Computer clouds support not only network-centric computing, but also network-centric content. For

example, we shall see in Chapter 8 that in 2013 Internet video is expected to generate over 18 exabytes

of data per month. The vast amount of data stored on the cloud has to be delivered efficiently to a large

user population.

Content-delivery networks (CDNs) offer fast and reliable content delivery and reduce the communication

bandwidth by caching and replication. A CDN receives the content from an origin server, then replicates

it to its edge cache servers. The content is delivered to an end user from the “closest” edge server.

CDNs are designed to support scalability, increase reliability and performance, and provide better

security. The volume of transactions and data transported by the Internet increases dramatically every

year. Additional resources are necessary to accommodate the additional load placed on communication

and storage systems and to improve the end-user experience. CDNs place additional resources provisioned

to absorb the traffic caused by flash crowds7 and, in general, to provide capacity on demand.

The additional resources are placed strategically throughout the Internet to ensure scalability. The

resources provided by a CDN are replicated, and when one of the replicas fails, the content is available

from another one. The replicas are “close” to the consumers of the content, and this placement reduces

the startup time and the communication bandwidth. A CDN uses two types of server: the origin server

updated by the content provider and replica servers that cache the content and serve as an authoritative

reference for client requests. Security is a critical aspect of the services provided by aCDN; the replicated

content should be protected from the increased risk of cyber fraud and unauthorized access.

A CDN can deliver static content and/or live or on-demand streaming media. Static content refers

to media that can be maintained using traditional caching technologies because changes are infrequent.

Examples of static content are HTML pages, images, documents, software patches, and audio and/or

video files. Live media refers to live events during which the content is delivered in real time from the

encoder to the media server. On-demand delivery of audio and/or video streams, movie files, and music

clips provided to end users is content-encoded and then stored on media servers. Virtually all CDN

providers support static content delivery, whereas live or on-demand streaming media are considerably

more challenging.

The first CDN was set up by Akamai, a company evolved from a Massachusetts Institute of Technology

(MIT) project to optimize network traffic. Akamai has placed some 20,000 servers in 1,000

networks in 71 countries since its inception. In 2009 it controlled some 85% of the market [285].

Akamai mirrors the contents of clients on multiple systems placed strategically through the Internet.

Though the domain name (but not the subdomain) is the same, the IP address of the resource requested

by a user points to an Akamai server rather than the customer’s server. Then the Akamai server is

automatically picked, depending on the type of content and the network location of the end user.

There are several other active commercial CDNs, including EdgeStream, which provides video

streaming, and Limelight Networks, which provides distributed on-demand and live delivery of video,

music, and games. There are several academic CDNs: Coral is a freely available network designed to

mirror Web content, hosted on PlanetLab; Globule is an open-source collaborative CDN developed at

the Vrije Universiteit in Amsterdam.

The communication infrastructure among different CDN components uses a fair number of protocols,

including Network Element Control Protocol (NECP), Web Cache Coordination Protocol (WCCP),

SOCKS, Cache Array Routing Protocol (CARP), Internet Cache Protocol (ICP), Hypertext Caching

Protocol (HTCP), and Cache Digest, described succinctly in [285]. For example, caches exchange ICP

queries and replies to locate the best sites from which to retrieve an object; HTCP is used to discover

HTTP caches, cache data, or manage sets of HTTP caches and monitor cache activity.

There are two strategies for CDN organization. In the so-called overlay, the network core does not

play an active role in content delivery. On the other hand, the network approach requires the routers

and the switches to use dedicated software to identify specific application types and to forward user

requests based on predefined policies.

The first strategy is based exclusively on content replication on multiple caches and redirection based

on proximity to the end user. In the second approach, the network core elements redirect content requests

to local caches or redirect data center incoming traffic to servers optimized for specific content type

access. Some CDNs, including Akamai, use both strategies.

Important design and policy decisions for a CDN are:

• The placement of the edge servers.

• The content selection and delivery.

• The content management.

• The request routing policies.

The placement problem is often solved with suboptimal heuristics using workload patterns and

network topology as inputs. The simplest, but a costly, approach for content selection and delivery is

full-site replication, suitable for static content. The edge servers replicate the entire content of the origin

server. On the other hand, partial-site selection and delivery retrieve the base HTML page from the

origin server and the objects referenced by this page from the edge caches. The objects can be replicated

based on their popularity or on some heuristics.

Content management depends on the caching techniques, cache maintenance, and cache update

policies. CDNs use several strategies to manage the consistency of content at replicas: periodic updates,

updates triggered by the content change, and on-demand updates.

The request routing in a CDN directs users to the closest edge server that can best serve the request;

metrics such as network proximity, client perceived latency, distance, and replica server load are taken

into account in routing a request. Round-robin is a nonadaptive request-routing method that aims to

balance the load; it assumes that all edge servers have similar characteristics and can deliver the content.

Adaptive algorithms perform considerably better but are more complex and require some knowledge

of the current state of the system. The algorithm used by Akamai takes into consideration metrics such

as the load of the edge server, the bandwidth currently available to a replica server, and the reliability.

CDN routing can exploit an organization in which several edge servers are connected to a service

node that’s aware of the load and the information about each edge server connected to it and attempts

to implement a global load-balancing policy. An alternative is DNS-based routing, in which a domain

name has multiple IP addresses associated with it and the service provider’s DNS server returns the IP

addresses of the edge servers holding the replica of the requested object; then the client’s DNS server

chooses one of them.

Another alternative is HTTP redirection; in this case a Web server includes in the HTTP header of

a response to a client the address of another edge server. Finally, IP any casting requires that the same

IP address is assigned to several hosts, and the routing table of a router contains the address of the host

closest to it.

The critical metrics of CDN performance are:

• Cache hit ratio, which is the ratio of the number of cached objects versus total number of objects

requested.

• Reserved bandwidth for the origin server.

• Latency, which is based on the perceived response time by the end users.

• Edge server utilization.

• Reliability, which is based on packet-loss measurements.

CDNs will face considerable challenges in the future due to increased appeal of data streaming and to

the proliferation of mobile devices such as smartphones and tablets. On-demand video streaming requires

enormous bandwidth and storage space as well as powerful servers. CDNs for mobile networks must

be able to dynamically reconfigure the system in response to spatial and temporal demand variations.

The concept of a content service network (CSN) was introduced in [226]. CSNs are overlay networks

built around CDNs to provide an infrastructure service for processing and transcoding.