An overview of Cloud based Content Delivery Networks: Research Dimensions and state-of-the-art

Meisong Wang[#$], Prem Prakash Jayaraman[#], Rajiv Ranjan[#], Karan Mitra*, Miranda Zhang[#$], Eddie Li[#$], Samee Khan[^], Mukkaddim Pathan[*], Dimitrios Georgeakopoulos[&]

[$]Research School of Computer Science, ANU, Canberra Australia
[#]CSIRO DP&S Flagship, Canberra, Australia
*Luleå University of Technology, SE-931 87 Skellefteå, Sweden
[^]North Dakota State University, USA
[*]Telstra Corporation, Australia
[&]Royal Melbourne Institute of Technology, Australia

Abstract. Content distribution networks (CDNs) using cloud resources such as storage and compute have started to emerge. Unlike traditional CDNs hosted on private data centers, cloud-based CDNs take advantage of the geographical availability and the pay-as-you-go model of cloud platforms. The Cloud-based CDNs (CCDNs) promote content-delivery-as-a-service cloud model. Though CDNs and CCDNs share similar functionalities, introduction of cloud impose additional challenges that have to be addressed for a successful CCDN deployment. Several papers have tried to address the issues and challenges around CDN with varying degree of success. However, to the best of our knowledge there is no clear articulation of issues and challenges problems within the context of cloud-based CDNs. Hence, this paper aims to identify the open challenges in cloud-based CDNs. In this regard, we present an overview of cloud-based CDN followed by a detailed discussion on open challenges and research dimensions. We present a state-of-the-art survey on current commercial and research/academic CCDNs. Finally, we present a comprehensive analysis of current CCDNs against the identified research dimensions.

1 Introduction

The digital universe is doubling in size every two years. It is expected that the data we create and copy will reach 44 zettabytes by 2020 [23]. The global internet video traffic alone will comprise 79 percent of all Internet traffic in 2016, up from 66 percent in 2013 [22]. In our current Internet-driven world, consumers expect fast, always-on data access from anywhere and any device. As a result, content providers are expected to confront with the challenge of delivering optimised and streaming content to application running on devices including tablets and smart-phones while ensuring high-speed access and superior performance. The major challenges that the emerging applications bring to the future internet [27] include the requirements of: 1) higher scalability, 2) higher capability, 3) higher quality of service (QoS), 4) stronger interactivity, 5) dealing with heterogeneity (e.g., device, network and application) and 6) security. Content delivery networks (CDNs) are often required to face the data deluge to efficiently and securely distribute content to a large number of online users. The growth of related technologies such as accelerated web performance, rich media content streaming, IPTV, management and delivery of user generated content over the last decade has led to the significant adoption of CDNs. Cisco has estimated that over half of the internet traffic generated will be carried out by content delivery networks by 2018.

A CDN is a distributed network of servers and file storage devices that replicates content/services (e.g. files, video, audio etc) on a large number of surrogate systems placed at various locations, distributed across the globe. CDNs are highly flexible and aims to improve the quality and scalability of the services offered over the Internet by reducing the latency and efficiency of delivering contents to clients. The CDN maximises the bandwidth for accessing to data from clients throughout the network by strategically placing content replica(s) at geographically distributed locations. The concept of a CDN was conceived during the early days of Internet. By the end of 1990's before CDNs from Akamai and other commercial providers managed to deliver Web content (i.e., web pages, text, graphics, URLs and scripts) anywhere in the world, and at the same time meet the high availability and quality expected by their end users. Today, Akamai [26] delivers between fifteen to thirty percent of all Web traffic, reaching more than 4 terabits per second.

In today's dynamic Internet landscape, it is more important than ever for content and service providers to understand the requirements and demands of users. For instance, consider a video distribution services such as Netflix, YouTube and Quickflix. When delivering video content to geographically distributed subscribers, the video experience can vary depending on the delivery path to the subscriber. Studies [25, 24] show that, the sensitivity of subscribers to video quality issues can greatly impact the subscriptions to the services offered by the video distribution service providers.

Cloud computing is an emerging computing model where a myriad of virtualized ICT resources are exposed as web utilities, which can be invoked and released in an on-demand fashion [28, 29]. The concept of cloud computing is an immediate extension of many well researched domains such as virtualisation, distributed, utility, cluster, and grid computing. The most comprehensive, widely used and referred definition of cloud computing in the literature is presented in [16]. It defines cloud computing as "A model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction". A number of public cloud providers including Amazon Web Services (AWS), Microsoft Azure, Salesforce.com and Google App Engine have been emerged to be very successful in the recent past. The advent of virtualization has led to the transformation of traditional data centres into flexible cloud infrastructure.

In the days before cloud, the main way to address issues regarding performance, availability and scale in CDN was for companies to physically replicate existing infrastructure in other geographical locations in order to decrease the physical distance between the end user and content servers. For example, deploy servers close to ISP gateways. This approach was not only expensive but companies had to determine the best replicate and server placement strategy [30]. The cloud model offers companies an alternative and less expensive way to expand infrastructure, in particular the ability to virtually scale across unlimited resources on demand without the need to buy expensive hardware. The cloud and CDN have both evolved to be complimentary utility platforms. The cloud provides virtually unlimited access to computational resources (processing, storage and network infrastructure) via array of physical servers deployed globally. Conversely, CDN provides an optimised repeatable delivery of content from servers to end users (one-to-many). Using the cloud and CDN together can deliver a holistic agile system that meets CDN demands and is economically viable. A cloud-based CDN architecture can provide the following advantages [31]:

- An elastic platform with ability to dynamically and easily scale capacity up and down
- Hides the infrastructure complexity from CDN applications and content provides
- Enable a QoS driven performance management

- Open standard approach to tap into the capabilities of public clouds to scale during peak demand

A few studies [4, 27, 32-36] in the past have investigated CDN presenting overview and technical challenges in designing and implementing effective CDNs. Most of the work has focused on commercial CDNs that work over private data centres. With the current trends and advances in cloud computing and the mutual advantages that can be leveraged by cloud and CDN, in this paper, we present a comprehensive study of Cloud CDNs. We present a state-of-the art survey on current commercial and research driven Cloud CDNs. We then present an analysis of current Cloud CDN based on a comprehensive taxonomy. We finally identify the opportunities in the Cloud CDN area.

2 Content Delivery Network and Cloud computing

2.1 Content Delivery Network

An overview of a typical CDN architecture is presented in Fig. 1. Depending on application and content type the architecture of CDNs may vary However, all CDN architectures mainly comprise of an origin server, a request redirecting mechanism and a large number of surrogate cache servers namely Point of Presence (POP).

Fig. 1. The Architecture of a Content Delivery network

1. **Origin server***:* is a powerful storage system that contains all the content and/or the metadata of all the content. To achieve high performance of the whole CDN, the content in the origin server are pushed to the POP servers (surrogate servers) that are located at different geographical locations across the globe.
2. **POP servers**: are distributed in a large numbers at diverse areas in a CDN. The main function of pop server is to offer the content based on user request. When the content is not available locally, the pop server should pull it from the origin server and store it for the next probable requirement; as it might be possible that the same/other user(s) in the region will require the content. Prefetching [2] is another important functionality provided by the POP server where it fetches the content that clients may be interested in from the origin server thereby reducing the chance of traffic congestion especially during the high demand. Needless to say, prefetching needs to predict the users' preferential contents by synthesizing and analysing the historical information such as access logs. It is evident that this kind of

prefetching techniques may require statistical data mining algorithms to determine what content to prefetch.

3. **Request Redirecting mechanism**: One of the functions of a CDN is to dynamically redirect clients to the most optimal servers based on several QoS parameters such as server load, latency, network congestion, client access networks, and proximity etc. There are a variety of methods that can be used to implement this mechanism as presented in Table 1 [3].

Table 1. CDN Request Redirecting Mechanisms

Global Server Load Balancing	Global awareness
	Smart authoritative DNS
DNS-based request routing	
HTTP redirection	
URL rewriting	URL modification
	Automation through scripts
Anycasting	IP anycast
	Application level anycast
CDN Peering	Centralized directory model
	Distributed Hash Table
	Flooded request model
	Document routing model

Global Server Load Balancing (GSLB): aims to optimize resource use, maximize throughput, minimize response time, and avoid overload of any one of the resources. The capabilities that allow global server load balancing include global awareness and smart authoritative domain name service (DNS). In GSLB, services nodes are aware of information and status of other service nodes. This provides intermediate switching nodes to be globally aware. To make use of the global awareness, intermediate switches act as smart authoritative DNS, each switching between the best surrogate servers.

DNS-based request-routing: is widely used in the Internet. DNS based request-routing is also used in many CDNs because of its ubiquity as a directory service. DNS servers handle the domain name of the desired web site or content. The client initiates a name lookup in a local DNS server, which is supposed to return the address of a surrogate server near the client. If local DNS cache misses, it forwards the name lookup to the DNS root server. DNS root server returns the address of the authoritative DNS server for the web site. The Authoritative DNS server then returns the address of a surrogate server near the client based on specialized routing, load monitoring and Internet mapping mechanism. Finally, the client retrieves the content from the designated surrogate server. A number of studies have examined and reported the performance and effectiveness of DNS [37, 38].

HTTP Redirection: takes advantage of the HTTP protocol’s redirection feature. This mechanism builds on special Web servers that can inspect a client request and chooses the most suitable surrogate server and redirect the client to those servers. This approach provides the flexibility of managing replication with finer granularity (e.g., at page level). However, it does pose significant overheads due to the introduction of extra messages round trips.

URL Rewriting: can be one of the best and quickest ways to improve the usability and search friendliness. A rewrite engine is software located in a Web application framework running on a Web server that modifies a web URL's appearance. Many framework users have come to refer to this feature as a "Router". This modification is called URL rewriting. For example, request for web sites with images, the router can rewrite the URLs of the images to point to the best surrogate servers.

Anycasting: is a new routing technology based on the Ipv6. It is a methodology in which datagrams from a single sender are routed to the topologically nearest node in a group of potential receivers, though it may be sent to several nodes, all identified by the same destination address. CDNs may use anycast for routing user request to their distribution centres or DNS.

CDN Peering: is a methodology where clients provide resources; the client can also use these resources based on their requirements. This means that unlike client-server systems, the content serving capacity of peer-to-peer networks can actually increase as more users begin to access the content (especially with protocols such as Bittorrent that require users to share). This property is one of the major advantages of using P2P networks because it makes the setup and running costs very small for the original content distributor. To locate the content in CDN peering [38], a centralised directory model, distributed hash table, flooded request model or document routing model can be used. In centralised P2P file-sharing service, a large server is used to provide directory service. The P2P application contacts the directory service, informing the directory service of its IP address and the names of objects in its local disk that it is making available for sharing. When an active peer obtains a new object or removes one, it informs the directory server, which then updates its database. In a distributed hash table, peers are indexed through hashing keys and are found through complex queries within a distributed system. This approach is good in performing load balancing and offloading loads to less-loaded peers. The flooded request model is simple but scales poorly. When a node wants to find a resource on the network, which may be on a node it does not know about, it could simply broadcast its search query to its immediate neighbours. If the neighbours do not have the resource, it then asks its neighbours to forward the query. This is repeated until the resource is found or all the nodes have been contacted, or perhaps a network-imposed hop limit is reached

2.2 Cloud Computing

Cloud computing [39, 40] assembles large networks of virtualised ICT services such as hardware resources (such as CPU, storage, and network), software resources (such as databases, application servers, and web servers) and applications. The advent of virtualization has led to the transformation of traditional data centres into flexible cloud infrastructure [49,50]. With the benefit of virtualization, data centres progressively provide flexible online application service hosting [18] such as: web hosting, search, e-mails, and gaming. Largely, virtualization provides the opportunity to achieve high availability of applications in data centres at reduced costs. In industry, these services are referred to as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [51,52]. Cloud computing services are hosted in large data centres, often referred to as data farms, operated by companies such as Amazon [10], Apple, Google, and Microsoft [8]. Cloud computing gives developers the ability to marshal virtually infinite computing and storage based on amount of data to be processed and stored; and number of people to be notified in real time. Cloud-based ICT resources can be acquired under pay-per-use models and as needed, instead of requiring upfront investments in resources that may never be used optimally. As defined by National Institute of Standards and Technology [16], the five essential characterises of cloud computing are:

- On-demand self-service

- Broad network access
- Resource pooling,
- Rapid elasticity
- Measured service

Another important characteristic of cloud computing that is gaining significant momentum is Quality of Service (QoS) driven service delivery. For reliable and efficient management of application performance hosted on the *aaS layers, system administrators have to be fully aware of the compute, storage, networking resources, application performance and their respective quality of service (QoS). QoS parameters (e.g, latency, renting cost, throughput, etc.) play an important role in maintaining the grade of services delivered to the application consumer and administrator as specified and agreed upon in the Service Level Agreement (SLA) document. The SLA guarantees scope and nature of an agreed QoS performance objective (also referred to as the QoS targets) that the cloud application consumer and administrators can expect from cloud service provider(s).

Though the notion of virtually unlimited resources is true in many aspects, there are practical limitations to the realisation of this concept. For example, how to automatically provision new resources as the demand for the service increases. Previous work on resource provisioning in distributed computing environments [7, 41] enables its users to manually modify the hardware resources of their running job flows.

3 Cloud CDNs

CDNs have made a significant impact on how content is delivered via the Internet to the end-users [25]. Traditionally content providers have relied on third-party CDNs to deliver their content to end-users. With the ever changing landscape of content types e.g. moving for standard definition video to high definition to full high definition, it is a challenge for content providers who either supplement their existing delivery networks with third-party providers or completely rely on them to understand and monitor the performance of their service. Moreover, the performance of the CDN is impacted by the geographical availability of the third-party infrastructure. A cloud CDN (CCDN) provides a flexible solution allowing content providers to intelligently match and place content on one or more cloud storage servers based on coverage, budget and QoS preferences [27]. The key implication is economies of scale and the benefits delivered by the pay-as-you-go model. Using clouds the content providers have more agility in managing situations such as flash crowds avoiding the need to invest in infrastructure development.

As stated previously, clouds provide the end users with virtually infinite pool of compute and storage resources with no capital investment in terms of hardware and software. Therefore, CCDN systems can be very valuable in data processing and delivery of content over the Internet. The main advantage of such a system would be that they provide a cheaper means of hosting and deploying multi-tiered applications that can scale based on the usage demands. Further clouds offer not only cheaper content storage and distribution functionality, but also compute functionality such that application and data processing can also be performed on clouds. Lastly, cloud offers pay-as-you-model whereby the end-users can start and terminate the cloud resources based on the amount of money they are willing to spend hosting their services without entering into a complex contract with the cloud provider. Migration from traditional client/server based CDNs to cloud computing model is a major transformation that introduces great opportunities and challenges. The major advantages and opportunities introduced by CCDNs include:

1. *Pay-as-you-go CCDN model*: CCDN allows the users to consume the delivery content using a pay-as-you-go model. Hence, it would be much more cost-effective than owning the physical infrastructure that is necessary for the users to be the part of CDN.
2. *Increased point-of-presence:* The content is moved closer to users with relative ease in the CCDN system than the traditional CDN due to the omnipresence of cloud. The Cloud-based content delivery network can reduce the transmission latency as it can rent operating resources from the cloud provider to increase the reach and visibility of the CDN on-demand
3. *CCDN Interoperability*: CDN interoperability has emerged as a strategic important concept for service providers and content providers. Interoperability of CDNs via the cloud will allow content providers to reach new markets and regions and support nomadic users. E.g., instead of setting up an infrastructure to serve a small group of customers in Africa, taking advantage of current cloud providers in the region to dynamically host surrogate servers.
4. *Support for variety of CCDN application*: The cloud can support dynamic changes in load. This will facilitate the CDNs to support different kinds of applications that have unpredictable bursting traffic, predictable bursting traffic, scale up and scale down of resources and ability to expand and grow fast.

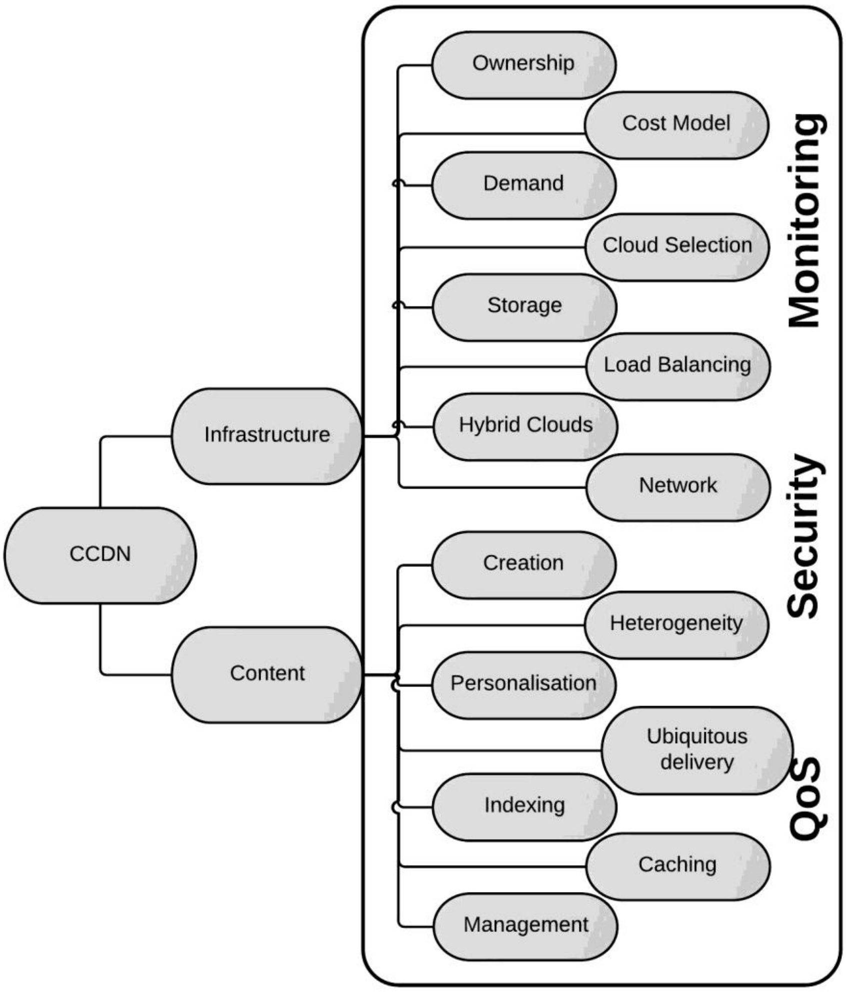
However, while cloud-based CDNs [5, 7] have made a remarkable progress in the past five years, they are still limited in a number of aspects. For instance, moving into the cloud might carry some marked security and performance challenges that can impact the efficiency and productivity of the CDN thus affecting the client's business. Further, current CCDNs are more suited to distributing static content such as audio, video and text. They are not well suited to serving dynamic content-based applications such as collaborative audio-video processing and streaming. Moreover, CDNs are usually owned by private and telecommunication companies making these services costly to the end-users as they have to enter in legal contract to use CDN services. A categorical list of technical issues and challenges in CCDN system is presented in Fig. 2 and the following sections.

3.1 Dynamic Content Management

CDNs are designed for streaming staged content but do not perform well in situations where content is produced dynamically. This is typically the case when content is produced, managed and consumed in collaborative activities. For example, an art teacher may find and discuss movies from different film archives; the students may then edit the selected movies. Parts of them may be used in producing new movies that will be sent to the students' friends for comments and suggestions. Current CDNs do not support such collaborative activities that involve dynamic content creation.

3.2 Content Creation

Traditional CDNs are not designed to manage content (e.g., find and play high definition movies). This is typically done by CDN applications [47, 48]. For example, CDNs do not provide services that allow an individual to create a streaming music video service combining music videos from an existing content source on the Internet (e.g., YouTube), his/her personal collection, and from live performances he/she attends using his/her smart phone to capture such content. This can only be done by an application managing where and when the CDN will deliver the video component of his/her music program. With CCDN, the end-user will act as both content creator and consumer. CCDN needs to support this feature



inherently. User-generated content distribution is emerging as one of the dominant forms in the global media market.

Fig. 2. Classification of CCDN Challenges and Issues

3.3 Content Heterogeneity

Existing Web 2.0 technologies currently support the authoring of structured multimedia content (e.g., web pages linking images, sounds, videos, and animations). The CCDNs will need to extend and broaden existing Web 2.0 strengths with a new environment aimed at supporting the creation and consumption of interactive multimedia content (e.g., interactive audio and video), as well as other novel forms of multimedia content (e.g., virtual and augmented reality) that are currently not supported by existing Web 2.0 technologies and tools.

3.4 CCDN Ownership

Cloud CDN service providers either own all the services they use to run their CDN services or they outsource this to a single cloud provider. A specialized legal and technical relationship is required to make the CDN work in the latter case

3.5 CCDN Personalisation

CDNs do not support content personalization. For example, if the subscriber's behaviour and usage pattern can be observed, a better estimation on the traffic demand can be achieved. The performance of content delivery is moving from speed and latency to on-demand delivery of relevant content matching end-user's interest and context.

3.6 Cost models for Cloud CDNs

The cloud cost model works well as long as the network consumption is predictable for both service provider and end-user. However, such predictions become very challenging with distributed cloud CDNs.

3.7 Security

CDNs also impose security challenges due to the introduction public clouds to store, share and route content. The use of multi vendor public clouds further complicates this problem. Security is the protection of content against unauthorised usage, modification, tampering and protection against illegal use, hack attacks, viruses and other unwanted intrusions. Further, security also plays an important role while accessing and delivering content to relevant users [44].

3.8 Hybrid Clouds

The integration of cloud and CDN will also allow the development of hybrid CCDN that can leverage on a combination and private and public cloud providers. E.g. the content provider can use a combination of cloud service platforms offered by Microsoft Azure and Amazon AWS to host their content. Depending on the pay-as-you go model, the content provider can also move from one cloud provider to another. However, achieving a hybrid model is very challenging due to various CCDN ownership issues and QoS issues.

3.9 CCDN Monitoring

The CCDNs can deliver end-to-end QoS monitoring by tracking the overall service availability and pinpoint issues. Clouds can also provide additional tools for monitoring specific content e.g. video quality monitoring. However, developing a CCDN monitoring framework is always a challenge.

3.10 CCDN QoS

With the notion of virtually unlimited resources offered by the cloud, quality for service plays a key role in CCDNs to maintain a balance between service delivery quality and cost. Defining appropriate SLA's to enforce QoS and guarantee service quality is very important and is also challenging. Further, the notion of hybrid clouds further complicate CCDN QoS challenges due to the involvement of multiple cloud providers with varying SLAs. CCDNs must accommodate highly transient, unpredictable users behaviour (arrival patterns, service time distributions, I/O system behaviours, user profile, network usage, etc.) and activities (streaming, searching, editing, and downloading).

3.11 CCDN Demand Prediction

It is critical that CCDNs are able to predict the demands and behaviours of hosted applications, so that it can manage the cloud resources optimally. Concrete prediction or forecasting models must be built before the demands and behaviours of CDN applications can be predicted accurately. The hard challenge is to accurately identify and continuously learn the most important behaviours and accurately compute statistical prediction functions based on the observed demands and behaviours such as request arrival pattern, service time distributions, I/O system behaviours, user profile, and network usage.

3.12 CCDN Cloud Selection

The diversity of offering by Cloud providers make cloud section to host CDN components a complex task. A practical question to be addressed is: how well does a cloud provider perform compared to the other providers? For example, how does a CDN application engineer compare the cost/performance features of CPU, storage, and network resources offered by Amazon EC2, Microsoft Azure, GoGrid, FelxiScale, TerreMark, and RackSpace. For instance, a low-end CPU resource of Microsoft Azure is 30% more expensive than the comparable Amazon EC2 CPU resource, but it can process CDN application workload twice as quickly. Similarly, a CDN application engineer may choose one provider for storage intensive applications and another for computation intensive CDN applications. Hence, there is need to develop novel decision making framework that can analyse existing cloud providers to help CDN service engineers in making optimal selection decisions.

3.13 Ubiquitous content delivery

Content delivery services will interact with the network and appropriately adjust its QoS as needed to deliver content to a specific user based on content and user requirements for maintaining its integrity, the device the user is using, his/her location, and the service contracts. This is a requirement for CCDNs with the growing complexity in media types, end-user access devices and intermediate network architectures.

3.14 Flexible content storage, compression, and indexing

Cloud storage resources allow content producers to store content on virtualized disks and access them anytime from any point on the Internet. These storage resources are different from the local storage (for example, the local hard drive) in each CPU resource (e.g., Amazon EC2 instance types), which is temporary or non-persistent and cannot be directly accessed by other instances of CPU resources. Multiple storage resource types are available for building content orchestrator. Naturally, the choice of a particular storage resource type stems from the format (e.g., structured vs. unstructured) of the content. For instance, Azure Blob (https://azure.microsoft.com/en-us/) and Amazon S3 (http://aws.amazon.com/) storage resources can hold video, audio, photos, archived email messages, or anything else, and allow applications to store and access content in a very flexible way. In contrast, NoSQL (Not Only SQL) storage resources have recently emerged to complement traditional database systems [12]. Amazon SimpleDB, Microsoft Azure Table Storage, Google App Engine Datastore, MongoDB, and Cassandra are some of the popular offerings in this category.

Though cloud environments are decentralized by nature, existing CDN application architecture tends to be designed based on centralized network models. It is worth noting that none of the existing cloud storage resources exposes content indexing APIs. It is up to the CDN application designer to come-up with efficient indexing structure that can scale to large content sizes to help end-users find and retrieve relevant content effectively and efficiently. To facilitate new and better ways of content delivery using CCDNs, advanced distributed algorithms need to be developed for indexing, browsing, filtering, searching and updating the vast amount of information.

3.15 Other Challenges

Apart from the above CCDN specific challenges, there are also several important factors specific to the CDN in a CCDN that affect the performance of service within the cloud infrastructure. These include [1]:

- *Network proximity*: reduces the response time for improving the customers' experience about the services offered via the CDN.
- *Load balancing*: improves the capability of the whole network by decreasing the flash crowd situation i.e., it distributes the load to different nodes in a network such that response times and system throughput improve.
- *Local caching*: fetches the content for the customer from the origin server and stores it in a local server closer to the customer. This technique helps in significantly reducing the response time.
- *Request redirecting*: plays a very pivotal role in the performance of a CDN service as it redirects the customer's request to the nearest cache server.

4 CCDNs Architecture and Services

Technically, architectures of CCDNs in existence are various in terms of the correlation between CDN and Cloud. For instance, some CCDNs adopt the cloud-based store as their origin server. In this kind of CCDNs, the general mechanism is similar as the traditional CDNs'. Other architecture includes Master/Slave mechanism [4]. Specifically, in this kind of CCDNs [43, 14, 15, 27, 4] the functionality of master node is managing, monitoring and provisioning slave nodes on demand. The slave nodes combine the functions of POP servers. The data is replicated in the master nodes which act as the origin server. When the slave node has to get some contents users require, it only needs to communicate with the master node to fetch the content. A typical CCDN architecture is presented in Fig 3. As depicted in the Fig. 3, the POPs are distributed across multiple cloud providers while the master node/origin server is responsible to orchestrate the entire CDN functionality. Based on demand from various geographical locations and QoS constraints, the master node will fire new slave POP nodes in close proximity to origin of user requests.

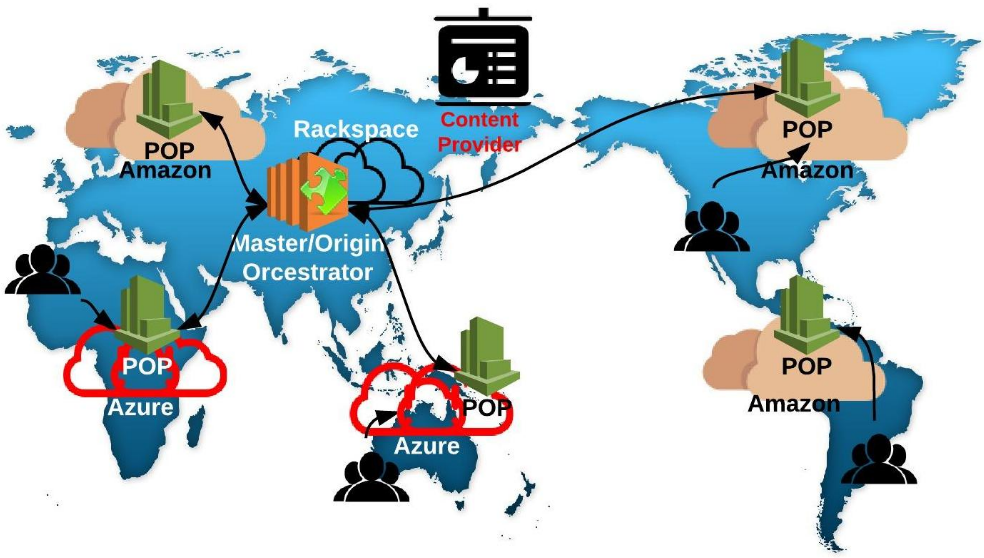


Fig. 3. Cloud CDN Typical Architecture

Cloud-based CDNs offer a large number of additional services compared to traditional CDNs. These include:

Cloud Security. The Cloud-based CDN providers can combine CDN security with the performance of cloud-based distributed infrastructures to keep their customers' websites both high performance and secure. The following aspects are some instances of services the providers may provide in this area:

1. Data Security: The Cloud-based CDN providers can apply and support advanced standards and methods of security like PCI compliance, secure socket layer, and digital rights management to offer the protection of their customers' data.
2. High Availability: The providers offer the Cloud-based delivery of robust website and application functionality in a high-performance manner.
3. Cloud DDoS Protection: It means the protection of websites via DDoS mitigation. The CDN can support proactive monitoring and alerting.
4. Regulatory Compliance: The Cloud-based CDN enhance the CDN infrastructures and services to meet the requirement of industrial and governmental standards for protecting the customers' personal or financial information.

Cloud-Based DNS. The Domain Name System (DNS) is a very important Internet infrastructure that enables visitors to reach their website. DNS redirection has a very crucial role to play in a Content Delivery Network, for it enables the users to get the content they want from the available nearest surrogate server to reduce the response time. Fetching the users' preferential content form the optimal cache server can not only improve the performance but reduce the chance of traffic congestion especially during the rush time. Actually, DNS redirection is one of the mainly two techniques the most CDN providers adopt in their architecture to achieve the aim of redirecting the clients to the nearest surrogate server, the other is URL re-writing.

Cloud Storage. Taking advantage of Cloud Storage is a significant difference between the Cloud-based CDN and the traditional CDN. For users, to Store, maintain, and deliver a great mess of media, software, documents, or other digital objects is an essential part of ensuring an outstanding online experience. By using the Cloud storage functionality, the clients can effectively store a great amount of data and serve these data to the users who need such data in different locations over the world reliably and fastly. Furthermore, it is a very economical option. Though most Cloud-based CDN providers in today's world allege their Cloud storage designed for reliability, scale or speed, anyway they always mention the advantages of Cloud storage as more as possible, there are also some limitations of the Cloud storage [6]:

1. Due to some defective causes of the machines in the Cloud, the users' data which are stored in a Cloud Storage system can be corrupted and this would lead to the situation that the Cloud Storage system returns incorrect results to the users.
2. The attacker may make a bug in the user's programs to steal valuable information, even control the user's client to do Dos attacks or to spam.
3. In some rush time, because of the traffic jam, users might not be able to get access to their data which are stored in the Cloud storage system accidentally.

The above disadvantages of Cloud storage may be extremely impossible, if the Cloud Storage system is robust, well-management, well-designed etc. However, the first importance of the fact is that when these bad events happen, it will be very difficult to find that who should be responded for that among the Cloud provider, the CDN provider and the Customer when something goes wrong. As a consequence, it would be necessary to build an accountable Cloud system which means it is easy to find whose false when some mistakes happen in such kind of Cloud system. Implementing data mining algorithms to analyze the log system is a good choice to address this problem.

Cloud Load Balancer. The Cloud load balancer provides the customers the flexibility to manage their content delivery strategy. This service enables customers to specify content delivery policies based on real-time conditions and user targets. The typical cloud load balancing technology manages the customers' application traffic and makes decisions of where to route it. When a node in the Cloud system fails, a health check process will remove it from rotation to keep maximum availability of the whole system. The Cloud load balancer service should follow the pay-as-you-go model as well in term of the hours the customers use, number of current connections and bandwidth [7].

Cloud Orchestrator. Cloud orchestration service offers enhanced flexibility and elasticity of CCDN as it manages public and private cloud resources using the pay-as-you-go model. Cloud orchestration operations include: (i) production: create and edit; (ii) storage: uploading and scaling of storage space; (iii) keyword-based content tagging and searching and (iv) distribution: streaming and downloading. At Cloud service level, the orchestrator capabilities span across a range of operations such as selection, assembly, deployment of cloud resources to monitoring their run-time QoS statistics (e.g., latency, utilization, and throughput). The orchestrate supports deployment, configuration and monitoring of content and cloud resources deployed across hybrid cloud platforms using web-based widgets. These widgets hide the underlying complexity related to cloud resources and provide an easy do-it-yourself interface for content management. The cloud orchestration service is also responsible to manage the cloud resources based on service providers SLAs.

5 Existing Cloud-based CDNs

The current landscape of CCDNs leverages the flexibility of the cloud to easily and quickly distribute content across the internet. The CCDNs landscape diversifies into two primary forms namely web site content distribution and media distribution. Web site content focuses mostly on serving static pages with a combination of text and other media content while the media delivery CCDNs are dedicated to deliver high speed video form content providers such as Netflix. The majority of the system use the architecture presented in Fig 3 with proprietary implementation of cloud storage architecture, security, DNS, load balancer, CDN orchestrator and indexing mechanisms. In this section, we will analyse the current state-of-the are in commercial and academic/research based CCDN solutions.

5.1 Rackspace Cloud Files

RackSpace offers ”Cloud Files” [9] as a Cloud-based CDN service where the customers can use virtually unlimited and on-demand cloud storage and high speed content delivery over the Internet all over the world. The high-level architecture of Cloud Files is shown in Fig. 4. As ”Cloud Files” is a cloud based system, it offers pay-as-you-go model which means that the users only need to pay for the amount of storage and network bandwidth based on the actual usage. The ”Cloud Files” takes advantage of the

Akamai Content Delivery Network to deliver the content worldwide. Akamai CDN is one of the largest CDN providers in the world and has a large number of surrogate servers around the world so that the content access latency is significantly minimized even if the customers are far away from the origin server. In terms of content hosting, Cloud Files make use of OpenStack for file storage functionality. The Cloud Files supports API to that cloud and CDN resources can be managed programmatically. The Cloud Files system uses Time-to-Live (TTL) timers to manage content that change dynamically. The dynamic content to be shared using the CDN are associated with a CDN-enabled container. The TTL of the container navigates to each file in the container. When the TTL expires, the edge servers (POPs) will synchronise with the origin server to update the changed content. It is not possible to have a TTL associate with each individual file within a CDN-enabled container.

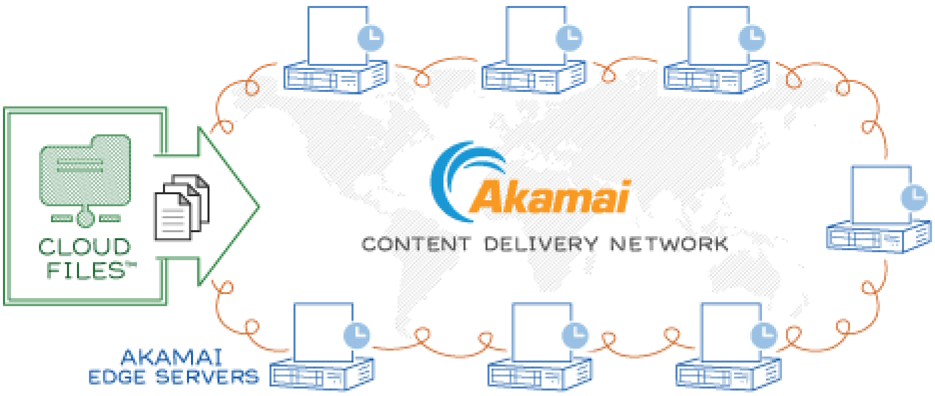


Fig. 4. Architecture of Rackspace Cloud Files

5.2 Amazon CloudFront

Amazon offers CloudFront as a content delivery service that can be integrated with their widely popular Amazon Elastic Cloud Compute service [10]. Similar to Rackspace, CloudFont is also offered as a pay-as-you-go model and supports both static and dynamic content delivery along with live media streaming functionality. Using CloudFront, the customers can store their content on the origin servers, or use the Amazon's Cloud Store service (Amazon S3).The customers can use simple APIs or the AWS Management Console to register their origin servers with Amazon CloudFront. When the customer has more than one server, he/she can use URL pattern matching to find which origin server has the content, and the customer can assign one of those origin servers as the default server. The most significant feature of Amazon's CloudFront is that it can be co-operated with several other Amazon Cloud Services. The architecture of interactions between Amazon CloudFront and other AWS services is presented in Fig 5. One of the major difference between Rackspace Cloud Files and Amazon CloudFront is that Rackspace utilizes Akamai CDN service that offers 219 CDN edge locations worldwide compared to only 32 CDN edge locations offered by Amazon. The CloudFront enables handing of dynamic content while delivering web content that change for each end-user. It uses the concept of URL pattern matching which has to be defined for the dynamic content being served to control the cache behaviour. When a URL match succeeds for a dynamic content request, the corresponding cache behaviour is invoked.

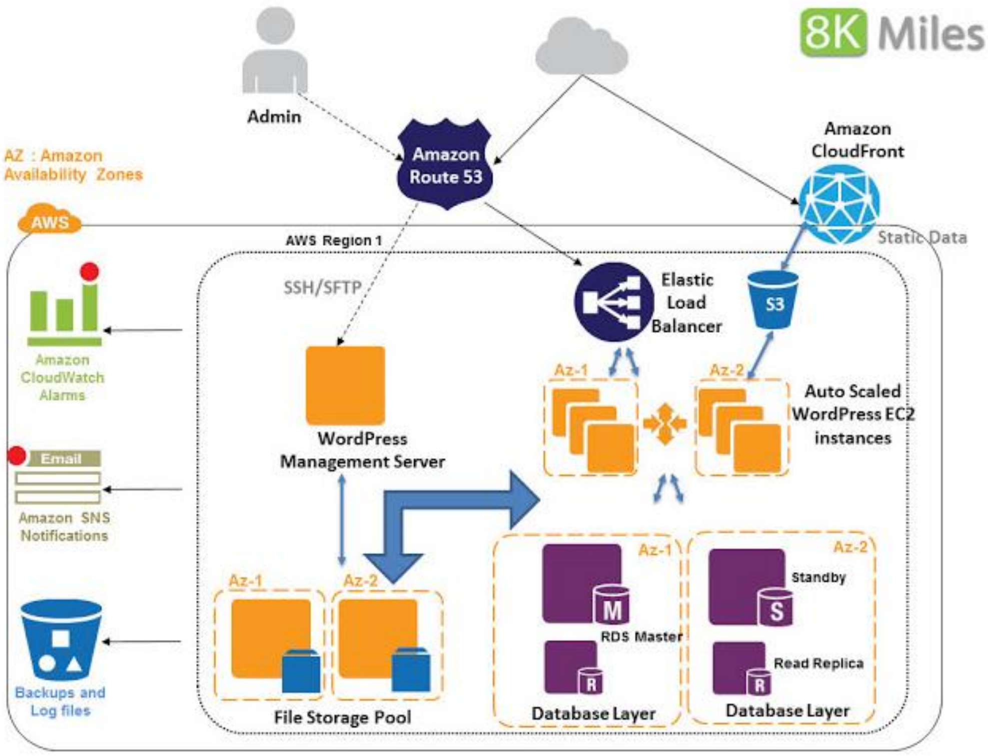


Fig. 5. Integration between the CloudFront and other Amazon services

5.3 MetaCDN

MetaCDN [11] is another content delivery provider that offers two kinds of CDN services: one for static content (e.g., websites) acceleration, and another for live multimedia streaming. Unlike other CDN providers that have their own global distributed system, MetaCDN take advantage of existing storage clouds and compute technology to support its own services. Contrary to cloud providers such as Amazon and Rackspace that offer diverse kind of additional services using their own infrastructure, MetaCDN offers its services by integrating the offerings from several other public cloud providers worldwide, thereby having in excess of 120 edge locations across the world for static content delivery. In case of live streaming, they also have more than 40 edge servers located around the world. As a consequence, MetaCDN is clearly illustrates the power and value of combining the Cloud with the CDN for optimized content delivery over the Internet. Fig 6 presents an overview of MetaCDN architecture [11].

The MetaCDN platform uses connectors to interface with public cloud storage providers such as Amzaon S3, Limelight networks. The connector has the basic sets of operations that are supported by most cloud storage providers. The MetaCDN also have a number of core components responsible for functioning of the service. These include the MetaCDN manager, QoS monitor, Allocator, Database and Load redirector. The allocator selects the optimal service provider. The QoS monitor keeps track of cloud storage performance and the CDN Manager tracks each user's current deployments. The database is used to store vital user and cloud storage mapping information and finally the Load Redirector is responsible for distribute end-user requests to appropriate POP servers. The MetaCDN system also provides user interfaces and APIs to configure system via the web and programmatically.

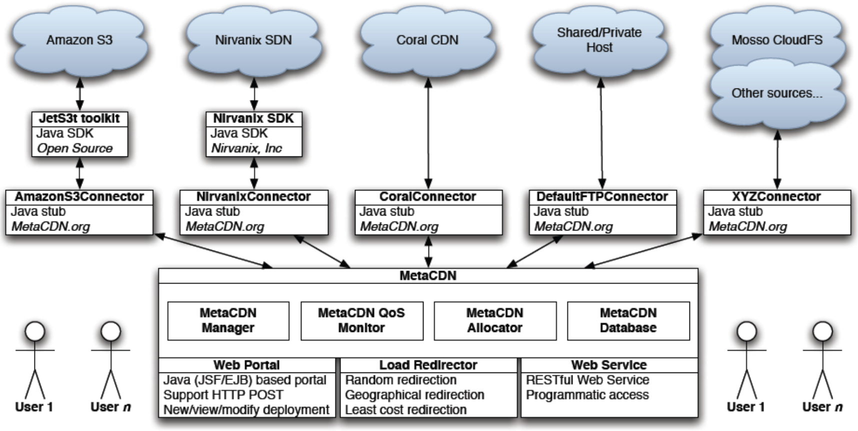


Fig. 6. METACDN Architecture

5.4 Limelight Orchestrate: Limelight Networks

Limelight [13] is one of the biggest CDN providers in the world and offers services such as cloud storage, web acceleration and media delivery. There are some typical products the Limelight offer like "Deep Insight" which gives the customers analytic data which would be helpful for them to make business decisions. The Limelight orchestrate is a content delivery network offered by Limelight networks. This service is one of the world's largest CDN. The Limelight orchestrate service features cloud storage, content control, security, traffic direction and mobile device content delivery. The cloud part of the system take advantage of Limelight networks cloud storage service. Fig 7 presents an overview of the orchestrate service.

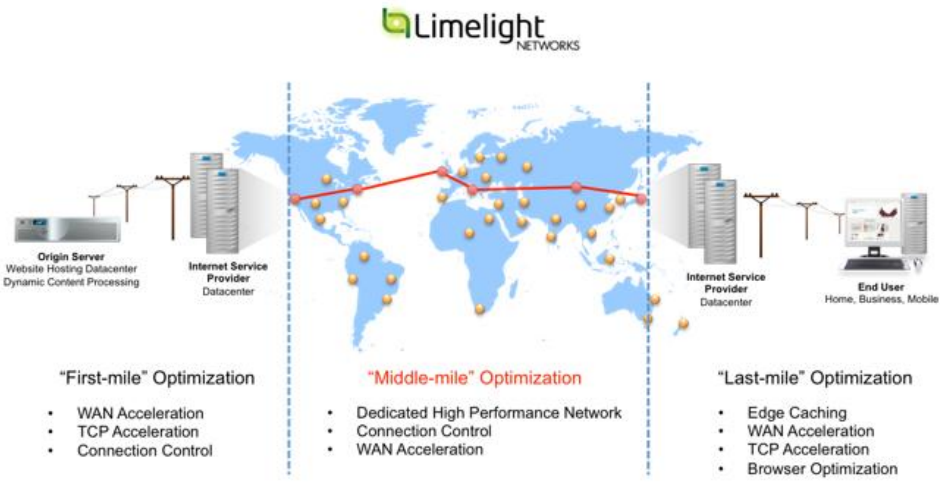


Fig. 7. Limelight Orchestrate – Overview (source: http://www.esg-global.com/lab-reports/limelight-orchestrate-performance/)

5.5 MediaWise Cloud

The MediaWise cloud [14,15] offers a novel cloud orchestration framework where any user can become a CDN provider. As in MetaCDN, the MediaWise cloud leverages multiple cloud providers and offers pay-as-you-go model. The end user can select any public cloud provider simultaneously (e.g., Amazon and Rackspace) based on SLA, price and QoS requirements to leverage services such as compute, storage and content distribution at significantly lower costs. The main highlight of this approach is that a customer is not locked to any particular cloud and CDN provider. Compared to other cloud-based CDNs, another major highlight of the MediaWise cloud is that it supports dynamic content delivery to enable collaborative activities such as collaborative content creation, indexing, storage and retrieval [42].

Fig. 8 shows the reference architecture of MediaWise cloud. As can be seen in this figure, the MediaWise cloud consists of a several components. These include: content orchestrator, hybrid clouds, content access portal and the content management portal. Using the content management portal, the users (content producers) can add, delete or update content on any public clouds (e.g., Amazon and Rackspace). This content is then available to the end users via the content access portal. As mentioned previously, the MediaWise Cloud supports dynamic content creation and delivery. Using this functionality, several users using the MediaWise clouds create multimedia content together via the content access/management portal. This dynamic content can also be annotated using keywords for efficient indexing, search and retrieval. As soon as the request the content is generated from a user(s), it is forwarded it to the MediaWise Cloud content orchestrator (MCCO) [43].

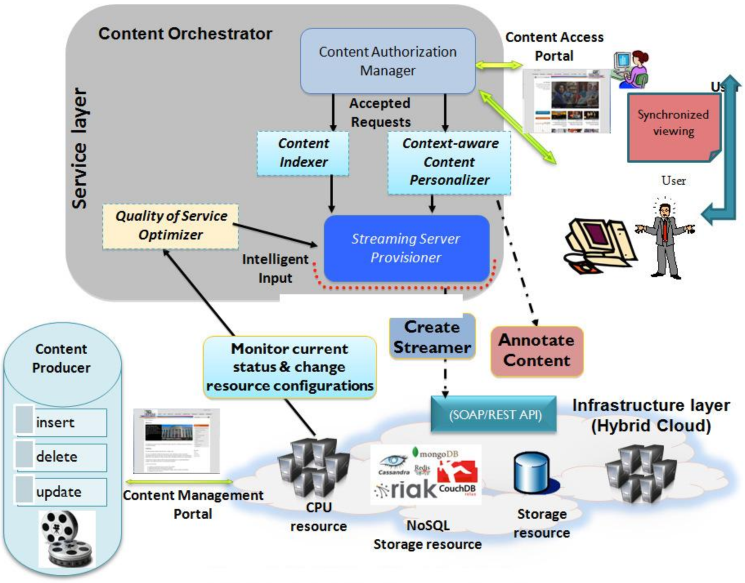


Fig. 8. The MediaWise Cloud architecture [14]

MCCO is the heart of MediaWise cloud. It monitors hybrid clouds and provide mechanisms for QoS-aware cloud selection, scheduling and admission control. For example, as soon as the use request comes from the end user (via the content access portal) for content processing and delivery, the MCCO decides which virtual machine (VM) to provision out of several VMs running on several public clouds. This decision is based on the type of request and the QoS status of the VM on a particular public cloud. Hence, the MediaWise cloud offers QoS-based content placement, delivery as well as compute functionality that is critical in matching end-user SLAs.

5.6 Codeen

Codeen is an academic CDN test-bed developed at Princeton university (http://codeen.cs.princeton.edu/). It is primarily used to support services delivered the Planet Lab project, a global research networks that supports developments of new network services. Codeen has many proxy nodes distributed at various planet lab node locations. The proxy perform the role of POPs and request redirectors. A number of related projects that use the Codeen CDN include web-based content distribution service, name lookup, synchronisation tools, activity monitoring and visualisation tools. A Codeen user sets their cache to a nearby high bandwidth plant lab node location. Request to the codeen node at the location is directed to the most appropriate member of the planet lab system that has a cached copy of the file. This file is forwarded to the client. However, this system lacks support for dynamic content distribution.

5.7 COMODIN

COMODIN (COoperative Media On-Demand on the InterNet) is an academic CDN providing streaming media service on current Internet infrastructure [45]. COMODIN enables a collaborative experience while streaming media content via the Internet. For examples, a group of users can coordinate the state of a media file (e.g. play, pause etc). COMODIN follows a two layer architecture comprising of a base plane and a distributed set of playback components. The system employs IP-multicast to stream data across multiple clients. This academic CDN focuses more on content control collaboratively rather than content creation or distribution.

5.8 CoDaaS:An Experimental Cloud-Centric Content Delivery Platform for User-Generated Contents

CoDaaS [47] is a cloud-centric content delivery system focused on distributing user-generated content in the most economical fashion while respecting the Quality-of-Service (QoS) requirements. It enables on-demand virtual content delivery to a targeted set of users. The system is built of hybrid cloud environments. Fig 9 presents an architecture overview of the CoDaaS system.

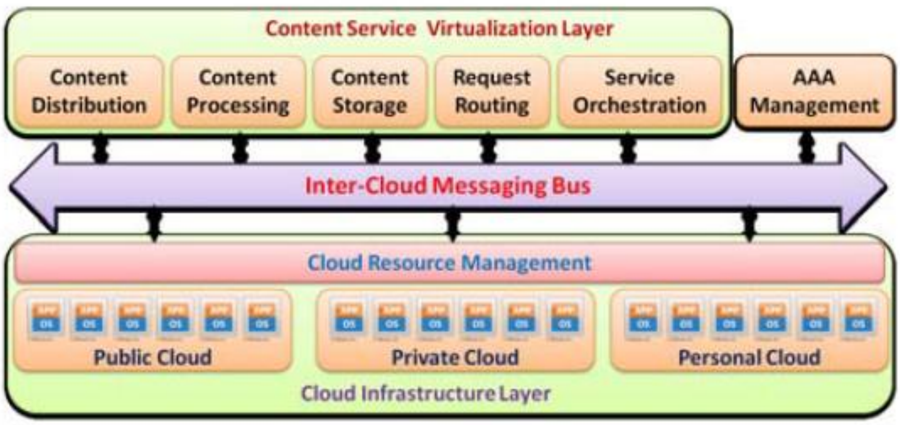


Fig. 9. CoDaaS system architecture

The CoDaaS architecture consists of three layers namely the cloud layer, the content service virtualizations layer and the security layer. The cloud layer comprises the hybrid cloud from which resources are used to develop a content distribution overlay. It also has a set of media rendering engines, managerial and service orchestration function. The content service visualization does the operation of content distribution, processing, storage and routing. This layer performs the function of a typical CDN. The security module is responsible to ensure authorized, authenticated and accountable access to resources across hybrid clouds. Finally, the inter-cloud messaging bus is employed to integrate all participating components into an integrated media application.

6 Analysis of CCDNs current state-of-the-art

As mentioned in the previous section, most commercial CCDNs follow an identical architecture. In the previous section, we presented some of the most popular commercial CCDN architectures and a few academic CDNs. In this section, we will present a twofold comprehensive analysis of commercial CCDNs based on services offered and research dimensions. The services analysis is based on the following characteristics 1) target audience; 2) services; 3) technology and characteristics. The research dimensions used for analysis presented in section 3 are 1) Hybrid cloud support; 2) Content creation and management; 3) Content personalization; 4) Quality of Service; 5) Indexing; 6) Cloud Selection; 7) Content Type. For academic CCDNs we only focus on the research dimensions.

Table 2. Commercial CDNs service analysis

CDN Provider Name	**Services**	**Customers**	**Technology**	**Characteristics**
Akamai [5]	Delivering static, dynamic content and streaming media.	A variety of corporations, such as Rackspace and HP	1) Distributing a large number of surrogate servers 2) Utilizing DNS system 3) BGP (Border Gateway Protocol)	1) Solving the problem of flash crowd 2) The commercial CDN leader 3) Superior load balance system.
Rackspace Cloud Files [9]	1) CDN 2) API for programmatic access 3) File storage	A variety of corporations such as Metro Trains Melbourne, Kogan Australia Private users for file storage and distributions.	1) Powered by Open Stack	1) Global presence using Akamai CDN 2) Large file support 3) Support for dynamic content using API 4) High Performance

CDN Provider Name	Services	Customers	Technology	Characteristics
Amazon CloudFront [10]	1) A variety of content delivery 2) API support 3) Wildcard CNAME support 4) Private content storage and manaement	Various commercial entities such as IMDb, Sega etc.	Related Amazon Cloud computing technology (AWS)	1) High performance 2) Support for dynamic content using low TTL 3) Cloud Storage (Amazon S3) 4) Management Console
Accellion (www.accellion.com)	1) On-demand content transfer solution for exchanging content safely 2) Sending large attachments (gigabyte-sized) 3) Online desktop	Industries, such as Media production, healthcare, consumer goods	1) SeOS (SmartEdge Operating System) 2) SFTA (Secure File Transfer Appliances)	1) Handling large sized files effectively 2) Backup and Recovery Solutions
EdgeStream (http://www2.edgestream.com/es/)	1) IPTV streaming. 2) Video on-demand.	Network providers, Telco's, CDNs, ISPs, content owners and so on	1) Congestion Tunnel Through 2) Continuous Route Optimization	1) High quality of video streams
CloudFlare (https://www.cloudflare.com/features-cdn)	1) Web site content acceleration 2) Web site analytics 3) CDN to support web acceleration	Numerous commercial entities including eHarmony, CISCO, QuickSilver and GOV.UK This service is offered as both free and as pay-as-you-go.	1) 28 data centres around the world 2) CloudFlare optimizer to reduce page service latency 3) Security 4) Analytics for web page content serving .	1) High performance 2) Works with static and dynamic web content 3) Interoperates with other CDNs
MetaCDN [11]	1) Multi CDN 2) Live content streaming 3) Video on demand and video encoding	Many organisation and universities including Sony, The University of Melbourne and Harvard Business School.	1) CDN with 120+ POPs 2) Based on public cloud platforms 3) Uses API connectors to interface with	1) Fast 2) Reliable

CDN Provider Name	Services	Customers	Technology	Characteristics
			public cloud providers	
Mirror Image (http://www.mirror-image.com/)	1) application logic 2) Content Delivery 3) Streaming Media Delivery (video on demand, multi screen, live streaming)	Creative, Open System, and SiteRock to name a few.	1) Patented global dynamic delivery network 2) Edge computing to run application closer to end-users 3) Device optimisations	1) Automatic elastic Scalability 2) Low latency 3) Worldwide coverage 4) Strong SLAs
MaxCDN (http://www.maxcdn.com)	1) Web Content delivery solutions 2) API support for total automation 3) Real time analytics	1,000 customers, 200 large websites including Garmin, Kodak, The Washington times.	1) HTTP Caching 2) Automatic provisioning.	1) Support for static and dynamic content 2) Multi-path network 3) Super POPs 4) Web-based control panel

Table 3. Commercial CDNs research dimensions analysis

CDN Provider Name	Hybrid Cloud Support	Content Creation and Management	Personalisation (User/Device)	QoS	Indexing	Cloud Selection	Content Type
Akamai [5]	No	Partial content management.	No/Yes	Yes	Title and Keyword	Not applicable	Web (static and dynamic) and Media
LimeLight Networks [13]	No	Publish content using web interface	No/Yes	Yes	Title and Keyword	Not applicable	Web (static and dynamic) and Media
Rackspace Cloud Files [9]	No	Mainly content storage and distribution using Akamai	No	Yes. Using Akamai	No	Not applicable	Cloud store for any type of content

CDN Provider Name	Hybrid Cloud Support	Content Creation and Management	Personalisation (User/Device)	QoS	Indexing	Cloud Selection	Content Type
Amazon CloudFront [10]	No	Publish content using web interface and APIs	No/Yes	Yes	Title and Keyword	Not applicable	Web (static and dynamic) and Media
Accellion (www.accellion.com)	No	Mainly file sharing and mobile collaboration	No/yes	No	No	Not applicable	Cloud store for any type of content
EdgeStream (http://www2.edgestream.com/es/)	No	Support to publish content by providers	No/Yes (edge stream enabled device)	Yes	Title and Keyword	Not applicable	Video streaming
CloudFlare (https://www.cloudflare.com/features-cdn)	No	No	No/yes	Yes	No	Not applicable	Web site content acceleration
MetaCDN [11]	Yes	Interface to upload content (cannot orchestrate new content from existing content)	No/Yes	Yes	Title and Keyword	Yes.	Web (static) and various content media types (video, audio, images etc)
Mirror Image (http://www.mirror-image.com/	No	Dynamic content publishing support	No/Yes	Yes	Title and Keyword	Not applicable	Mostly video (live streaming, video on demand)
MaxCDN (http://www.maxcdn.com)	No	Only serve web pages (static and dynamic)	No/Yes	No	No	Not applicable	Web site acceleration
MediWise Cloud [14, 15]	Yes	Yes. Group of users can create content from existing content.	Yes/Yes	Yes	Title, keyword and video context across	Yes	Video, audio and images.

CDN Provider Name	Hybrid Cloud Support	Content Creation and Management	Personalisation (User/Device)	QoS	Indexing	Cloud Selection	Content Type
					clouds		
CODEEN (http://codeen.cs.princeton.edu/)	No	Specifically used content storage and distribution	No/No	No	No	Not applicable	Any content
COMODIN [45]	Yes	Partially. Group of users can control media collaboratively	No/No	No	No	No	Video, audio and images.
CoDaaS [47]	Yes	Yes. Support for user-generated content.	No/Yes	Yes	Name	No	Video, audio and images.

7 Conclusion

Cloud-based CDNs have gained significant importance due to the wide-spread availability and adoption of cloud computing platforms. The integration of Cloud and CDN has mutual benefits allowing content to be efficiently and effectively distributed in the Internet using a pay-as-you-go model promoting the content-as-a-service model. We identified the key challenges and research dimensions that need to be addressed in the cloud-based CDN space. We have presented a state-of-the-art survey on existing commercial and academic cloud CDN solutions. Finally, we provided a comprehensive analysis of commercial and academic cloud CDNs against the service they offer and the research dimensions identified in this paper.

Our findings show that current cloud CDN providers are mostly based on one cloud platform and lacks support for the emerging form of content distribution namely dynamic user-generated content. Since, the solutions are based on a single cloud providers, the services lack consideration for cost models when taking advantage of cloud content storage spaning multiple cloud providers. Further, most commercial and few academic solutions do not support personalisation at user level. We believe, the future of CCDN will be based around the need to support user created content and ability to support hybrid cloud platforms and addressing the challenges such as QoS, SLA, costing introduced by hybrid clouds.

8 Reference

1. Wang. L, Pai. V and Peterson. L, “The Effectiveness of Request Redirection on CDN Robustness”, web.cs.wpi.edu/~rek/DCS/D04/CDN_Redirection.ppt, accessed: 10/09/2014

2. George Pallis and Athena Vakali. 2006. Insight and perspectives for content delivery networks. Commun. ACM 49, 1 (January 2006), 101-106
3. Al-Mukaddim Khan Pathan and Rajkumar Buyya, “A Taxonomy and Survey of Content Delivery Networks”, Technical Report, GRIDS-TR-2007-4, Grid Computing and Distributed Systems Laboratory, The University of Melbourne, Australia, Feb. 12, 2007.
4. Li Ling; Ma Xiaozhen; Huang Yulan, "CDN cloud: A novel scheme for combining CDN and cloud computing," Measurement, Information and Control (ICMIC), 2013 International Conference on , vol.01, no., pp.687,690, 16-18 Aug. 2013
5. Akamai Technologies, Inc., www.akamai.com, 2014
6. A case for the Accountable Cloud http://aws.amazon.com/cn/cloudfront/
7. K. Krauter, R. Buyya, et al., "A Taxonomy and Survey of Grid Resource Management Systems for Distributed Computing," Software: Practice and Experience (SPE), vol. 32, pp. 135-164, Feburary 2002.
8. Microsoft Azure Cloud Services: http://www.windows.azure.com. Accessed: 10/09/2014
9. Rackspace Cloud Files. http://www.rackspace.com.au/cloud/files, Accessed: 8/06/2014
10. Amazon CloudFront. http://aws.amazon.com/cloudfront/, Accessed: 10/09/2014
11. Broberg. J, Buyya. R, Tari. Z, “MetaCDN: Harnessing ‘Storage Clouds’ for high performance content delivery”, Journal of Network and Computer Applications, Volume 32, Issue 5, September 2009, Pages 1012-1022, ISSN 1084-8045, http://dx.doi.org/10.1016/j.jnca.2009.03.004.
12. NoSQL, http://nosql-database.org/, Accessed: 8/06/2014
13. Limelight Orchestrate CDN. http://www.limelight.com/services/orchestrate-content-delivery.html, Accssed: 8/06/2014
14. D. Georgakopoulos, R. Ranjan, K. Mitra, X. Zhou, *“MediaWise - Designing a Smart Media Cloud”*, in Proceedings of the International Conference on Advances in Cloud Computing (ACC 2012), Banglore, India, July 26-28. [Online] http://arxiv.org/ftp/arxiv/papers/1206/1206.1943.pdf
15. R. Ranjan, K. Mitra, D. Georgakopoulos, *“MediaWise Cloud Content Orchestrator”*, Journal of Internet Services and Applications, Springer
16. P. Mell and T. Grance, "The NIST definition of cloud computing (draft)," NIST special publication, vol. 800, p. 145, 2011.
17. A. Letaifa, A. Haji, M. Jebalia, and S. Tabbane, "State of the Art and Research Challenges of new services architecture technologies: Virtualization, SOA and Cloud Computing," International Journal of Grid and Distributed Computing, vol. 3, 2010.
18. C. Gong, J. Liu, Q. Zhang, H. Chen, and Z. Gong, "The characteristics of cloud computing," in Parallel Processing Workshops (ICPPW), 2010 39th International Conference on, 2010, pp. 275-279.
19. S. Zhang, S. Zhang, X. Chen, and X. Huo, "Cloud computing research and development trend," in Future Networks, 2010. ICFN'10. Second International Conference on, 2010, pp. 93-97.
20. M. Ahmed, A. S. M. R. Chowdhury, M. Ahmed, and M. M. H. Rafee, "An Advanced Survey on Cloud Computing and State-of-the-art Research Issues," International Journal of Computer Science Issues(IJCSI), vol. 9, 2012.
21. L. Atzori, F. Granelli, and A. Pescape, "A network-oriented survey and open issues in cloud computing," 2011.
22. Cisco, “Cisco Visual Networking Index: Forecast and Methodology, 2013 – 2018. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.html. Accessed: 10/09/2014
23. IDC, “Data Growth, Business Opportunities, and IT Imperatives”. Available: http://www.emc.com/leadership/digital-universe/2014iview/executive-summary.htm. Accessed: 10/09/2014

24. Akamai, “The Importance of Delivering A Great Online Video Experience”. Available: http://www.akamai.com/dl/reports/jupiter_onlinevideoexp.pdf. Accessed: 10/09/2014
25. IneoQuest, “The Case for Leveraging the Cloud for Video Service Assurance”. Available: http://www.ineoquest.com/wp-content/uploads/2013/10/Whitepaper_Cloud_Services.pdf. Accessed: 10/09/2014
26. BigData & CDN, Available: http://www.slideshare.net/pavlobaron/bigdata-cdnoop2011-pavlo-baron, Accessed: 03/07/2014
27. Hao Yin; Xuening Liu; Geyong Min; Chuang Lin, "Content delivery networks: a bridge between emerging applications and future IP networks," Network, IEEE , vol.24, no.4, pp.52,56, July-August 2010
28. Wang L, Kunze M, Tao J, Laszewski G (2011) “Towards building a cloud for scientific applications”. Adv Eng Softw 42(9):714–722
29. Wang L, Fu C (2010), “Research advances in modern cyber infrastructure”. New GenerComput 28(2):111–112
30. Mastin. P; “Is the Cloud a CDN Killer”, Available: http://cloudcomputing.sys-con.com/node/2628667. Accessed: 03/07/2014
31. Compton. K; “Marching towards cloud CDN” Available: http://blogs.cisco.com/sp/marching-towards-cloud-cdn/. Accessed: 03/07/2014
32. G. Peng, “CDN: Content Distribution Network,” Technical Report TR-125, Experimental Computer Systems Lab, Department of Computer Science, State University of New York, Stony Brook, NY, 2003. http://citeseer.ist.psu.edu/peng03cdn.html
33. A. Vakali, and G. Pallis, “Content Delivery Networks: Status and Trends,” IEEE Internet Computing, IEEE Computer Society, pp. 68-74, November-December 2003.
34. J. Dilley, B. Maggs, J. Parikh, H. Prokop, R. Sitaraman, and B. Weihl, “Globally Distributed Content Delivery,” IEEE Internet Computing, pp. 50-58, September/October 2002.
35. H. T. Kung, and C. H. Wu, “Content Networks: Taxonomy and New Approaches,” The Internet as a Large-Scale Complex System, (Kihong Park and Walter Willinger eds.), Oxford University Press, 2002.
36. Yonggang Wen; Xiaoqing Zhu; Rodrigues, J.J.P.C.; Chang Wen Chen, "Cloud Mobile Media: Reflections and Outlook," Multimedia, IEEE Transactions on , vol.16, no.4, pp.885,902, June 2014
37. A. Shaikh, R. Tewari, and M. Agrawal, “On the Effectiveness of DNS-Based Server Selection,” In Proceedings of IEEE INFOCOM, Anchorage, AK, USA, pp. 1801-1810, April 2001.
38. Z. M. Mao, C. D. Cranor, F. Boughs, M. Rabinovich, O. Spatscheck, and J. Wang, “A Precise and Efficient Evaluation of the Proximity between Web Clients and their Local DNS Servers,” In Proceedings of the USENIX 2002 Annual Technical Conference, Monterey, CA, USA, pp. 229-242, June 2002.
39. M. Armbrust et al., “A View of Cloud Computing,” Communications of the ACM Magazine, Vol. 53, No. 4, pp. 50-58, DOI=10.1145/1721654.1721672, ACM Press, 2010.
40. D. A. Patterson, “Technical perspective: The Data Center Is The Computer,” Communications of the ACM Magazine, Vol. 51, No. 1, pp. 105-105, ACM Press, 2008.
41. L. Hui, "Realistic Workload Modeling and Its Performance Impacts in Large-Scale eScience Grids," IEEE Transactions on Parallel and Distributed Systems, vol. 21, pp. 480-493, 06/15 2010.
42. C. Wang, R. Ranjan, X. Zhou, K. Mitra, S. Saha, M. Meng, D. Georgakopopulos, L. Wang, and P. Thew, “A Cloud-based Collaborative Video Story Authoring and Sharing Platform”, CSI Journal of Computing**,** 1(3), 66-76.
43. R. Ranjan, K. Mitra, S. Saha, D. Georgakopoulos, A. Zaslavsky, “Do-it-Yourself Content Delivery Network Orchestrator“, in Proceedings of the 13th International Conference on Web Information System Engineering (WISE 2012), Paphos, Cyprus, pp. 789-791, Nov 2012.

44. Zhang. X; Du. H; Chen. J-Q; Lin. Y; Zeng. L-J, "Ensure Data Security in Cloud Storage," Network Computing and Information Security (NCIS), 2011 International Conference on , vol.1, no., pp.284,287, 14-15 May 2011
45. Russo. W, Mastroianni. C, Palau. C. E, Fortino. G, "CDN-Supported Collaborative Media Streaming Control", IEEE MultiMedia, vol.14, no. 2, pp. 60-71, April-June 2007
46. Chen. F; Guo, K.; Lin, J.; La Porta, T., "Intra-cloud lightning: Building CDNs in the cloud," INFOCOM, 2012 Proceedings IEEE , vol., no., pp.433,441, 25-30 March 2012
47. Jin. Y; Wen. Y; Shi. G; Wang. G; Vasilakos, AV., "CoDaaS: An experimental cloud-centric content delivery platform for user-generated contents," Computing, Networking and Communications (ICNC), 2012 International Conference on , vol., no., pp.934,938, Jan. 30 2012-Feb. 2 2012
48. Li. H; Zhong. L; Liu. J; Li. B; Xu. B, "Cost-Effective Partial Migration of VoD Services to Content Clouds," Cloud Computing (CLOUD), 2011 IEEE International Conference on , vol., no., pp.203,210, 4-9 July 2011
49. Lizhe Wang, Dan Chen, Jiaqi Zhao, Jie Tao: Resource management of distributed virtual machines. IJAHUC 10(2): 96-111 (2012)
50. Lizhe Wang, Wei Jie: Towards supporting multiple virtual private computing environments on computational Grids. Advances in Engineering Software 40(4): 239-245 (2009)
51. Lizhe Wang, Gregor von Laszewski, Andrew J. Younge, Xi He, Marcel Kunze, Jie Tao, Cheng Fu: Cloud Computing: a Perspective Study. New Generation Comput. 28(2): 137-146 (2010)
52. Lizhe Wang, Dan Chen, Yangyang Hu, Yan Ma, Jian Wang: Towards enabling Cyberinfrastructure as a Service in Clouds. Computers & Electrical Engineering 39(1): 3-14 (2013)