

Creating a ‘Cloud Storage’ Mashup for High Performance, Low Cost Content Delivery

James Broberg¹, Rajkumar Buyya¹, and Zahir Tari²

¹ Department of Computer Science and Software Engineering,
The University of Melbourne, Australia

² Department of Computer Science and Information Technology,
RMIT University, Australia

Abstract. Many ‘Cloud Storage’ providers have launched in the last two years, providing internet accessible data storage and delivery in several continents that is backed by rigorous Service Level Agreements (SLAs), guaranteeing specific performance and uptime targets. The facilities offered by these providers is leveraged by developers via provider-specific Web Service APIs. For content creators, these providers have emerged as a genuine alternative to dedicated Content Delivery Networks (CDNs) for global file storage and delivery, as they are significantly cheaper, have comparable performance and no ongoing contract obligations. As a result, the idea of utilising Storage Clouds as a ‘poor mans’ CDN is very enticing. However, many of these ‘Cloud Storage’ providers are merely basic storage services, and do not offer the capabilities of a fully-featured CDN such as intelligent replication, failover, load redirection and load balancing. Furthermore, they can be difficult to use for non-developers, as each service is best utilised via unique web services or programmer APIs. In this paper we describe the design, architecture, implementation and user-experience of MetaCDN, a system that integrates these ‘Cloud Storage’ providers into a unified CDN service that provides high performance, low cost, geographically distributed content storage and delivery for content creators, and is managed by an easy to use web portal.

1 Introduction

Content creators, ranging from large media companies to smaller, independent start-ups have a need to store and distribute large files (such as audio and video files and rich documents) cost effectively, but with both a global reach and good performance for end-users (consumers) of these files. Traditional CDNs such as Akamai [1] can be too expensive for all but the largest enterprise customers [2]. Most major CDN providers do not publish prices but are anecdotally 2–15 times more expensive, and require 1–2 year commitments¹. ‘Cloud storage’ providers such as Amazon S3 and Nirvanix SDN are an appealing alternative, as they provide internet accessible data storage and delivery services in several continents

¹ Information obtained from <http://www.cdnpricing.com>, part of a popular website and blog for CDN and streaming media professionals run by StreamingMedia.com.

that are backed by rigorous Service Level Agreements (SLAs), guaranteeing specific performance and uptime targets. They offer *utility* pricing (only pay for what you use), and have no ongoing commitments or obligations. As such, a content creator can choose to harness these services only when required for peak load [3].

These emerging services have reduced the cost of content storage and delivery by several orders of magnitude, but they can be difficult to use for non-developers, as each service is best utilised via unique web services or programmer API's, and have their own unique quirks. Many websites have utilised individual Storage Clouds to deliver some or all of their content [3], most notably the New York Times [4] and SmugMug [5], however there is no general purpose, reusable framework to interact with multiple Storage Cloud providers and leverage their services as a unified CDN. Furthermore, a customer may need coverage in more locations than offered by a single provider. To address these issues, in Section 2 we introduce MetaCDN, a system which utilises numerous storage providers in order to create an overlay network that can be used as a high performance, reliable and redundant geographically distributed CDN. MetaCDN makes it easy to harness the performance and coverage footprint of multiple 'Cloud Storage' providers, removes single vendor lock-in, and allows content creators to agilely manage their deployments by expanding and contracting them as needed.

In this paper we focus on the design, architecture, implementation and user-experience of the MetaCDN system. Interested readers are directed to our previous work [6] which gives an extensive background on the 'Cloud Storage' providers used by MetaCDN (listed in Table 1), and demonstrates that they provide sufficient performance (i.e. predictable and sufficient response time and throughput) that is consistent with previous performance studies of dedicated content delivery networks [7,8].

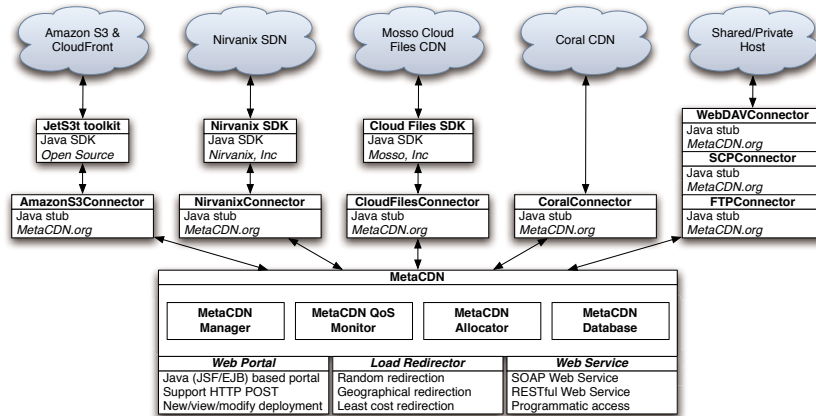


Fig. 1. MetaCDN Architecture

Table 1. Cloud Features (< 2TB for Nirvanix, < 10TB for Amazon, < 5TB for Mosso)

Feature	Nirvanix SDN	Amazon S3	Mosso Cloud Files	Coral CDN
SLA	99.9	99-99.9	*	None
Max. File Size	256GB	5GB	5GB	50MB
US PoP	Yes	Yes	Yes	Yes
EU PoP	Yes	Yes	Yes ²	Yes
Asia PoP	Yes	Yes ³	Yes ²	Yes
Australasia PoP	No	No	Yes ²	Yes
Per File ACL	Yes	Yes	Yes	No
Automatic Replication	Yes	Yes ³	Yes ²	Yes
Developer API	Yes	Yes	Yes	No
Bittorrent Support	No	Yes	No	No
Sideloading Support	Yes	No	No	No
Incoming data (\$/GB)	0.18	0.10	0.00	0.00
Outgoing data (\$/GB)	0.18	0.17(US/EU), 0.21(HK) ³ , 0.22(JP) ³	0.22	0.00
Storage (\$/GB)	0.25	0.15(US), 0.18(EU)	0.15	0.00
Requests (\$/1,000 PUT)	0.00	0.01	0.02	0.00
Requests (\$/1,000 GET)	0.00	0.01	0.00	0.00

2 The MetaCDN System

The aim of the MetaCDN system is to build a low cost, high performance CDN that harnesses the power of ‘Storage Clouds’, and is presented to users as a cohesive, unified interface. In this section we discuss the design, architecture, implementation and user-experience of the MetaCDN system.

2.1 Overall Design and Architecture of the System

The MetaCDN service (shown in Figure 1) is made available to users as a web portal, allowing users to harnessing the storage, performance capabilities and locality of multiple Cloud Storage providers, whilst hiding the complexity involved with interacting with these different entities. The web portal is most suited for small or ad-hoc deployments. A SOAP Web Service has been developed for MetaCDN that is useful for users with more complex and frequently changing content delivery needs. A RESTful Web Service is also under development, that will provide a lightweight Cloud Storage solution for mashup developers that is trivial to utilise. The web portal was developed using Java Enterprise and Java Server Faces (JSF) technologies, with a MySQL back-end to store user accounts, deployments, and the capabilities and pricing of service providers. The MetaCDN system integrates with its’ upstream providers via *connectors*, which are discussed further in Section 2.2.

The web portal acts as the entry point to the system and also functions as an application-level load balancer for end-users that wish to download content

² When used with Limelight’s CDN service.

³ When used with Amazon’s CloudFront CDN service.

that has been deployed by MetaCDN. In order to utilise the MetaCDN system effectively, content must be deployed and managed via the portal or the Web Service, as MetaCDN is unaware of content uploaded directly to participating providers. Currently, the MetaCDN portal (and backing MySQL database) is deployed at a single location (Melbourne, Australia) but in the near future we intend to deploy MetaCDN portals (and replicated backing databases) in all major continents to improve responsiveness and locality for users of MetaCDN, and consumers of the content deployed by the system. This aspect is discussed further in Section 2.5.

The MetaCDN system offers a number of functions via the web portal interface⁴, including:

1. The creation of an account in the MetaCDN system, where a user registers their details, as well as credentials for any service providers (listed in Table 1) they wish to utilise.
2. Intelligent deployment of content based on geographical regions of the user’s choice, their storage and transfer budget, or specific quality of service parameters (described in Section 2.3).
3. Viewing, modifying or deleting existing content deployment.
4. Viewing the physical location of deployed content replicas as a Google Maps Geolocation mashup (described in Section 2.4).

2.2 Integrating ‘Cloud Storage’ Providers

The MetaCDN system integrates with each storage provider via a *connector* that provides an abstraction to hide the complexity arising from each provider having their own unique Web Service API. An abstract class, *DefaultConnector*, is defined that prescribes the basic functionality that each provider could be expected to support, that must be implemented for all existing and future connectors. These include basic operations like creation, deletion and renaming of files and folders, and more advanced operations like creating Bittorrent deployments, and sideloaded files (replicating a file from a publicly available *origin* URL). If an operation is not supported on a particular service, then the connector for that service should throw a *FeatureNotSupportedException*.

Whilst the intent of the MetaCDN system is to provide its users a consistent, unified interface to disparate ‘Cloud Storage’ systems, there are some important differences in functionality and cost between the various providers (as noted in Table 1). For example, Amazon S3 supports Bittorrent deployment of files, whilst the other providers do not. There are also differences in the largest file size that can be deployed, or whether files can be sideloaded as well as directly uploaded to a given service. MetaCDN users do not need to be aware of these subtle differences, as the content they wish to replicate is intelligently matched to the most appropriate provider that suits their specific requirements.

⁴ A screencast of the web interface is available at <http://www.metacdn.org>

2.3 Content Deployment Options

Users of the MetaCDN web portal are presented with a number of different deployment options for replicating their content. These include:

1. Maximising coverage and performance, where MetaCDN deploys as many replicas as possible to all available locations.
2. Deploying content to specific locations a user nominates, where MetaCDN matches the requested regions with providers that service those areas.
3. Cost optimised deployment, where MetaCDN deploys as many replicas in the locations requested by the user as their storage budget will allow.
4. Quality of Service (QoS) optimised deployment, where MetaCDN deploys to providers that match specific QoS targets that a user specifies, such as average throughput or response time from a particular location, which is tracked by persistent probing from the *MetaCDN QoS monitor*.

Once a user deploys using the options above, they are either returned a set of publicly accessible URLs, pointing to the specific locations of the replica files, or a single MetaCDN URL, <http://www.metacdn.org/FileMapper.jsp?itemid=XX>, where *XX* is a unique hash key associated with the deployed content. This provides a single namespace which may be more convenient for users, and can provide automatic and totally transparent load balancing for end-users. This functionality is described further in Section 2.5.

2.4 Integration of Geo-IP Services and Google Maps

Cloud Storage offerings are already available from providers located across the globe. The principle of cloud computing and storage is that you shouldn't need to care where the processing occurs, or where your data is stored - the services are essentially a black box. However, your software and data are subject to the laws of the nations they are executed and stored in. Cloud storage users could find themselves inadvertently running afoul of the Digital Millennium Copyright Act (DMCA)⁵ or Cryptography Export laws that may not apply to them in their own home nations. As such, it is important for Cloud Storage users to know precisely where their data is stored. Furthermore, this information is crucial for MetaCDN load balancing purposes, so end-users are redirected to the closest replica, to maximise their download speeds and minimise latency. To address this issue, MetaCDN offers its' users the ability to pinpoint exactly where their data is stored via geolocation services and Google Maps integration. When MetaCDN deploys replicas to different Cloud Storage providers, they each return a URL pointing to the location of the replica. MetaCDN then utilises a geolocation service (either free⁶ or commercial⁷) to find the latitude and longitude of where the file is stored. This information is stored in the MetaCDN database, and can be overlaid onto a Google Maps view inside the MetaCDN portal, giving users a birds-eye view of where their data is currently being stored.

⁵ Available at <http://www.copyright.gov/legislation/dmca.pdf>

⁶ Hostip.info is a free community-based project to geolocate IP addresses.

⁷ MaxMind GeoIP is a commercial IP geolocation service.

2.5 Load Balancing via DNS and HTTP Redirection

Load balancing for both MetaCDN users and consumers of the content that the system replicates is achieved in two stages. First, users or consumers of MetaCDN are directed to their closest portal at the DNS resolution stage. Currently, there is a MetaCDN portal running in Australia, and soon there will be portals running in Europe and North America. For MetaCDN consumers, if they are attempting to access a file via a MetaCDN URL, then they are redirected (by a HTTP 302 Found directive) to the most appropriate replica. What constitutes the most appropriate replica depends on the deployer of the content (and the preferences they expressed, described in Section 2.3). This could be the highest performing replica, the geographically closest replica or even the cheapest replica.

3 Conclusion

In this paper we gave an overview of the design, architecture, implementation and user-experience of MetaCDN, a system that integrates ‘Cloud Storage’ providers into an unified CDN service that provides high performance, low cost, geographically distributed content storage and delivery for content creators, and is managed by an easy to use web portal. More information on the ongoing development of MetaCDN can be found at <http://www.metacdn.org>

This work is supported by Australian Research Council (ARC) as part of the Discovery Grant ‘Coordinated and Cooperative Load Sharing between Content Delivery Networks’ (DP0881742, 2008-2010).

References

1. Maggs, B., Technologies, A.: Global internet content delivery. In: First IEEE/ACM International Symposium on Cluster Computing and the Grid, pp. 12–12 (2001)
2. Pathan, M., Buyya, R.: A Taxonomy of CDNs. *Content Delivery Networks*, 33–78 (2008)
3. Elson, J., Howell, J.: Handling Flash Crowds from your Garage. In: USENIX 2008: 2008 USENIX Annual Technical Conference (June 2008)
4. Gottfrid, D.: Self-service, prorated super computing fun! OPEN: All the code that is fit to print (2007), <http://open.nytimes.com/2007/11/01/self-service-prorated-super-computing-fun>
5. MacAskill, D.: Scalability: Set Amazon’s Servers on Fire, Not Yours. In: ETech 2007: O’Reilly Emerging Technology Conference (2007), <http://blogs.smugmug.com/don/files/ETech-SmugMug-Amazon-2007.pdf>
6. Broberg, J., Buyya, R., Tari, Z.: MetaCDN: Harnessing ‘Storage Clouds’ for high performance content delivery. Technical Report GRIDS-TR-2008-10, Grid Computing and Distributed Systems Laboratory, The University of Melbourne (August 2008)
7. Johnson, K., Carr, J., Day, M., Kaashoek, M.: The measured performance of content distribution networks. *Computer Communications* 24(2), 202–206 (2001)
8. Su, A., Choffnes, D., Kuzmanovic, A., Bustamante, F.: Drafting behind Akamai (travelocity-based detouring). *ACM SIGCOMM Computer Communication Review* 36(4), 435–446 (2006)