



# A Survey on Dynamic Data Delivery through Content Delivery Networks

Vijaya Murari T., Dr.K.C.Ravishankar

.NMAM Institute of Technology/Department of CSE, Nitte, 574110, India  
Government Engineering College/Department of CSE, Hassan, 573201, India  
\*Corresponding Author Email [vijayamurari.t@nitte.edu.in](mailto:vijayamurari.t@nitte.edu.in) , [kcr@gechassan.ac.in](mailto:kcr@gechassan.ac.in)

## Abstract

The growth of the internet users has been exponential over a period of time. This has led to the increased use of web for sharing content and services. Huge load on the bandwidth of the internet and the large number of requests for the content and services present in the website is due to the increased use of web for variety of purposes. This has restricted the ability of the internet to send the required data to the users in a time bound manner. The emergence of Content Delivery Networks (CDN's) has helped to alleviate these limitations by providing the necessary infrastructure and mechanisms to the internet. This paper focuses on CDN's, overview of CDN's, dynamic data delivery through CDN along with the merits and demerits, various Streaming Content Distribution Networks (SCDN) and challenges in streaming multimedia over CDN.

**Keywords:** Content Delivery Network, dynamic data, Streaming Content Distribution Networks, streaming, security.

## 1. Introduction

The internet's growth has been tremendous over the last few decades. Internet provided the facility to share the content and the services effortlessly and seamlessly by its users. As the number of users grew, demand for the content and services also grew. Due to the heavy demand by the users, the internet could not provide the content and the services at the user perceived level. Such a sudden increase in the requests for web content is termed as flash crowds. Flash crowds increase the requests on the web server(s) and then make them temporarily unavailable [1].

A Content Delivery Network (CDN) consists of network items which are spread over the Internet and are collaborative in nature. The efficient delivery of data to the end users is done by replicating the data over several replicas of the web servers. Association between the distributed CDN components can occur over nodes in both homogeneous and heterogeneous environments [1]. A set of surrogate servers are used by content delivery architecture which deliver the copies of content to the users. Fig.1 depicts the distribution of the content by a Content Delivery Network [1]:

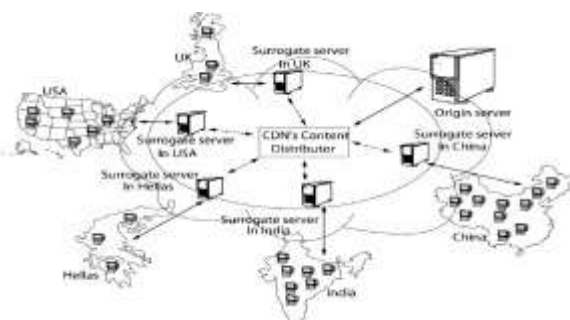


Fig.1: Model Of A Content Delivery Network

When content or a service is requested by a user from a server, it is directed to a replicate CDN server close to it and the required content is provided by that CDN server. The CDN distributes the content and the services to the globally scattered web servers so that the end users can get the content and the services in a dependable and timely manner. The duplication of the content will be carried out in two ways: either on-demand when users request for it or it can be replicated beforehand, by pushing the content to the distributed Web servers [1]. Initially the development of CDN's was done for the distribution of frequently required image files from the web servers which were popular. The present day CDN's support the delivery of any dynamic content which includes different varieties of streaming media [2].

In this paper, section II gives the overview of the content delivery networks. In section III, we examine the various strategies to stream the dynamic data via CDN along with its advantages and disadvantages. In section IV, we throw light on the various Streaming Content Distribution Networks (SCDN) such as commercial, academic and cloud based CDNs. Section V provides challenges in streaming multimedia over CDN.

## 2. Overview

A CDN is responsible for sending the content to the end users in an efficient and appropriate way. Efficient and in time delivery of the content is possible due to the strategic placing of the surrogate servers across the world. Many researchers have come up with algorithms like tree based, greedy and hotspot to place the surrogate servers. These algorithms specify the surrogate server placement with low infrastructure cost and improved performance. Another important factor in delivering the content with Quality of Service (QoS) guarantees is to determine the optimal number of surrogate servers which determines cost of the traffic pattern. The two typical ways to agree on the best possible number of surrogate

(proxy) servers are:

- **Single ISP approach-** In this approach, 40 surrogate servers is deployed at the network edge. Due to this, the ISP's get required geographical coverage and they need not depend on other ISP's. A disadvantage in this approach is that proximity of the surrogate servers will be less with CDN's clients.
- **Multi-ISP approach-** Here, numerous servers are deployed where the global ISP's are present. The objective is to place the surrogate servers close to the clients and then serve the content reliably and quickly. The advantage of this approach is that it reduces the number of access points required to deliver the content. The disadvantage is that if each of the surrogate servers receives fewer hits then it will lead to poor performance of the CDN [3].

The first approach suits better with sites having low to medium traffic and the second approach is good when the sites have high traffic [3]. The functionality of a Content Delivery Network is given as follows:

- **Request redirection and content delivery services:** This functionality uses the mechanisms to bypass the congestion and routes the request to the closest possible surrogate servers.
- **Content outsourcing and distribution services:** The content must be replicated and cached so that it can be distributed among the various proxy servers on behalf of the origin server.
- **Content Negotiation services:** are necessary to satisfy the particular needs of the user or group of users.
- **Management services:** will help in managing the network components, handle the accounting and will monitor and report on content usage [4].

### 3. Dynamic Data through Cdn

A Content Delivery Network is responsible for the delivery of static and dynamic data. Hitherto to CDN there were data grids, distributed databases and peer to peer networks to distribute the resources and thereby provide performance gain to the end users. The objective of the data grid system is to distribute the resources among various locations; Distributed databases are leveraged for processing the queries, optimization and management; Peer to peer networks is designed for direct sharing of computer resources and it suits individual content providers. The objective of CDN is to cache the content closer to user's location and provide a quick response to user's query. In fact, some CDN's use the peer to peer networks to deliver the content required by the end users. In this section, we focus on the different platforms with various strategies to deliver a kind of dynamic data that is, interactive media streaming.

#### 3.1. Cooperative Media on Demand on the InterNet (COMODIN)

In [5][6], they have described the cooperative networking strategy to deliver the dynamic data. Fortino "et al" [5] have developed a collaborative on demand platform known as Cooperative Media On Demand on the InterNet (COMODIN). This platform provides an internet based interactive media service like e-learning and e-entertainment among the selective group of clients who cooperate with each other to control the remote media playback. According to them, COMODIN's system architecture is organized into two planes:

- **The Base plane:** delivers on-demand archived media streams on an open CDN platform. The base plane consists of three components:
  - a. **Origin server:** Media objects to be distributed are stored in this server. It provides Uniform Resource Identifier (URI) namespace for these objects to be distributed and delivered by the CDN system. Surrogates get the content from CDN through bulk transfer mechanisms over the distribution network.
  - b. **Surrogate:** is the partial copy of the original server where it temporarily stores the content and delivers it to the clients. They also share the popular content with other surrogates. Surrogate component contains following main modules:
    - **Portal:** provides access to the media content stored in the CDN through HTTP-based server. File transfer from origin servers to surrogates and between the surrogates is also supported by the portal. The portal leverages the database  $DB_{Surgt}$  that contains a list of all available media objects and information.
    - **Media Streaming Server (MSS):** streams the requested media to clients by using Real-time Transport Protocol or Real Time Control Protocol (RTP/RTCP) and Real Time Streaming Protocol (RTSP) is used for the control of streaming media.
    - **Local monitor agent (LMA).** This is a Simple Network Management Protocol (SNMP) agent which monitors the surrogate's performance periodically (for example, CPU and RAM usage, available disk space, and number of Transmission Control Protocol or TCP connections) and for each given client performs an on-demand network measurements.
  - c. **Client:** is an individual PC that requests specific media content stored in the CDN. Through a Web browser client issues a request and the QuickTime plug-in is used to view the media object. A Domain Name System-based redirection is used by them to optimize CDN performance and the local DNS client cache of the clients is eliminated so that new requests aren't biased by previous DNS resolutions.

The Base plane, supports the regulation of optimal surrogate selection, surrogate monitoring, and location and flow of content throughout the CDN with the help of the control components like redirector and content manager.

- **The Collaborative plane:** provides collaborative play back services with the help of components and protocols for group organization, group-based streaming control, and interactive questioning. The components of collaborative plane are:
  - a. **Collaborative Playback Session Manager (CPSM):** organizes the group and coordinates the component based on the CPS Management protocol. The formation of the group, initiation of the subscription or Unsubscription, joining, leaving and termination of Collaborative Playback Session's (CPS) is performed by the component of CPSM.
  - b. **Collaborative Playback Control Server (CPCS):** is based on the Multipolicy Collaborative Playback Control Protocol (MCPCP). CPCS is integrated with MSS and is located in surrogates. The media streaming shared among the members of a CPS served by the same surrogate can be remote controlled with the help of CPCS. Multiple CPSs are served by CPCS spawning a front end for each CPS whose members are served by the MSS.

- c. **CPCS Coordination Channel (CCC):** coordinates the distributed CPCSs through the Coordination Channel Protocol (CCP). CCC coordinates the CPCS front ends serving the same CPS by spawning a front end for each initiated CPS. Coordination between CPCS front ends is necessary for deciding which CPCS front end is enabled to accept a control command, for synchronizing the CPCS front ends with respect to the media session time, and for allowing clients to join asynchronously.
- d. **Comodin client:** is a multithreaded Java-based multimedia application that provides an interface between the user and the Comodin system.

In COMODIN, there is a text based stateful protocol known as CPS Management Protocol (CMP) which supports the interaction between the pair of components like <CC, CPSM>, <CPSM, CCC> and <CPCS, CPSM> through the appropriately defined messages. Multi-policy Collaborative Playback Control Protocol is used for the interaction between CC and CPCS which is based on the MCPCP that adopts a multi-policy based approach for managing the control commands (for example, pause, play, seek, and stop) that are issued by clients to control the media playback. Coordination Channel Protocol (CCP) is a text-based stateful protocol that allows the interaction between CPCS and CCC and coordinates the CPCS front ends of the same CPS.

The merits and possible demerits of their proposal have been identified by them based on the comparison of the performance of the CDN architecture with that of a one-server architecture, referred to as Star. In star architecture, the media content is maintained and provided by a single MSS server located in some other location with all the clients directly connected to it. Comparison between the CDN architecture and the Star architecture is carried out based on the cooperative and non cooperative modes. The performance evaluation of their proposal is done based on the reports of denial probability that is the probability that a client request is not actually served, because either the local CPCS or the remote CCC server denies the request. According to their implementation, irrespective of the mode (cooperative or Non-cooperative), the CDN architecture has lower denial probability than the Star architecture. The denial probability is reduced between 50 to 60 percent due to the combined adoption of CDN architecture with cooperative mechanism. For medium and large sized groups, the CDN with the cooperation mechanism has advantages over the one server architecture. For small sized groups (fewer than six clients), there is a considerable overhead imposed by the CDN architecture which is higher than obtainable benefits, thus making it more efficient to use the one-server architecture.

### 3.2. Cooperative Networking (CoopNet) Project

In [6], Padmanabhan "et al" have described a strategy in which live streaming and on demand media content can be distributed through cooperative networking. Multiple Description Coding (MDC) techniques is used in [6][7][8] for distributing the media content down the distribution trees. Cooperative networking delivers the content on the basis of peer to peer distribution. Cooperative networking approach doesn't replace the client server paradigm; but complements which means that there will be a server that hosts clients and directly serves it to clients. Cooperative networking is invoked only when server finds it difficult to handle the requests posed by the clients.

According to Padmanabhan "et al", the basic protocol for streaming live media (it may be real time or recorded) is based on the centralized server which processes node joins and leaves. Server has the knowledge of the topology of all the distribution trees. When a node wants to join the system, it has to first contact the server. The new node also informs the server about its available network bandwidth to serve future downstream nodes.

The server responds with a list of designated parent nodes, one per distribution tree. The designated parent node in each tree is chosen as follows. With the server as the starting point, search for the one or more nodes that have the necessary spare capacity (primarily network bandwidth) to serve as the parent of the new node is carried out at different levels of the distribution tree. The server then randomly picks up one such node and designates that a parent of the new node. Upon receiving the server's message, the new node sends (concurrent) messages to the designated parent nodes to get linked up as a child in each distribution tree. Nodes may depart in one of the two ways: graceful departure or node failure. In first case, the node which intends to leave the tree informs the server about it. Using the top-down procedure described above, the server identifies the children of the departing node and executes a join operation one each child (and implicitly the sub tree rooted at the child) for each distribution tree. In the latter case, the node leaves the tree suddenly and does not notify the either the server or any other node of its departure.

In multiple description coding method, the audio and/or video signals are encoded into  $M > 1$  separate streams or descriptions such that any subset of these descriptions can be received at the receiving end. The signal received will have distortions (with respect to original signal) depending on the number of descriptions received i.e., if more number of descriptions are received, then the decoded signal will have less number of distortions and the quality of the received signal would be high.

On demand streaming is concerned with the distribution of recorded streaming media content on demand. In this context, there will be many unsynchronized streams corresponding to different users. On receiving such a request, the server streams the data if the current load of the server permits it. If the server is overloaded, then it sends a response message consisting a list of IP addresses of clients who have downloaded (part or all) of the requested stream and have shown interest in cooperative networking. The requesting client then communicates one or more of these peers to download the required content. On demand streaming addresses the following issues:

1. If the peer is having a part of the requested content, the client determines which part of the requested content is available at peer during the initial handshake and then plans to request required missing content with the corresponding peers.
2. The untimely departure of the peer nodes and/or network connectivity problems with one or more peers. This is handled by a technique known as distributed streaming where a stream is subdivided into a number of sub streams each of which may be served by a different peer. Each sub stream corresponds to a description created using MDC.

In [6], the performance evaluation of live streaming through cooperative networking is based on trace characteristics, effectiveness of MDC and impact of repair time. According to their trace characteristics, the highest number of simultaneous clients exceeds 17,000. More than 70% of the clients tune into the live stream for less than a minute. Short lifetimes may be because of the poor quality of the video which caused frustration among the users and made them leave the system. Based on their two simulation experiments, the MDC has clear advantage over Single Distribution Tree and significant tree diversity is maintained over a long period of time. According their results on the repair time, when a repair time is 1 second, on an average, client would receive 90% of descriptions whereas with a 10 second repair time only 30% descriptions are available. In the performance evaluation of on demand streaming, they carried out simulation on over 3, 00,000 requests during 2 hour period. However, only 6% or 18,000 requests were successfully served at an average rate of 20 Mbps with mean session duration of 20 minutes.

### 3.3. Multiple Descriptions CDN (MD-CDN)

During the transmission of the static data over the network, if there is a packet loss, conventional approaches deal with it by retransmitting the data which may not be possible in a streaming context. Thus, streaming media delivery over packet networks requires additional mechanisms. Among the different techniques to improve quality of streaming media, a method of multiple description coding (MDC) with path diversity was proposed by John Apostolopoulos "et al" [7]. In their work, they exploit the virtue of CDN in which it provides multiple paths of different characteristics to its clients by replicating the same content at multiple geographically or topologically separated surrogates. Their MD-CDN architecture addresses the issues like the distribution of multiple descriptions across the surrogates, MD surrogate selection and MD surrogate placement. Their work is based on the MD video coder present in [8][9]. Some important characteristics are: (1) high compression efficiency where the MDC properties are achieved with only slightly higher total bit rate than conventional Single Description (SD) compression schemes, (2) It provides the ability to use correctly received descriptions to repair corrupted descriptions over time, (3) It provides ability to successfully operate over paths that support different or unbalanced bit rates and (4) standard compatibility is provided as this MD coder is a standard-compatible enhancement to MPEG-4 Version 2 (with NEWPRED) and H.263 Version 2 (with RPS). They make a study of the loss characteristics of SD and MD video streams with the help of two test sequences. They have used Foreman which is head and shoulders type sequence similar to a video conferencing application and a Bus sequence similar to a conventional movie. These sequences were coded with MD and SD at the same constant video quality and the same total bit rate. Then they have compared the performance of MD and SD video coding based on single or burst packet losses. Based on their results, they conclude that as long as losses afflict only one channel at a time, then MD coding is more resilient to single losses and burst losses than SD coding. If the losses affect both the channels SD recovers more quickly by using the extra intra frame coding. Based on their simulation experiments on various topologies and settings, they found that MD streaming performs better than SD streaming in SD streaming biased environments. Finally, they conclude that combining MD coding with path diversity from a CDN can provide significant performance benefits over a conventional SD-CDN.

### 3.4. Decentralized Media Streaming Infrastructure (DeMSI)

In [6][7][8], the media content (on demand or live) is streamed through multiple paths of the network using multiple description coding. In their strategy, content provider doesn't have any control over routing the media content over multiple paths which is a major disadvantage. In [11], the streaming media is stored in multiple server clusters at different geographical positions where the server clusters act like proxies when providing service to the consumer nodes. Due to this arrangement (known as edge architecture), whenever a consumer node requires a streaming media service, it has to contact the proxy server local to it which in turn reduces the latency and number of hops required to media content to reach the consumer node. However, the system suffers from a single point of failure as the stream is pushed from single source over single path. In order to overcome the drawbacks of previous strategies, in [11], Alan Kin Wah Yim "et al" have proposed the Decentralized Media Streaming Infrastructure (DeMSI) whose primary objective is to reduce the cost of storing the content and workload of video distribution in CDN by delegating the resources like streaming server, network and storage to consumer/subscriber machines and their available upstream internet bandwidth without affecting the video quality. DeMSI has designed in such a way that it can stream Constant Bit

Rate (CBR)/ Variable Bit Rate (VBR) video of any format. Here, each peer stores only a small portion of the content and multiple peer subscribers stream the same video content to the consuming peer.

The motivation for the development of DeMSI is the most popular CDN models for general file sharing applications and infrastructures like Napster, Gnutella, FastTrack, eDonkey and Bit Torrent. DeMSI is also inspired by another popular P2P CDN model known as Application Level Multicast (ALM). The concept of ALM is used in CoopNet by Padmanabhan "et al" [6], DirectStream [12], P2Cast [13] and SpreadIt [14]. DeMSI also incorporates the concept of aggregated streaming in which multiple peers push sub streams of same media content simultaneously to a consuming peer by sharing the network bandwidth that is originally required for a single media stream. Like any other peer to peer application DeMSI has to deal with the following reliability issues: (1) Connection between the serving peers to consuming peers may be congested which will result in delays and packet loss (2) A peer may go down normally or abnormally (3) As the content is stored at peer end, the integrity of the content becomes vulnerable. In order to deal with the first two issues, DeMSI consumer has to make a selection among the list of candidate subscriber peers discovered or previously contacted by others as consumers so that following goals can be achieved: (1) maximum utilization of the network and peers (2) Minimum number of peers should serve the content (3) Frequency of rescheduling during the course of streaming should be minimum. Rescheduling is done in two ways:

1. **Proactive Scheduling:** In this type of scheduling, the actively serving peer should have largest end to end streaming bandwidth, smallest packet loss rate, largest portion of the content and should share no or very few congested links with other actively serving peers. The consumer peer constantly monitors for each serving peer with these network metrics and stores them in their knowledge database. During streaming session, consumer peer incrementally learns which peer connections are sharing the congested links in the network without contributing additional overhead on streams.
2. **Reactive Scheduling:** The network characteristics of the peer consumer connections and the availability of the peer change over time. Rescheduling of streaming tasks is done using a highly adaptive, flexible divide and conquer based scheduling algorithm.

In DeMSI, the integrity of the content is protected by SHA-1. During the request for streaming, the consumer sends the SHA-1 code of the content segment to the target peer. The peer then verifies the SHA-1 code against the local copy and replies either by commencing the stream or by sending a negative acknowledgement. The design and implementation of DeMSI is done as group of autonomous agents. The architecture of DeMSI consists of the following components:

1. **Storage:** Every subscriber peer stores a part of the media content and not the full replication of it. This leads to the need for dividing media content into segments before distributing it to peers. The segments that are local to a peer and the segments received from other peers are stored in Segment Cache. These segments are encoded using the Reed-Solomon based Forward Error Correction (FEC) algorithm which helps to avoid re-transmission of lost packets during the streaming.
2. **Peer Cache:** is a semi persistent knowledge base of the peers discovered for monitoring and serving as active peers. This peer cache doesn't maintain the global collection of peers.
3. **Peer Hunter:** Peer hunter agent is used by DeMSI to find peers that carry on or more of the k segments required by the

requested content. It uses peer selection technique that makes use of the network congestion avoidance strategy.

4. **Scheduler:** is an agent that is responsible for the coordination of the task like peer hunting and dispatching various streaming and peer monitoring tasks to be carried out during the streaming session based on the request from Player agent.
5. **Segment receiver:** is an agent which listens to the UDP port for fragments from active serving peers.
6. **Peer Monitor:** does the following jobs for each active serving peer: 1) Sends a request in order to measure the round trip time between the consumer and the active serving peer.2) Sends the remaining redundant fragments of the segment through another group of candidate peers by assigning another instance of the Re-scheduler agent. 3) Examines the dynamic service level information at an interval of one per second.
7. **Re-scheduler:** is an agent that helps in overcoming the lagging rate of aggregated streaming and time to play deadlines by providing additional peers.
8. **Segment sender:** Based on the delivery request from the consumer, this agent delivers either the part of a segment or as a whole based on the context of a sub-stream of fragments.

In order to support the aggregated media streaming by decentralized collection of unreliable subscriber resources, they have proposed the DEMSI architecture. Based on the experimentation, they conclude that: (1) According to the goals set on peer selection strategy, using the congestion avoidance strategy during peer selection helps in getting better performance than the use of best bandwidth first strategy (2) Their approach provides unhindered reactive rescheduling of aggregated streaming of tasks. (3) Their algorithm becomes vulnerable due to false positives obtained from the correlation tests. However, by the use of their conservative approaches during correlation tests have resulted in the significant reduction of in the number of false positives in the results.

#### 4. Summary

The benefit(s) and drawback(s) of the projects with their strategies to stream the media content over the CDN are given in the table 1 below:

**Table 1:**summary

Project	Strategy	Type of distribution coding	Benefit(s)	Drawback(s)
COMODIN	Cooperative networking	Single Distribution Coding	1.Beneficial for medium and large sized groups	1.There is a considerable overhead for a group with less than six clients
CoopNet	Cooperative networking	Multiple Description Coding	1.Tree diversity is maintained due to multiple description coding 2. More than 70% of the clients subscribed for very short	1. In the context of live streaming, clients left the system due to poor quality of the videos. 2. 30% descriptions are

			time in the context of live streaming. 3. 90% of the descriptions were available to the clients when the repair time is 1 second.	available when the repair time is 10 seconds. 3. Only 6% of the clients were served with video in the situation of on demand streaming.
MD-CDN	Path diversity	Multiple Description Coding	1. The resilience of MD coding is higher to single losses and burst losses than SD coding when losses affect one channel at a time. 2. The performance of MD streaming is superior to SD streaming in SD streaming biased environments .	1. If the losses affect both the channels SD recovers more quickly by using the extra intra frame coding.
DeMSI	Aggregated Media Streaming	Single Distribution Coding	1. Provides better performance by using the congestion avoidance strategy.	1. Provides false positives which makes the inference algorithm vulnerable.

### 5. Streaming Content Distribution Networks

The objective of Streaming Content Distribution Networks (SCDN's) is enhancing the streaming of the multimedia data to the end users. The content distribution networks are classified into three categories: commercial CDNs, academic CDNs and cloud based CDNs. In this section, a brief overview on commercial CDNs, academic CDNs and challenges faced in streaming multimedia content over CDN is provided.

#### 5.1. Commercial CDNs

Commercial CDNs must provide high quality of experience to its customers by using the existing technical infrastructure and framework. Commercial CDN providers fall into one of the two categories of the business model that is, either content centric or business centric. In the Content centric business model, providers of the data/content pay providers of CDN services to speed up the delivery of their own data through the network to the end-user devices which enhances the quality of experience to the end users. Akamai is an example of content centric CDN. The access centric business model is similar to content centric business model in every aspect except the revenue flow where the access provider stakeholders pay the access centric CDN. Edgix, Orblinx and Skyserver are examples of access centric CDNs [15]. There are various commercial CDNs providing services to customers. They are:

- **Accellion-** provides solutions for managing the data and appliances for transferring the files securely when demanded

over the network. Distributed storage of files and network setup for enterprise applications provided by the Smart Edge Operating System (SEOS) technology is used by the Accellion products to transfer, duplicate and handle the large sized files. Secure exchange of files can be done using an on-demand file transfer solution known as Accellion Courier Secure File Transfer Appliances (SFTA). Using SFTA, large attachments outside of email infrastructure can be sent with convenience of email experienced by both sender and receiver. Industries in the area like manufacturing, advertising/media production, healthcare, consumer goods, higher education etc are covered by Accellion products [4].

- **Akamai**- servers provide the delivery of static, dynamic content and the streaming of audio and video files. The main objective of developing Akamai was to solve the flash crowd problem. Akamai handles the flash crowd problem by providing more servers to the sites experiencing the high load problem while serving all the clients from nearby servers. Akamai continuously monitor the state of services provided by their servers and networks using the DNS based load balancing technique. Static as well as dynamic content is handled by the Edge Platform; applications are managed by Edge Control and Network Operations Control Center (NOCC) is used to control the network operations [4]. Edge Platform, Edge Control and NOCC are products of Akamai to oversee the content delivery over the networks. Akamai's deployment of the most pervasive, highly-distributed content delivery network (CDN) consists of more than 1,600 networks around the world with more than 233,000 servers spanning across 130 countries [15][16].
- **Limelight Networks**- High performance, cost effective delivery services of data like video, images, files, music from websites to nearly any device present in any part of the world is provided by Limelight Content delivery services. Limelight Orchestrate platform contains huge global private infrastructure, which can send the multimedia data with advanced capabilities and services supporting the data transfer. Limelight Orchestrate platform deliver the content to customers with low latency, high reliability and best origin offload through Content Delivery Services. Content Delivery Services ensure quick delivery of the content to users by placing the content of the website in caches of edge servers, speeding up the last mile delivery, tailoring the requests and responses, and making the suitable access a surety [17].
- **Mirror Image**- optimal delivery of the real time data content is done by the Mirror Image using a global Dynamic Delivery Network which is patented by that company. It provides functionalities like caching of the content and downloading of files to a wide range of desktops, mobile devices, IPTV set-top boxes and other devices.

Content Access Point (CAP) in the Mirror Image global Dynamic Delivery Network delegates customers' origin servers and moves the content closer to the user and therefore ensuring optimal performance by the website. The user experience is enriched by allowing large object file downloads and global content caching [18].

## 5.2. Academic CDNs

Video contents are streamed and cached by the academic CDNs for research purposes in domain of CDN. The research in the distribution of multimedia data with quantitative measurable goals is enhanced by the experimental nature of the academic CDNs. The popular academic CDNs are CoDeeN, CoralCDN, GlobuleCDN and Flash Crowd Alleviation Network (FCAN) [4][15].

- **CoDeeN**- Network Systems Group at Princeton University, USA has developed an academic test bed known as CoDeeN on top of the PlanetLab. This test bed is comprised by a network of high-performance proxy servers. Currently, proxy servers have been deployed on many PlanetLab nodes. These proxy servers behave both as request redirectors and server surrogates. They cooperate with each other and collectively provide a fast and robust web content delivery service to CoDeeN users. The number of projects similar to CoDeeN are: CoBlitz (a scalable Web-based distribution service for large files), CoDeploy (an efficient synchronization tool for PlanetLab slices), CoDNS (a fast and reliable name lookup service), CoTop (a command-line activity monitoring tool for PlanetLab), CoMon (a Web-based general node/slice monitor that monitors most PlanetLab nodes), CoVisualize (a visualization tool graphically displaying PlanetLab activity) [19].

## 5.3 Cloud based CDN

Earlier, the content providers delivered their multimedia content to the end users with the help of third party CDNs. With the evolution of multimedia content in size and quality, it is a serious issue for the data providers to deliver the content using the existing CDN. The emergence of the cloud based Content Distribution Network has led the content providers to move from traditional CDNs hosted on private data centers to cloud based CDNs to take advantage of the geographical availability and the availability of infrastructure without owning it. Cloud based CDNs (CCDN) provide a flexible solution to the content providers to place their content on one or more cloud storage servers based on coverage budget and Quality Of Services preferences [20][21]. Rackspace Cloud Files, Amazon CloudFront, MetaCDN, Limelight Orchestrate, MediaWise Cloud provide data delivery as service via cloud computing. The major advantages provided by CCDNs are: 1) It allows the users to use the delivered content in a cost effective way based on the model of content delivery provided by cloud. 2) It increases the location of data and presence of CDN. 3) It supports the interoperability of CDNs. 4) It supports dynamic changes in load of the network which in turn facilitates the CDNs to support different kinds of applications having content which is either unpredictable and bursting or predictable bursting, size up and size down of resources and ability to expand and grow fast [20]. There are challenges which are specific to the CDN in a CCDN that affects the performance of service within the cloud infrastructure. They are [20][22]:

1. Network proximity: Customers' experience on the services offered via CDN is improved by reducing the response time.
2. Load Balancing: Response time and system throughput is improved by distributing the load to different nodes in a network.
3. Local Caching: Content required by the customer is fetched from origin server and stored in the local server.
4. Request redirecting: The performance of a CDN server is dependent on the redirection of the customer's request to the nearest cache server.

## 5.4 Challenges in Streaming Multimedia over CDN

With the boom in streaming multimedia content over the unreliable internet through the applications like video on demand and peer to peer content delivery, providing the security services such as confidentiality and authentication to those applications becomes imperative. Traditional techniques for authenticating the data cannot be used for securing the multimedia content sent over

the unreliable channel. The various parameters to be considered during the authentication of streaming multimedia are:

1. **Verification Probability:** Receiving end must be able to authenticate all the packets received by it. This causes high overhead and computational costs. Therefore, there is a requirement for other techniques where the probability of verification is high but with lower computational costs.
2. **Computational overhead:** As the multimedia content has huge amount of continuous data, efficient computing resources are necessary for the generation of the signature at one end(the sender) and verification of the same at the other end(the receiver). The requirement becomes even critical for mobile devices with restricted computational capabilities
3. **Communication overhead:** Transmission of the authentication data along with the multimedia content requires additional transmission rate which becomes an overhead when the total available data rate is limited.
4. **Sender Delay:** Before transmission, there will be additional delay due to authentication processing which can be reduced by placing a huge storage space at the sender.
5. **Receiver Delay:** is the delay that occurs between the time of receiving of the packet and time of authenticating the packet by the receiver. Often, a large buffer is required at the receiver to deal with the high receiver delay. Usually, the streaming media has deadline to play the media files after which they become useless. So, design of the delay at the receiver side due to authentication should be such that the packet does not miss its deadline.

Qibin Sun "et al" classify the authentication schemes for the multimedia data delivered from one end to the other end into techniques based on content and streams.

1. **Content-based authentication:** Content level authentication of the media stream is the main objective of this category of approaches. In this technique, media content is authenticated by authenticating the features of the media rather than media data.
2. **Stream Based Authentication:** In this approach, authentication of the streaming multimedia is carried out at the stream or packet level. They are further classified into methods based on ECC (Error correcting code) and methods based on graphs. Qibin Sun "et al" focus their approach on graph based methods.

These two categories of approaches for authenticating streaming multimedia can be robust to the loss of packets. However, employing them for specific applications is a challenging issue. Both the approaches provide benefits which are opposite to one another. Stream based approaches provide the similar security to the system as the traditional techniques for data security. The stream based approaches can withstand the loss of packets but usually not to other manipulations. The content-based approaches can be designed in such a way where it can withstand the problems related to packet delivery. However, they have weak mathematical proof or lower empirically tested levels of security. So, it is better to combine the two categories of approaches to provide robustness to both the loss of packets and other problems during the data delivery.[21].

## 6. Conclusion

Present day applications like live webcast of sports events/on demand streaming of movies require accurate and efficient delivery of the dynamic data to its subscribers. Content delivery

networks are responsible for the delivery of both static and dynamic data. In this paper, we have focused on the various strategies used to deliver the dynamic data via CDN, advantages and disadvantages of these strategies. Some systems deliver the dynamic data based on the peer to peer resource sharing model; while others exploit the criteria like collaborative playback services, Multi description coding and path diversity for delivering the dynamic data. Finally, we describe the different types of SCDNs and their characteristics and challenges in streaming multimedia over CDN.

## References

- [1] Rajkumar Buyya, Mukaddim Pathan, Athena Vakali, "Content Delivery Networks", Springer-Verlag Berlin Heidelberg, 2008.
- [2] Novella Bartolini, Emiliano Casalicchio, and Salvatore Tucci, "A Walk through Content Delivery Networks", Springer-Verlag Berlin Heidelberg, 2004.
- [3] Athena Vakali, George Pallis, Aristotle University of Thessaloniki, "Content Delivery Networks: Status and Trends", IEEE Computer Society, November - December 2003.
- [4] Al-Mukaddim Khan Pathan, Rajkumar Buyya, "A Taxonomy and Survey of Content Delivery Networks", 2007, <http://www.cloudbus.org/reports/CDN-taxonomy.pdf>.
- [5] Giancarlo Fortino, Wilma Russo, Carlo Mastroianni, Carlos E. Palau, Manuel Esteve, "CDN-Supported Collaborative Media Streaming Control", IEEE Computer Society, 2007.
- [6] Venkata N. Padmanabhan, Helen J. Wang, Philip A. Chou, Kunwadee Sripanidkulchai, "Distributing Streaming Media Content Using Cooperative Networking", ACM, NOSSDAV'02, May 12-14, 2002.
- [7] John Apostolopoulos, Tina Wong, Wai-tian Tan, Susie Wee, "On Multiple Description Streaming with Content Delivery Networks", IEEE INFOCOM, July 2002.
- [8] J.G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," Visual Communications and Image Processing (VCIP), January 2001.
- [9] J.G. Apostolopoulos, "Error-resilient video compression via multiple state streams," Proceedings of International Workshop on Very Low Bitrate Video Coding (VLBV'99), pp. 168-171, October 1999.
- [10] Yao Wang, Amy R. Reibman, Shunan Lin, "Multiple Description Coding for Video Delivery", Proceedings Of The IEEE, VOL. 93, NO. 1, January 2005.
- [11] Alan Kin Wah Yim, Rajkumar Buyya, "Decentralized media streaming infrastructure (DeMSI): An adaptive and high-performance peer-to-peer content delivery network", Journal of Systems Architecture 52 (2006) Elsevier 737-772.
- [12] Yang Guo, Kyoungwon Suh, Jim Kurose, and Don Towsley, "A Peer-to-Peer On-Demand Streaming Service and Its Performance Evaluation", Proceedings of 2003 IEEE International conference on Multimedia and Expo, Baltimore, MD July 2003.
- [13] Yang Guo, Kyoungwon Suh, Jim Kurose, and Don Towsley, "P2Cast: Peer-to-peer Patching Scheme for VoD Service", Proceedings of the 12th World Wide Web Conference, Budapest, Hungary, May 2003.
- [14] H. Deshpande, M. Bawa, H. Garcia-Molina, "Streaming Live Media over a Peer-to-peer Network", Technical report, Stanford University, August 2001.
- [15] Muslim Elkotob, Karl Andersson, "Challenges and Opportunities in Content Distribution Networks: A Case Study", the 4th IEEE International Workshop on Mobility Management in the Networks of the Future World, 2012.
- [16] <https://www.akamai.com/us/en/about/facts-figures.jsp> as accessed on 14/11/2017.
- [17] <https://www.limelight.com/resources/data-sheet/content-delivery> as accessed on 14/11/2017.

- [18] <http://www.mirror-image.com/services/content-delivery> as accessed on 15/11/2017.
- [19] <http://codeen.cs.princeton.edu/> as accessed on 15/11/2017.
- [20] Wang M. "et al". (2015),"An Overview of Cloud Based Content Delivery Networks: Research Dimensions and State-of-the-Art" In: Hameurlain A., Küng J., Wagner R., Sakr S., Wang L., Zomaya A. (eds) Transactions on Large-Scale Data- and Knowledge-Centered Systems XX. Lecture Notes in Computer Science, vol 9070. Springer, Berlin, Heidelberg.
- [21] [21] Qibin Sun "et al"," Quality-Optimized and Secure End-to-End Authentication for Media Delivery", proceedings of the IEEE, volume :96, Issue:1, January 2008.

### Authors' Profiles



**Vijaya Murari T.** is currently working as Assistant Professor in the Department of CSE at NMAM Institute of Technology, Nitte. He has completed his B.E in Computer Science and Engineering from VTU, Belagavi and M.Tech in Computer Science and Engineering from NITK, Surathkal. He has thirteen years of teaching experience

and five journal and four conference publications. His main research interests are in the area of cryptography, multimedia processing, cloud computing and data mining.



**Dr. K. C. Ravishankar** is currently working as professor in the Department of CSE and Principal at Government Engineering College, Hassan. He completed his B.E. from MCE, Hassan in 1990, M.Tech from IIT, Delhi in 1998 and Ph.D. from Visvesvaraya Technological University in 2009. He served at MCE from 1990-2010. His areas of interest

include Databases, Image Processing and Cryptography. He has 6 international and 16 national publications to his credit. He is the Life member of Institution of Engineers, Indian Society for Technical Education and Computer Society of India.