (7) below provides a high-level view of the request-routing in a CDN environment. The interaction flows are following these steps:

1- The client requests content from the content provider by specifying its URL in the Web browser. Client's request is directed to its origin server.

2- When origin server receives a request, it makes a decision to provide only the basic content (e.g. index page of the Web site) that can be served from its origin server.

3- To serve the high bandwidth demanding and frequently asked content (e.g. embedded objects – fresh content, navigation bar, banner ads etc.), content provider's origin server redirects client's request to the CDN provider.

4- Using the proprietary selection algorithm, the CDN provider selects the replica server which is 'closest' to the client, in order to serve the requested embedded objects.

5- Selected replica server gets the embedded objects from the origin server, serves the client requests and caches it for subsequent request servicing.

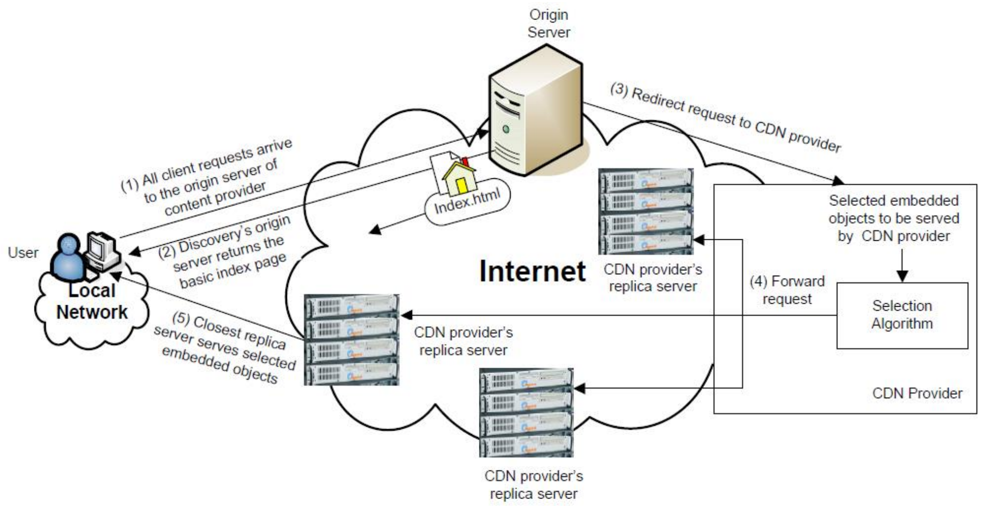

Figure (7): Request-Routing process in the CDN [7].

3-4 Performance Measurement

Performance measurement of a CDN is done to measure its ability to serve the customers with the desired content and/or service. Customers utilizing the service need feedback on the performance of the CDN as a whole. Performance measurement offers the ability to predict, monitor and ensure the end-to-end performance of a CDN. The measurement is achieved with a combination of hardware and software-based probes distributed around the CDN, as well as using the logs from various servers within the CDN. Typically five key metrics are used by the content providers to evaluate the performance of a CDN [11]. Those are:

- Cache hit ratio: It is defined as the ratio of the number of cached documents versus total documents requested. A high hit rate reflects that a CDN is using an effective cache policy to manage its caches.
- Reserved bandwidth: It is the measure of the bandwidth used by the origin server. It is measured in bytes and is retrieved from the origin server.
- Latency: It refers to the user perceived response time. Reduced latency signifies the decreases in bandwidth reserved by the origin server.
- Surrogate server utilization: It refers to the fraction of time during which the surrogate servers remain busy. This metric is used by the administrators to calculate CPU load, number of requests served and storage I/O usage.
- Reliability: Packet-loss measurements are used to determine the reliability of a CDN. High reliability indicates that a CDN incurs less packet loss and is always available to the clients.

4 Content Delivery Networks (CDN) Benefits

After everything mentioned before, it seems fair to say that CDNs' are considered one of the most important parts In the Internet today. Its features and applications are everywhere from providing the contents to the cell phones laptops, streaming media, and MP3 players to the different types of servers as the figure (8) below shows:

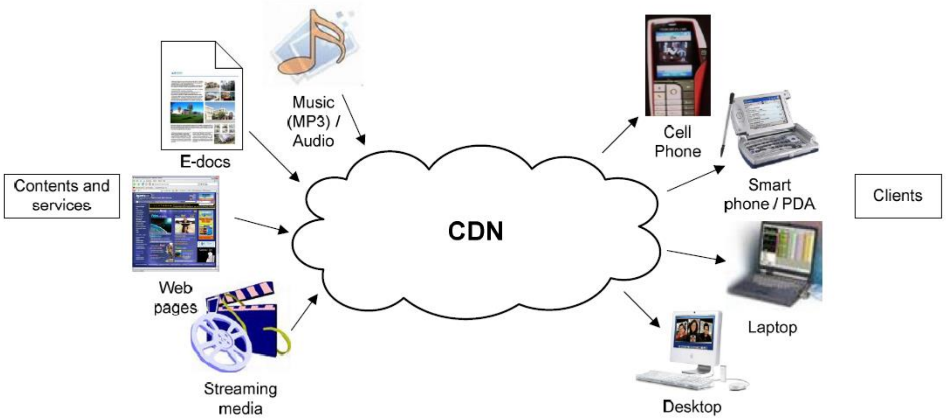

Figure (8): contents and services provided by CDNs

The **Benefits of Using a CDN** can be summarized as following:

1. Different domains: Browsers limit the number of concurrent connections (file downloads) to a single domain. Most permit four active connections so the fifth download is blocked until one of the previous files has been fully retrieved. You can often see this limit in action when downloading many large files from the same site. CDN files are hosted on a different domain. In effect, a single CDN permits the browser to download a further four files at the same time.

2. Files may be pre-cached: jQuery is ubiquitous on the web. There's a high probability that someone visiting your pages has already visited a site using the Google CDN. Therefore, the file has already been cached by your browser and won't need to be downloaded again.

3. High-capacity infrastructures: someone may have great hosting but it probably doesn't have the capacity or scalability offered by Google, Microsoft or Yahoo. The better CDNs offer higher availability, lower network latency and lower packet loss.

4. Distributed data centers: If the main web server is based in Dallas, users from Europe or Asia must make a number of trans-continental electronic hops when they access that website files. Many CDNs provide localized data centers which are closer to the user and result in faster downloads.

5. Built-in version control: It's usually possible to link to a specific version of a CSS file or JavaScript library. And we can often request the "latest" version if required.

6. Usage analytics: Many commercial CDNs provide file usage reports since they generally charge per byte. Those reports can supplement our own website analytics and, in some cases, may offer a better impression of video views and downloads.

7. Boosts performance and saves money: A CDN can distribute the load, save bandwidth, boost performance and reduce your existing hosting costs — often for free.

Also, the analysis of today's CDNs results in that CDNs focus on the following business goals [12]:

Scalability – The main business goal of a CDN is to achieve scalability. Scalability refers to the ability of the system to expand in order to handle new and large amounts of data, users and transactions without any significant decline in performance. To expand in a global scale, CDNs need to invest time and costs in provisioning additional network connections and infrastructures.

Security – One of the major concerns of a CDN is to provide potential security solutions for confidential and high-value content [13]. Security is the protection of content against unauthorized access and modification. Without proper security control, a CDN platform is subject to cyber fraud, distributed denial-of-service (DDoS) attacks, viruses, and other unwanted intrusions that can cripple business [12]. A CDN aims at meeting the stringent requirements of physical, network, software, data and procedural security. Once the security requirements are met, a CDN can eliminate the need for costly hardware and dedicated component to protect content and transactions.

Reliability, Responsiveness and Performance – Reliability refers to when a service is available and what are the bounds on service outages that may be expected. A CDN provider can improve client access to specialized content through delivering it from multiple locations. Responsiveness implies, while in the face of possible outages, how long it would take a service to start performing the normal course of operation again. Performance of a CDN is typically characterized by the response time (i.e. latency) perceived by the end-users. Slow response time is the single greatest contributor to customers' abandoning Web sites and processes [12].

5 Conclusion

From all what was mentioned before, it is obvious that the Internet as we know it today cannot survive without the content delivery networks. More and more demands of users for better (in quality and quantity) data traffic means that the Internet Engineering Task Force (IETF) needs to adapt the CDN as a basic part of it and encourage more research on its improvement and compatibility with the other parts of the Internet. Benefits of CDNs are everywhere in the Internet and with the increasing demands for better service, many companies like Akamai, Amazon, CDN77, and the other main vendors within the CDN industry today need to increase their efforts to make it better and more secure.

References

[1] Vakali, A. and Pallis, G. Content delivery networks: Status and trends. IEEE Internet Computing 7, 6 (2003), 68–74.

[2] Content Delivery Networks (CDNs) – A Reference Guide, Matthew Liste from Cisco.

[3] M. Pathan. Content distribution networks (cdn) - research directory, May 2007. URL http://www.cs.mu.oz.au/~apathan/CDNs.html.

[4] Jaison Paul Mulerikkal. An Architecture for Distributed Content Delivery Network. MSC thesis submitted to the Science, Engineering, and Technology Portfolio, in the Royal Melbourne Institute of Technology. Melbourne, Victoria, Australia. 2007.

[5] Stefan S, Krishna P. G., Richard J. D., Steven D. G., and Henry M. L. An Analysis of Internet Content Delivery Systems. University of Washington. 2004.

[6] N. Bartolini, E. Casalicchio, and S. Tucci, “A Walk Through Content Delivery Networks,” In Proceedings of MASCOTS 2003, LNCS Vol. 2965/2004, pp. 1-25, April 2004.

[7] Pathan, Mukaddim, and Rajkumar Buyya. "A Taxonomy of CDNs." Content Delivery Networks Lecture Notes Electrical Engineering: 33-77.

[8] D. C. Verma. Content Distribution Networks: An engineering approach. Wiley Inter-Science, 2002.

[9] C. M. Chen, Y. Ling, M. Pang, W. Chen, S. Cai, Y. Suwa, O. Altintas, “Scalable Request-Routing with Next- Neighbor Load Sharing in Multi-Server Environments,” In Proceedings of the 19th International Conference on Advanced Information Networking and Applications, IEEE Computer Society, Washington, DC, USA, pp. 441-446, 2005.

[10] S. Sivasubramanian, M. Szymaniak, G. Pierre, and M. Van Steen, “Replication of Web Hosting Systems,” ACM Computing Surveys, Vol. 36, No. 3, ACM Press, NY, USA, 2004.

[11] F. Douglis, and M. F. Kaashoek, “Scalable Internet Services,” IEEE Internet Computing, Vol. 5, No. 4, 2001, pp.36-37.

[12] Akamai Technologies Inc., “Akamai-The Business Internet - A Predictable Platform for Profitable E-Business,” 2004.

[13] R. Brussee, H. Eertink, W. Huijsen, B. Hulsebosch, M. Rougoor, W. Teeuw, M. Wibbels, and H. Zandbelt, “Content Distribution Network State of the Art,” Telematica Instituut, June 2001.

[14] Buyya, Rajkumar, Al-Mukaddim Pathan, James Broberg, and Zahir Tari. "A Case for Peering of Content Delivery Networks." IEEE Distributed Systems Online IEEE Distrib. Syst. Online 7.10 (2006).

[15] Website: http://www.sitepoint.com/7-reasons-to-use-a-cdn/.