

# Overview of Content-Delivery Networks Architecture

Afonso Silva, Alfredo Gomes, and Axel Ferreira

University of Minho, Department of Informatics, 4710-057 Braga, Portugal  
e-mail: {a70387,a71655,a53064}@alunos.uminho.pt

**Abstract.** Given the diversity of existing Content Delivery Networks (CDN) architectures and the lack of a reference explicitly organising this information, this paper intends to overview three main CDN architectures. A description of Traditional Commercial *CDNs*, Peer-to-Peer (P2P) *CDNs* and Hybrid *CDNs* is presented. For each architecture, the advantages, disadvantages, technical challenges and application scopes are presented. Implications for further research and practice are discussed.

## 1 Introduction

THE Internet has its origins in the early 1980's and presented an exponential growth when commercial companies started to link to the existing academic and military networks during the 90's. As the popularity of the Internet increased the number of devices connected started to see an exponential growth.

At that time networks were unreliable and Internet communication protocols were designed in a robust fashion. As an example, Hypertext Transfer Protocol (HTTP) was designed to survive multiple packet losses and thus being a very chatty protocol, there are multiple Round-trip Time (RTTs) this causes a latency problem over long distance communication. Modern networks are faster and more reliable, but most of the core protocols mentioned above don't take advantage of the increased reliability. As the Internet role in daily life grew, new technologies start to emerge providing images videos and other dynamic content. This caused a bottleneck in servers with popular content. The above problems, led to the development Content Delivery Networks.

A *CDN* is a large network of distributed storage servers which cache the content in multiple locations strategically spread. Traditionally *CDN* servers are stored in ISP's facilities. This reduces both infrastructural costs to *CDN* Providers and ISP's costs by keeping popular content local, in contrast to bringing it from another ISP's network.

A *CDN* works by keeping a copy of the content in different locations. Whenever content is requested, management software calculates the best Server for accessing the content. This reduces distance to server which improves latency and minimizes packet loss. Since the calculations are dynamic, in case of a server malfunction another Server is chosen, providing redundancy.

Traditional *CDN* business model target bandwidth as a cost factor. This brings two advantages over a 'In House' solution: First, there are no initial infrastructural costs, this is a big advantage. An example of the dimension of such structure's size would be Akamai's infrastructure. It is composed by more than 61,000 servers, spread across 1000 networks in 70 countries worldwide[2]; Second, provides bandwidth on demand. In comparison with an 'In House' solution, there is no need to deal with both predictable and unpredictable traffic peaks. Due to its large distributed infrastructure, and high bandwidth on demand *CDNs* are especially well suited to absorb a Denial of Service (DoS) attacks[?].

It is worth noting that *CDNs* bring cost per performance down. Despite that, *CDNs* are expensive, and currently are used mostly by big companies. It is imperative that *CDNs* get cheaper. This will allow more parties to use them, thus enhancing Internet experience to the final consumer and reducing total Internet traffic at the same time. An example of a service using a *CDN* (Amazon EC2) would be the popular movie streaming service Netflix.

## 1.1 Purpose

This paper's purpose is to provide organised information about different *CDN* architectures: the traditional large server infrastructure *CDN*, the P2P *CDN* and a new trend of Hybrid architecture which relies on both Servers and Peers.

## 2 Academic *CDN* P2P

As said before *CDN Commercial Architecture* brings mostly advantages to clients but it comes with high financial cost. This happens due to the large distributed infrastructure that causes huge financial costs. Therefore only companies with large financial resources can afford it, limiting the use of *CDN* services to bigger companies. As a result, individual content creators, as well as smaller and medium sized companies cannot afford the services.

In order to improve this big disadvantage another architecture was implemented: the *Academic Architecture*.

### 2.1 Application Scope

In contrast to commercial architectures, *academic* ones are free, both in terms of profit and advertisement. *P2P* architectures, where each user shares a some of their machine's resources (Storage, CPU cycles, ...) with the remaining peers. In other words, P2P architecture creates a distributed storage medium that facilitates the search, the publishing and recovery of files by members of the network.

**Main Advantages** This alternative makes the network independent from entrepreneurial servers, which makes it significantly cheaper. The costs associated with this architecture are the small resources one has to share in order to use the *CDN P2P*. Another advantage of this architecture would be the scalability, both quality and speed of the network increase as the number of peers increases. i.e.: For instance, high quality streaming to a great number of users may benefit from using an *academic architecture*, as the provider of that same stream won't have to deal with the cost of renting a server and, beyond that, the high number of users will guarantee a wide array of available resources to a global video transmission. Similar with what happens with popular P2P files. The more popular it is, the more peers it has, and consequently the faster it is possible to download due to having more resources to download from.

### 2.2 Associated Challenges

Despite being more cost-effective than commercial *CDNs*, it still has some disadvantages. The biggest problem is the availability and quality of content reception of unpopular content, since the number of peers is smaller. Another big problem has to do with the fact that this *CDN* is dependent on volunteers and therefore don't follow a clear set of rules causing a lack of standard specification of *CDN* interfaces. Consequently, this makes it hard to maintenance and support end users.

## 3 Hybrid *CDN* Architecture

As shown before, both *commercial* architecture and *academic* architecture present some drawbacks. *Commercial* architecture is cost-effective for large scale applications only, and *Academic* architecture's quality is dependant on the number of *peers*. The solution to solving the above problems might be a compromise between both. Currently, there are a number of Hybrid Content Delivery Networks propositions. This alternative provides a solution for

both cost reduction and performance guarantee. It works as a smaller CDN of distributed low cost nodes that dynamically calculate and accelerate content, and might or might not have auxiliary storage servers as backup, and use Peers's storage resources for mirroring content. "Distributed Content Delivery Network" [4] being an example of a Hybrid *CDN* that doesn't use storage servers.

### 3.1 Application Scope

This architecture model could be used in any situation since it delivers both reasonable costs and guarantees reasonable performance at all times.

### 3.2 Example of a Hybrid CDN

The following example explains the model presented in M. Jaisson Paul and K. Ibrahim's paper [4].

**Framework** The DCDN are composed by a well defined hierarchical structure that allows a better relationship among the network's elements. This way, it is possible to define the following entities (from higher to the lower level of hierarchy):

- **Content provider** - creator and manager of the shared content;
- **Administrators** - a group of users with privileges on the network responsible for the network's maintenance and support
- **Servers** - a set of servers, that are utilized, not for data storage, but for the efficient distribution of network content. They can be *Master* type (directly connected to the *content provider*, distributing the content by the various regions where the *CDN* acts) or *Local* (connected to the *master servers* and channel the content to the *surrogates* acting on a closer level to the client).
- **Surrogates** - set of users of P2P network whose resources are shared;
- **Client** - final entity, that gets the content;

**Content Distribution Method** Given that the main point of D *CDN* is to optimize the content access and minimize its costs, it makes sense that the content replicas are as close as possible to the client, i.e., on the *surrogates*. This way, content distribution is made sequentially in the following way:

1. The *content providers* ask permission to *administrators* to insert a new content on the network;
2. If the request is successful, *master servers* send the updated content to the *local servers* and from them to the *surrogates*;
3. At the same time that the content is sent, servers update their records with the same contents being that the *master* will own more general data (like the network area where the content is distributed) and the *local* more specific informations (i.e., that *surrogates* are in possession of the contents);
4. When there is a content access request by the *client*, the *local server* will choose the valid *surrogates* for the content share. This way, this is distributed between the *surrogates* and the *client* through the *P2P* network.

### 3.3 Main Advantages

The main advantage of this architecture, when compared to the *comercial*, are the reduced costs of implementation due to the lack of storage servers requirement. When compared to the *academic* architecture, this model assures a better content stability and availability and better route calculation, since it does not depend solely on Peers availability.

### 3.4 Associated Challenges

This architecture has some drawbacks, the main being that this is only an hypothetical model, based on simulations and few to none real world implementations. Another problem spotted, compared to the *academic* and *comercial CDNs*, *Distributed CDN (DCDN)* adds some levels of complexity with hierarchy, which may difficult a physical implementation. The migration process from a CDN to another is unecessarily difficult due to incompatibility between different CDN providers. The lack of standards pay a big role here that should be dealt with as soon as possible[6].

## 4 List of Current CDN Providers

The following list contains the most popular CDN service providers, almost all of them being commercial CDNs. This list provides a brief explanation on each CDN specialisation. It is important to keep in mind that this is a brief list containing only a few as an example of the variety of CDN specialisations.

### **Akamai**

It provides content delivery and streaming media services, along with global traffic management. According to [1], Akamai, being the biggest CDN provider, is responsible for as much as 15%–30% of the internet traffic.

### **AT&T Intelligent Content Distribution Service**

Is responsible for monitoring the original website in content and replicates those changes within minutes on mirror sites across their worldwide networks and data facilities.

### **Adero's GlobalWise Network**

Is a worldwide network of intelligent, multi-server nodes that include carrier-grade Sun Enterprise servers and best-of-breed software specialised in websites.

### **AppStream's Virtual Hosting Network**

Monitors the usage of central databases and applications, segments them, and proactively moves the computing resources to application servers closest to the users that need them.

### **CacheFlow**

Provides unique solutions for three areas: enterprise, service provider, and e-commerce.

### **Digital Island**

Provides delivery of all major kinds of content, including streaming media, and features multiple authentication methods to provide secure content delivery.

### **Dynamai**

Caches static plus some dynamic content using a kind of reverse proxy in front of the clients server(s). Ensures updated dynamic content using an event-driven feedback loop that notifies the Dynamai cache when dynamic content is updated.

### **Edgix**

A satellite-based Internet content delivery company providing high-speed Edge Services to Internet Service Providers worldwide. ISPs gain greater bandwidth efficiency because they are able to offload a significant amount of traffic from their backbone connection to Edgix's network.

### **epicRealm**

Providing an improvement in the e-business experience for both parties, users experience quicker response times, prioritised transactions and continual communication with their host while businesses serve the freshest content to their customers all the time.

### **Mirror Image Internet**

Mirror Image Internet provides global content distribution, streaming, and caching services that speed Internet traffic and improve quality of service.

### **SolidSpeed Network**

SolidSpeed uses intelligent routing and network optimisation to bypass internet bottlenecks. They work to find the most efficient route between customers and content.

### **Speedera's Content Delivery Network**

Speedera's CDN provides an integrated, innovative and customer-focused service for powering content providers. Content is pushed from web origin sites to caching servers at the "edge" of the Internet, closer to users.

### **XOSoft**

XOSoft's CloneQuick is the first CDN to employ delta encoding, which sends only the changes of documents. XOSoft combines mirrors and caches to synchronise content worldwide, and deliver fresh content to users quickly.

## **5 Conclusion and Future Work**

Based on the above analysis, Commercial CDNs will continue to be needed for the near future because of its ability to deliver uncompromised performance[5]. On the other hand, P2P architectures are most likely to be replaced with DCDN architecture, as soon an open source DCDN alternative is available. The later will be possible as soon as infrastructure required is standardised and made available[4]. Despite the promising theoretical results, further practical testing is needed to take better conclusions.

## **References**

1. Anya George Tharakan, Subrat Patnaik: "Strong dollar hurts Akamai's profit forecast, shares fall" (<http://www.reuters.com/article/2015/04/28/us-akamai-tech-results-idUSKBN0NJ2IV20150428>)
2. Erik Nygren, Ramesh K. Sitaraman, and Jennifer Sun.: "The Akamai Network: A Platform for High-Performance Internet Applications" (ACM SIGOPS Operating Systems Review, vol. 44, no. 3, July 2010)
3. "How Content Delivery Networks Work" (<http://www.CDNetworks.com/blog/how-content-delivery-networks-work/>). Networks. Retrieved 22 September 2015.
4. Jaison Paul Mulerikkal, Ibrahim Khalil: "An Architecture for Distributed Content Delivery Network" (School of Computer Science, RMIT University, Melbourne 3000, Australia)
5. Hai Jiang ; Jun Li ; Zhongcheng Li ; Xiangyu Bai: "Performance Evaluation of Content Distribution in Hybrid CDN-P2P Network" (Future Generation Communication and Networking, 2008. FGNCN '08. Second International Conference Vol 1)
6. Menai, M.F.; Fieau, F.; Souk, A.; Jaworski, S.: "Demonstration of Standard IPTV Content Delivery Network Architecture Interfaces: Prototype of Standardized IPTV Unicast Content Delivery Server Selection Mechanisms" (<http://ieeexplore.ieee.org/xpls/icp.jsp?arnumber=4785012>)