Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Communications 29 (2006) 551–562

computer communications

www.elsevier.com/locate/comcom

From content distribution networks to content networks — issues and challenges

Thomas Plagemann[a,*], Vera Goebel[a], Andreas Mauthe[b], Laurent Mathy[c], Thierry Turletti[d], Guillaume Urvoy-Keller[e]

[a] *University of Oslo, Department of Informatics, Oslo, Norway*
[b] *Darmstadt University of Technology, Multimedia Communications Lab, Darmstadt, Germany*
[c] *Lancaster University, Distributed Multimedia Systems Research Group, Lancaster, United Kingdom*
[d] *INRIA, Planète Team, France*
[e] *Institut Eurecom, France*

Available online 11 July 2005

Abstract

Due to the technical developments in electronics the amount of digital content is continuously increasing. In order to make digital content respectively multimedia content available to potentially large and geographically distributed consumer populations, Content Distribution Networks (CDNs) are used. The main task of current CDNs is the efficient delivery and increased availability of content to the consumer. This area has been subject to research for several years. Modern CDN solutions aim to additionally automate the CDN management. Furthermore, modern applications do not just perform retrieval or access operations on content, but also create content, modify content, actively place content at appropriate locations of the infrastructure, etc. If these operations are also supported by the distribution infrastructure, we call the infrastructure Content Networks (CN) instead of CDN. In order to solve the major challenges of future CNs, researchers from different communities have to collaborate, based on a common terminology. It is the aim of this paper, to contribute to such a terminology, to summarize the state-of-the-art, and to highlight and discuss some grand challenges for CNs that we have identified. Our conception of these challenges is supported by the answers to a questionnaire we received from many leading European research groups in the field.

Keywords: Multimedia content; Metadata; Distribution overlay; Content management

1. Introduction

Due to the technical developments in electronics, digital representation is becoming more and more dominant compared to analogue representation of multimedia content. For example, new capturing devices like digital video cameras enable professional content producers to directly produce high-quality movies for broadcasting and movie theatres in digital formats e.g. MPEG or DV based video formats. Also, in the private domain the use of digital cameras is constantly increasing. It is obvious that this trend will continue and as a result the amount of digital multimedia content is continuously increasing.

Multimedia content is produced for later use in different contexts, for example in entertainment, training, etc. Usually, content is not used at the same geographic location where it is produced and/or stored. Furthermore, there is in most cases a potentially very large user population. In order to efficiently deliver content to the users, an infrastructure is used that is called Content Distribution Network (CDN). The first large scale type of CDNs that has been used for many years (and probably will still be used in the near future) are broadcast networks for television. However, the role of the Internet as a basic platform for CDNs is becoming more and more important because of two reasons. First, the amount of digital multimedia content is

* Corresponding author.
E-mail addresses: plageman@ifi.uio.no (T. Plagemann), goebel@ifi.uio.no (V. Goebel), andreas.mauthe@kom.tu-darmstadt.de (A. Mauthe), laurent@comp.lancs.ac.uk (L. Mathy), turletti@sophia.inria.fr (T. Turletti), urvoy@eurecom.fr (G. Urvoy-Keller).

0140-3664/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2005.06.006

continuously increasing; and second, the number of end-users that are connected to the Internet with reasonable bandwidth links, e.g. ADSL, is also increasing. Consequently, the usage of Internet based CDNs and by this also their importance will increase.

The usage of CDNs is changing. Currently, for instance, the dominant application is content presentation. The first content presentation applications have been simple video streaming applications. Due to achievements in computer architecture, networking, compression technologies, and synchronization technologies, modern applications are used to present and interact with synchronized multimedia documents, which are composed out of synchronized continuous and discrete media elements. These applications allow users to select media elements as well as to control the application. In other words, the degree of interactions between user and content has increased compared to the first generation of content presentation applications and play-out is just one of many functions that are supported. It is the task of the CDN to provide the user efficient read access as well as navigation and search facilities to the content.

Content consists of two main parts, the encoded media and metadata. Metadata, such as content description but also format and location related metadata provides the means for efficient content retrieval and placement of content in the CDN. Further, it enables the control of the content within a CDN. Metadata can be created manually, but ideally should be extracted from the media and production process automatically. Only if this automatic support can be realized can the vast amount of newly produced content be efficiently handled [29]. Automatic annotation and updating relevant metadata, as well as metadata management are in our view going to become part of the infrastructure. Further, the management of content according to its properties (i.e. its description with metadata and the administration of different copies within the infrastructure) is also becoming part of the content infrastructure and referred to as content management.

So far, we based our discussion on an intuitive understanding of the terms content and CDN. Since there is no generally accepted definition of these terms, we give in the following a precise characterization of these terms:

The term Content Distribution Network (CDN) implies a networked infrastructure that supports the distribution of content. Content in this context consists of encoded data or multimedia data, e.g. video, audio, documents, images, web pages, and metadata, i.e. data about data. Metadata allows identifying, finding and managing the multimedia data, and also facilitates the interpretation of the multimedia data.

Content can be pre-recorded or retrieved from live sources; it can be persistent or transient data within the system. Distribution refers to the active retrieval or the active transmission of information. The infrastructure has to provide communication support and ought to contain mechanisms that facilitate effective delivery or increase availability of content (such as caching, replication, perfecting).

Thus, CDNs should exploit the inherent structural and/or semantic characteristics (i.e. distinct properties) of content as well as access patterns and delivery modes for effective content handling and distribution. Within a CDN the content items are the prime objects of distribution and management, i.e. the entire operations within a CDN centre around content, its distribution and all processes related to it.

In contrast to other infrastructures that can also be used to distribute content, CDNs are specifically designed for the transmission and retrieval of content. Since other systems or infrastructures can be (and in fact are) used for the delivery of content, it is necessary to distinguish and delineate CDN from other systems that are also communicating data. This differentiation should not discriminate against other systems, but it should make clear what the differences are. CDNs also have to be compared to other, non-CDNs that are used for content delivery. If these systems are better or equally efficient for content delivery, the purpose of CDNs has not been fulfilled. It should be noted that the above characterizations covers not only Internet based CDNs, but other infrastructures for the distribution of content (such as Broadcast Networks) as well. For Internet based CDNs, which are the focus of our work, it should be noted that CDNs can go beyond the simple client-server computing model. On top of the transport infrastructure of the Internet additional functionality is implemented, typically as an overlay network. CDNs can take various forms and structures. On the one hand, they can be centrally controlled, hierarchical infrastructures under the administrative domain of a service provider; on the other hand, they can be completely decentralised systems (such as P2P file sharing applications). In between, they can be various forms of interworking and control sharing between different entities forming a CDN. The crucial point about CDNs, however, is that they exploit the inherent content characteristics to provide a better service.

2. Architectural principles and issues in content (distribution) networks

As we have stated earlier, the main concern of current CDNs is the efficient delivery and increased availability of content to the consumer. Therefore, mechanisms like caching, replication, pre-fetching, batching of requests in multicast streams etc. have been developed. The caches and proxies that implement these mechanisms are structured in an overlay network. In the early years of CDN research and development the main focus has been on the delivery mechanism and the management and maintenance of the CDN itself has been done by hand, i.e. explicitly and very often hard coded into the system. This might be suitable for the placement of PCs in the network and their integration into a particular overlay structure as well as for the creation

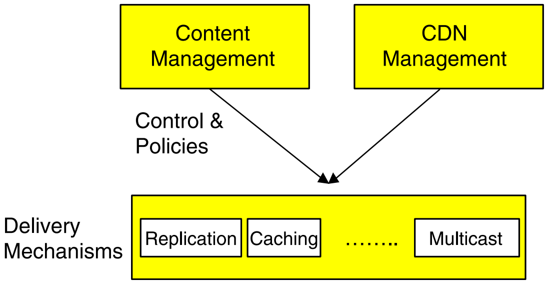

Fig. 1. High-level architecture.

and placement of content replicas. More modern CDN solutions, however, aim to automate the CDN management. One task of CDN management would be to maintain the overlay topology of a CDN with respect to the current network conditions.

We have seen that modern applications do not just perform retrieval or access operations on content but also create content, modify content, actively place content at appropriate locations of the infrastructure, etc. If these operations, which we so-far call content management operations, are also supported by the infrastructure, we call the infrastructure Content Networks (CN) instead of CDN. This is done in order to distinguish the more sophisticated, upcoming systems from those systems focusing on the pure distribution of content.

A major issue in this context is the architecture of such infrastructures. There are various possibilities to design such systems. However, they can be all assessed according to the kind of operations they support and the way content is handled. As illustrated in Fig. 1, we differentiate three classes of operations, i.e. content management operations, CDN management operations, and delivery mechanisms. In order to support the different needs of applications it is necessary to apply the classical principle of systems design in CNs again, i.e. the separation of mechanisms and policies. Delivery mechanisms are controlled and guided by policies that are provided by CDN management and content management. One example for this is the automatic decision when to establish where a new replica, based on meta-data describing the content and meta-data describing user patterns.

3. State-of-the-art

General considerations on CDN design can be found in [5]. First generation CDNs have mostly focused on Web documents, either static or dynamic [8,9]. Rabinovich et al. [6] cover caching and replication techniques (both operational and research solutions) for the specific case of the Web. Research efforts in this domain gave birth to successful companies like Akamai [7].

Second generation CDNs (that have not currently reached the market) deal with Video-on-Demand (VoD) and audio and video streaming. Wu et al. provide in [1] a general overview of the problems related to video streaming over the Internet (at the server and client sides and also in the network). A rich literature exists on scalable video-on-demand algorithms with open-loop, e.g. [2], and close loop algorithms, e.g. [3,4].

In the remainder of this section, we address the following topics that we deem important in the context of CDNs and CNs. First, we review the standardization effort in CDNs and Content Management. This is followed by a discussion of the communication aspects of content distribution. Next, we discuss both P2P research efforts and operational solutions to provide large scale services. Finally, we focus on emerging content management systems.

3.1. CDN standardization

There are two major standardization initiatives that have addressed issues related to content distribution. These are namely the MPEG-21 framework for the management and delivery of content and the IETF CDI (content distribution internetworking) working group.

3.1.1. MPEG-21

The Moving Pictures Expert Group (MPEG) has recognized the need to define a framework that describes how the different elements of content management and delivery fit together in order to provide ‘the big picture’ [51]. This resulted in the specification of the MPEG-21 Multimedia Framework. MPEG-21 is concerned with the entire (fully electronic) workflow of digital multimedia content creation delivery and trading. Its aim is to cover interaction with multimedia content and to provide a framework for the transparent usage of various content types and multimedia resources on multiple devices connected through a wide range of networks.

MPEG-21 has seven key-elements [29]:

- Content Handling and Usage is an interface specification that covers all workflow steps in the content value chain from content creation, over its manipulation, search and storage, to its delivery and re-use.
- Digital Item Declaration is a scheme for declaring Digital Items by a set of standardised abstract terms and concepts; i.e. it specifies the makeup, structure and organisation of essence and content objects called Digital Items.
- Digital Item Identification is a framework for the identification and description of entities regardless of their nature and granularity.
- Intellectual Property Management and Protection (IPMP) deals with IPR management and protection at all involved devices and networks.
- Terminal and Networks deals with the functional interoperability between heterogeneous networks and devices.

- Content Representation specifies how media resources are represented
- Event Reporting defines the metrics necessary to understand performance and event reports in a MPEG-21 system.

At its most basic level, MPEG-21 provides a framework in which one user interacts with another where the object of the interaction is a piece of content (called Digital Item). These interactions include creating content, providing content, archiving content, rating content, enhancing and delivering content, aggregating content, syndicating content, retail selling of content, consuming content, subscribing to content, regulating content, facilitating and regulating transactions that occur from any of the above.

The MPEG-21 standardization has commenced in late 1999, but considering the immense task and large scope of the project it is still at a very early stage. At present much effort is spend on Intellectual Property Rights (IPR) management and protection issues covering all the legal aspects involved in this.

3.1.2. IETF CDI initiative

Within the IETF content distribution issues have been addressed in the IETF CDI WG [45]. This working group represents the IETF's CDN and Content Distribution Internetworking (CDI) initiatives. CDNs and CDIs are intended to serve as platforms for content providers to distribute their content without having to manage an infrastructure. The work concerning CDI has remained in the requirements stage. Its main focus has been on large scale content distribution in a Web context. Issues that have been addressed by the working group include a model for CDI [44], architectural questions [46], distribution requirements [47], CDI scenarios [48], and CDI Authentication Authorisation and Accounting (AAA) requirements [49]. The central aspect in this context is the location, download and usage tracking of content in CDN and CDI.

A CDN provides an infrastructure for publishing content to a large user group. It is an overlay network that runs on top of the Internet using application level protocols and techniques. Caching and server farms are proposed to bring the content close to the user/consumer. This reduces network load and improves the perceived QoS to the user.

In order to increase the reach of a single CDN, commercial and technical internetworking between different providers is proposed. The key technologies for content internetworking are request routing, advertisement, content distribution and accounting procedures between different networks. Each CDN remains a 'black-box' to the peering CDN in a CDI.

Not in the focus of the initiative are content search, rights management and rights protection. Closed and protected content environments (e.g. for business-to-business processes) are also not part of the standardization effort. Further, while support for streamed and live transmission for continuous multimedia data is mentioned in the context of Surrogate servers and their potential role to act as splitters, continuous media support is not sufficiently addressed [50].

The different CDI related documents introduce a number of interesting and relevant aspects in the context of content distribution in an Internet environment but remain at the requirements and conceptual level. Since 2002, no further activities on these topics have been reported.

3.2. Communication systems

The design principles of the Internet were based around a model where the network forwards packets to the specified destination on a best-effort basis. Whilst these founding principles have contributed to the Internet's scalability and resilience to partial failures, they ultimately mean that the Internet is suited to supporting applications without strict timing requirements. A major research challenge over the past 15 years has been looking at ways to deliver media with strict timing restrictions across an infrastructure that is fundamentally offering a best-effort service. More recently, research has focused on developing a distribution infrastructure that not only operates on top of a best-effort infrastructure, but also copes with problems such as network congestion and overloaded servers.

During the 1990's, the research community invested heavily in the development of multimedia servers, and demonstrated that it was vital to examine the design principles of file servers before considering how they could support continuous media. In particular, servers offering real-time guarantees in order to support the timely delivery of continuous media were investigated [95,111]. Whereas ensuring that the continuity of a single continuous media stream is relatively straightforward, supporting multiple users without violating the integrity of other requested media streams necessitated the employment of appropriate admission control mechanisms. There was also a need to improve storage retrieval techniques in order that sufficient amounts of data were obtained from disk subsystems to meet the requirements of the streams currently in service. Continued work in server design has led to the development of high performance servers capable of providing content to a large number of users. However, this was only the first step. Delivery of content to users in remote locations was still a major problem, and led to considerable work in the deployment of a real-time service model for the Internet.

Attempts to solve the delivery problem prompted work on ways to 'layer' reservation and admission control mechanisms on top of IP, to provide several levels of network service. This work was carried out within the Integrated Services (IntServ) and Resource Reservation Protocol (RSVP) working groups of the IETF. The work was later complemented by research into a more lightweight mechanism for service differentiation, led by the

Differentiated Services (DiffServ) working group, and work on Multiprotocol Label Switching (MPLS). These solutions go some way to offering predictable behaviour for Internet users. However, their effectiveness in an end-to-end scenario is still unclear, due to the complexity involved in obtaining services across a number of ISP domains [96], which may require service level agreements to be negotiated between individual ISPs involved in the delivery process. There is also a question mark over the scalability of some of the signalling mechanisms involved in providing end-to-end QoS [96].

Network caching is seen as an orthogonal (and complementary) solution providing end-to-end QoS. Network caching entails storing frequently accessed content closer to users, therefore reducing the distance (number of hops) that must be traversed in order to retrieve the content. This has a number of potential impacts. Firstly, by locating content close to users, the latency and unpredictability of the delay in the Internet can be reduced. Second, by local caching of requested content from remote servers, fewer of these requests have to be sent out across the Internet to be fulfilled. As a consequence, the upstream bandwidth requirements are reduced, resulting in cost savings to the network service provider. Thirdly, as content is serviced locally, the overall load on the origin server is reduced, allowing it to service other requests.

The predominant use of network caching within the Internet has been in providing web-caching services for HTML documents consisting of static images and text. However, over recent years the web has undergone a process of evolution, moving away from what was once a predominantly text based information system to a fully-fledged multimedia information system. Studies have shown that the number of continuous media objects stored on web services is increasing [97]. Despite this, most Internet web caches are either unable to meet the timely demands of continuous media, or simply disregard the material for caching. As a result, research work has started on the design of scalable multimedia cache node architectures [98].

Recent research in multimedia caching has been developing new architectures and mechanisms to support the delivery of high quality live and on-demand streaming media. Work in this field includes SOCCER [99], developed at the Networking Software Research Department at Bell Labs, who have developed an architecture to provide better support for streaming media over the Internet, using segmentation of streaming objects, dynamic caching, and self-organizing cooperative caching techniques. Other work in this area includes Middleman, developed at Cornell University, consisting of a collection of cooperative proxy servers that, as an aggregate, cache video files within a local area network [100]. Research also considers the caching of hierarchical streams [101], caching policies to support multimedia content [102,103], and approaches to caching multimedia information [104–107].

Many of the above-mentioned caching architectures assume, and rely on, the use of multicast communication within the network. However, for various technical and economic reasons, multicast has not been widely deployed in the Internet [108]. This has triggered research to find ways to support, at the application level, functionality not ubiquitously provided in the Internet [109]. This has led to solutions relying on Application-Level Overlay Networks, which represent logical interconnection amongst end-systems. Examples of such overlays are application-level multicasting structures, Peer-to-Peer (P2P) networks and some CDNs. With the exception of content location for asynchronous downloads that lead to techniques based on Distributed Hash Tables, much of the work on overlay networks has been focused towards scalability issues for logical interconnection. Application-level Overlays can be tightly integrated with, and exploit the characteristics of, applications (e.g. data format, communication architecture, etc.)

3.3. Peer-to-Peer systems

P2P file sharing systems are probably the most popular IT content delivery systems at present. In 2001, Napster was the fastest-growing application in the Internet's history [52]. Whereas Napster still contained some central elements, Gnutella is a fully decentralized, unstructured system [53]. Since the advent of these systems a number of proposals using for instance Supernodes [54], [55] or decentralized structured approaches based on hash search and indexing [56–58] have been developed. Further, some systems use hybrid approaches that combine client/server aspects with P2P structured and unstructured concepts [61–63]. Structured P2P networks index content using hash indices. These hash indices represent meta-information to a certain extent. However, the kind of management and content support they provide is fairly limited. For instance range queries cannot so easily be supported by structured approaches. Usually only the search for specific content objects where the identity is known is possible. Further, although there are proposals to provide load-balancing (e.g. within DHT based systems [59,60]) active management of content within the system is commonly not part of the systems.

While P2P networks still need to improve their metadata management, they nonetheless offer the ability to leverage the computation, disk and bandwidth resources of a lot of hosts simultaneously in the Internet. P2P networks are expected to solve two important issues related to content distribution: First, the streaming of content to a large population of clients in the (best-effort) Internet, and second, the large scale distribution of files (e.g. virus patches or OS updates). We hereafter review the main proposals that address these two fundamental issues.

Two main classes of streaming applications can be distinguished: VoD applications and more delay-sensitive live streaming multimedia applications, such as the seminar or television broadcast.

A lot of mechanisms have been proposed for VoD-like applications and they usually leverage multiple sending peers and one receiver [64–67]. Some solutions also propose layered video coding or multiple description coding to handle the asymmetric properties of the access networks (e.g. ADSL or cable modems) [68,69].

Issues related to streaming live multimedia flows have also been addressed in the research community and solutions include single or multiple applicative multicast trees. The single tree approach is the most popular today and aims to reproduce the native IP multicast structure across tunnelled unicast connections between peers. It is used in protocols like SpreadIt [70], PeerCast [71], ESM [72], NICE [73] and Zigzag [74]. The main differences between the different schemes lie in the target goals (e.g. reliable vs. unreliable multicast transmission), the algorithms used to create/maintain the tree and to manage peer arrivals and departures. The second approach which is based on multiple multicast trees allows to achieve load balancing and obtains a better resiliency to node failures and packet loss by using appropriate data encodings. Such mechanisms include Splitstream [76], CoopNet [77] and P2PCast [78]. Bullet [79] is an original hybrid scheme combining a standard single-tree structure and a mesh made of random connections (orthogonal to the tree) used to send the bulk of data among peers which are far away in the tree hierarchy.

The second important issue addressed by P2P networks in the context of content distribution is the replication of files on a large set of peers. The most important challenge here is to devise distributed algorithms that, on one hand, enforce the cooperation among (selfish) hosts and, on the other hand, are robust to early departure of peers and to flash crowd arrivals of new peers. BitTorrent [80,24] is currently the most popular P2P replication application. The entry point to a BitTorrent session (replication of a single file) is a web server that redirects the client to the so-called tracker that keeps track of all the peers active in the session. The tracker itself is not involved in the file distribution. The latter function is based on the swarming technique, the file being cut into equal size chunks (typical size is 256 KB). Two algorithms control the exchange of chunks among peers. First, a peer selection algorithm that enables peers to consistently look for the fastest servers and second, a chunk selection algorithm that allows to maintain the diversity of chunks in the system. Overall, BitTorrent appears to be extremely efficient for large sessions (thousands of peers) with flash crowds [24]. Qui et al. provide in [81] an in-depth mathematical study of the BitTorrent algorithms, while [82] provides a more general study on large scale replication strategies (using chains, tree, and parallel trees). Slurpie [31] proposes more complex algorithms to achieve goals similar to BitTorrent. Preliminary results are promising but the actual performance of Slurpie (large number of clients and/or flash crowd) is unknown. FastReplica [83] offers a solution to large scale replication in the context where all peers are controlled by a single entity, as the case of the surrogate servers of a CDN provider.

3.4. Content management systems

At present, content is mainly managed within the application domain. In general, two types of content management systems (CMS) can be distinguished, i.e. systems managing Web pages and documents [35], and professional CMS that are used in content production (e.g. film and video production), handling and delivery [29]. Whereas the former is mainly concerned with managing information in a certain presentation context, the latter supports not only content administration but also all processes and workflows related to content production, handling and transmission. What these systems have in common is the architectural approach. In both cases, an architecture in which the different content parts are managed largely independently is the core of the system.

Usually, the actual media (i.e. video, audio, images, web-content, etc.) is managed separately from the metadata within special file or storage systems. For the media relatively large, storage intensive files have to be handled. The communication requirements can vary between a few Kb/s to hundreds of Mb/s (e.g. MPEG-4 based video [36]). The metadata is typically a structured representation of the data related to, respectively describing, content. Metadata can be represented in data models (e.g. the BBC's SMEF^{TM} [37]) or description schemes (e.g. MPEG-7 [38], Dublin Core [39]). Very often the metadata is encoded using XML [40]; for the professional domain a combination of the SMPTE Metadata Dictionary and the Key-Length-Value protocol is proposed [41]. Sometimes media and metadata is handled together for instance within MPEG-2 Multiple Program Transport Streams [29] or in content files (e.g. MXF [42] or BWF [43]).

For large CMS that span an entire organization a distributed architecture is being adopted. The different system components are connected via various different networks (LAN, WAN, Fibre Channel, SDI, etc.).

Further, autonomous CMS within the same organization are also linked to allow access to content under the control of another organizational domain (e.g. the archive CMS to editorial office CMS). At this level the integration ranges from simple message exchange over the use of APIs to a full integration using component based application development. Flexibility and scalability are ensured using concepts such as service groups and broker-manager models [29].

What is currently missing are automatic content migration according to information-life-cycle concepts and self-organizing CMS. The former is referring to automatically moving content to the most appropriate location in a distributed CMS infrastructure. For instance, news content could be automatically transferred to the archive CMS after a certain time period. Self-organizing

CMS is referring to networked systems that allow the flexible addition of systems and components without the need of central control. Ideally, new CMS would join the CMS network without any manual configuration.

In today's CMS, content management is entirely taking place within a CMS middleware layer or at the application level. The communication sub-systems are solely used to transfer data. IT networks such as LAN or the Internet are assumed to be purely best effort not providing any specific support for the transmission of content. Apart from this specific networks for the transmission of video and audio (e.g. SDI, SDTI, satellites, etc.) or dedicated networks (e.g. within storage area networks) are deployed as part of the CMS infrastructure.

4. A snapshot of ongoing research activities

In order to achieve a better understanding of how the research community is approaching the problem to advance the previously described state-of-the-art, we have distributed in 2004 a questionnaire to all members of E-Next. E-Next is a Network-of-Excellence, which is funded by the 6th Framework Programme of the EU and which is formed by 42 European research groups in networking and distributed systems. The particular goal of the questionnaire was to get a concise overview on ongoing research activities in the area and its recent results, like research projects, PhD works, publications etc. Additionally, we are interested to learn about the basic assumptions, like in which environments shall the solutions be working, which tools are used for CN research, and what are the main challenges for future CNs. The fact that 14 research teams from E-Next answered the questionnaire and helped us to identify 22 projects and 21 PhD theses in E-Next itself demonstrates that CNs are regarded as an important research field. In this section, we give a brief summary of the assumptions, applications and tools, while the next section is integrating most of the responses and the main challenges and discuss them in more depth.

In the research projects that have been identified, there is a clear consensus about the networking infrastructure that should be addressed: all assume an IPv4 based network and some do not exclude a later transition to IPv6, all consider best-effort networks and some combine them with DiffServ, and mainly large scale systems are targeted. The applications that are supposed to use the CNs that are developed in the different projects comprise the classical Video-on-Demand as well as News-on-Demand with high user interactivity, Distributed Content Management System for Professional use, Medical Digital Video Library, Media-on-Demand for Education, 3D Virtual Worlds for a large number of users that interact through multicast streams. The tools that are used for these research projects comprise various programming tools and simulation tools, of which ns-2 [94] is clearly the favourite tool for most researchers. However, many researchers have realized that simulation is just an early step in the development and PlanetLab [88] as a test-bed is regarded also as very important. Furthermore, topology generators, like Brite [89], workload generators, like MediSyn [90] and ProWGen [91], and content creation tools like Helix DNA Producer Command Line Application [92] and camtasia [93] are applied.

5. Challenges

There are many challenges that need to be addressed in future CN research. In this section, we state five of those challenges that we deem to be major ones:

1. How to handle unexpected resource demand and network conditions?
2. How to design and develop a research CN?
3. How many CDNs do we need to fulfill the requirements of different applications?
4. How to take advantage of application and content semantics?
5. How to deploy future large scale services?

5.1. Measurement effort

CNs rely on CDN services to distribute and manage content within the network. Fundamentally, the emergence of CNs does not necessarily require any new CDN mechanisms apart from the necessary customization required for each application so that CDN elements (surrogate servers, proxies) become aware of the content and application semantic. A challenge lies in the ability of the CDN elements to react fast enough to changes in network conditions to avoid any service disruption. The more interactive the service is, the more stressed the CDN elements are. This situation is somewhat new compared to first generation CDNs where the foremost purpose was to boost up access to Web content. The main challenge for the first generation CDNs was to offer a significantly better service compared to the legacy Internet best-effort service. In the case of CNs, the objective is not primarily to offer a better service than the legacy best effort service but to offer a consistent service throughout users sessions' lifetime. With respect to this constraint one of the biggest challenges for CNs is the ability of CNs to continuously monitor network conditions. Among other aspects, monitoring network conditions means assessing available bandwidth and links capacity on different network segments (from central to surrogate servers or from surrogate servers to clients). While techniques and tools have been proposed by the network measurement community (see [30] for a survey on bandwidth measurement techniques), we are still far from a satisfying solution to meet multimedia content requirements. Directing users to the correct surrogate server also requires specific measurement techniques. Extreme cases of network load changes are flash-crowd and DDOS attacks. The counter measures to be taken against these events might depend on the service that is provided. High added value services (e.g. interactive video-on-demand) might require specific provisioning while intermediate added value services (e.g. personalized news-on-demand) might rely on users' cooperation through some peer-to-peer mechanism to temporally offload the CDN [31–33].

Network monitoring if also important in the context of the wireless world, because it converges to a whole IP solution. Wireless users might have high download bandwidth (e.g. in 3G networks), but may suffer highly fluctuating network conditions due to mobility or fading effects.

5.2. Towards a research CN

The need for a CN research platform stems mainly from two aspects of CN research. There is the need to collect usage statistics to inform system design and engineering decisions. Here the challenge is to 'look into the future', by deploying new services to a user population and on a platform reflecting a real world deployment, rather than restrict testing to the labs. To be effective, such a CN should not only be very stable, but also operate within a realistic scenario.

On the other hand, researchers will also want to be able to quickly deploy and test new services and mechanisms, as part of a normal system research cycle. In this context, stability cannot be guaranteed and the drive for quick results would mean simulating user requests (using the real patterns measured on a stable CN) to generate real CN traffic. This points to the need for an integrated research framework that allows researchers to quickly assemble CN systems from existing components implementing parts of the system they do not directly focus on.

Furthermore, because of the performance issues encountered in CN research, researchers should be given the opportunity to experiment with low-level operating systems mechanisms (e.g. disk scheduling, network protocols, etc.). However, access to low-level system mechanisms can have dramatic impact on the overall system performance and stability and should therefore only be granted if proper isolation of concurrent experiments can be guaranteed.

It should be noted that world-wide testbeds already exist (e.g. PlanetLab [88]) which can be used to support part of the CN research. However, it is not clear that these testbeds are sufficient as their operating performance may not reflect the real operating circumstances one would find in a CN: for instance, due to other experiment being carried out on PlanetLab, node loads and offered traffic may be skewed away from reality, while the sharing model currently used in PlanetLab seriously limits the disk space available to each experiment. It may therefore be necessary to create a separate, but complementary, CN testbed with access control in order to limit concurrent experiments to a reasonable level. In this context, operating systems that provide virtual machines on top of the hardware (e.g. the Xenoserver [110]) can effectively be used to multiplex several virtual CN networks (including a stable, statistics collecting one) onto the same hardware platform, while providing the required isolation properties. Virtualization, coupled with appropriate time-multiplexing access control to the platform (i.e. booking system that limits the number of concurrent experiments) and advanced portal support (for the repository) of code and research results, seem to be a promising approach to providing a research CDN platform.

Whichever approach is taken to built such a CN research platform, common challenges include: access and sharing of interesting content to reflect normal user operations, the very difficult task (due to the distributed nature of the testbed itself over the Internet) of being able to measure and recreate conditions of an experiment, the possibility to quickly put together (i.e. integrate) the required functionality of a CN that incorporates specific new mechanisms, etc.

Finally, we must stress that with virtualization being introduced to PlanetLab in the near future [88], the major difference between the proposed research CN platform and PlanetLab will be in the access control model, whereby access to the CN platform will be based on keeping the number of concurrent experiments to a reasonable level at all times (while providing long-term access fairness), as opposed to providing a fair share of existing resources to an unlimited number of experiments. However, despite the differences in access control, both platforms would still complement each other with, for example, clients on PlanetLab generating requests for content on the research CN. This complementarity of the platforms is a key to large-scale CN research.

5.3. How many CDNs?

If one considers CDNs as the set of functionalities and active devices within the network and CNs as an upper layer that aims at transforming application and users needs and information into metadata, then two natural questions arise from this global picture:

1. To which extent can the application and users requirements be transformed into metadata that allow CDNs to operate without any further control of the application? The next section will elaborate in more details on this question
2. How many CDNs/CNs architectures do we need?

Concerning the latter question, two extreme answers are:

- One CDN architecture per application: An argument for this case is that each application has its own constraints (e.g. QoS constraints) that require specific mechanisms

(caching web documents is fundamentally different from caching video content)

- One CDN architecture shared by all applications: An argument for this case follows from Section 4, where it appears that current and future multimedia applications needs can be handled using a limited and well identified set of functionalities (caching, pre-fetching, etc.). We can thus expect the emergence of a generic CDN that all applications could share. A typical example along this line is presented in [34] where it is shown that an NVoD architecture, originally designed for the large scale distribution of long video movies (e.g. 90 min), was still efficiently performing when distributing small clips (e.g. 5 or 7 min).

As usually in such cases, we can expect the correct answer to lye in between the extremes. Accordingly, we can expect that a few CDN architectures will emerge. At first sight, we could expect these architectures to correspond to a few application profiles, depending on the level of interactivity, the business model, or the nature of the service, e.g. streaming vs. stored content distribution, etc. However, we believe that the exact number of CDN architectures is more fundamentally a function of the extent to which the CDN operations can exploit metadata independently of the application control. We elaborate more on this point in the next section.

While the number of distinct CDN architectures is an important issue, we can expect, from an operational point of view, that the number of operated CDNs will be large. As a consequence, there will be a need for inter-working between CDNs. This issue has already been addressed in Section 3.1. However, as pointed out by [9], such initiative has little chance to succeed due to the complexity of defining a peering service among CDN providers, as was similarly observed with the deployment of DiffServ or IntServ among ISPs. A more promising approach might be the leasing of on-demand resources, where a CDN provider would lease some CPU/storage resources, e.g. to cope with a sudden increase of audience during some special event like the Olympic Games.

5.4. Exploiting Application and Content Semantics

Current CDN approaches are either implicitly exploiting application and content semantics (e.g. caching strategies for Web content), use a rudimentary sub-set (e.g. P2P file sharing applications) only or largely ignore it. However, the goal of a CDN is to appropriately exploit them to optimize the delivery of content within a CN. The challenge in this context is how to find the right level of abstraction and balance of application knowledge within the communication sub-system.

The content characteristics that can be exploited at this level are material and location related metadata but also content related metadata. Material related metadata refer to the kind of multimedia data (e.g. video, audio, Web pages) and multimedia data formats. This allows providing optimized communication support for a specific multimedia data type (e.g. streaming of continuous multimedia data with bit-rate adaptation in the case of layered encoded video). Location metadata gives information about where a copy of the content can be found in the system. This refers to content in different formats and can include various versions of the same content object. Location information captured in Distributed Hash Tables (DHTs) is an example of how basic location information can be exploited in a CDN. The challenge is to use more sophisticated concepts that also include information about alternative versions, etc.

Whereas material and content related metadata are already being exploited to a certain extent by CDN, content related metadata is usually not considered. Content related metadata refers to all descriptive information. Relevant in the context of CN is information that allows identifying and finding content, and IPR related data. Together with context information this can be used to improve content placement and delivery within a CN. For instance, the knowledge about a certain event (e.g. international football game) can be exploited to place all the related content in close vicinity to the contestants' domain, i.e. it can be used to do (pro-) active content management within a CN. Together with the application semantic this should enable better and more efficient utilization of resources and/or an improved QoS for the service user.

The fact that IPR information can be used to optimize CNs is a valuable side effect of its main task to protect content against unauthorized, illegal usage. It has to be included at the level where content (and not just data) is being distributed. Thus, it has to be part of a CN.

Apart from the question how and to which extent the different metadata types should be exploited the issue also is how to represent it in the system. In general, metadata can be placed together with the content or may be managed independently linked via unique identifiers. Metadata as part of the multimedia data can for instance be found in different file and stream formats. Further, meaningful file names and ID can contain metadata information. Even a traditional TV signal caries metadata in the blanking inverval. However, the provisions for metadata as part of the media are not sufficient for rich metadata information sets. Further, it restricts the usage of this information together with the media. Therefore, it makes sense to keep metadata at other locations to use it for system-wide operations and not only within a local context. Traditionally, databases are used for the management of structured data. If and how they can be used within a CN infrastructure has to be researched. Further, other alternatives for the representation of metadata (e.g. information represented as part of the infrastructure) also have to be investigated to find the most optimal way for utilizing metadata within CN.

5.5. Large scale issue

Currently, Akamai, the largest existing CDN, operates more than 10,000 boxes. Akamai is offering a service for Web objects and also, at a smaller scale, an audio/video streaming service (see [87] for details). However, the type of applications that CDNs/CNs will support in the future will transform the current business model where content producers pay content providers to maximize the impact of their content into a business model where end users will also pay to receive (high quality) content. The main issue will thus be to deliver this high quality content in a scalable manner (scalability is necessary to maintain low operational costs). With respect to the scale of the problem, the following questions need to be altogether addressed:

- How to handle heterogeneity of receivers in terms of network fan-in, fan-out, and resources available at the terminal (e.g. CPU, screen size).
- How to handle scalability for broadcast events to a large audience, scalable congestion control. [85,86] have demonstrated the complexity of this latter issue.
- How to support low-latency streaming of live multimedia flows (e.g. minimize the number of hops, select not overloaded peers, etc.). The research platform described in Section 5.2 will of course play an important role in the study of scalability issues and solutions. For instance, it will be capable of supporting large-scale experiments that integrate, and study this integration of, various research results into a single system (e.g. hybrid CDN using techniques from P2P), within a realistic usage scenario.

6. Conclusions

CDNs are since several years subject to research. This fact might give the naïve observer the impression that current and future research results in this area can only be of incremental nature. However, this impression is wrong, as we have shown in this paper. First of all, a common terminology is missing, especially for researchers coming from different communities, like networking, VoD, multimedia database systems, and content management. By describing the high-level structure of CDNs and CNs, their various tasks and relationships, this paper aims to contribute to a common terminology. There are many important recent research results and research activities in the area and a brief overview on them is given in Sections 3 and 4. However, there are many grand challenges that have not yet been solved and different research groups have only recently started to address a few of them. Our conception of the open research challenges has been validated by the questionnaire that has been answered by many leading European research groups in the field. One of the common threads in all the research challenges we have addressed is the necessity to move from single mechanisms development towards real-live deployment. Individual mechanisms have to be tested, evaluated, and deployed in the context of a CN and not just in an independent simulation or emulation environment. Operational CNs have to be measured and semantics of content and applications have to be exploited to improve efficiency of CN maintenance. In order to be able to approach these research challenges in settings that are as realistic as possible, the proper test-beds are needed. Planetlab is a first step in this direction, but more specific CN support is necessary to enable realistic large-scale testing and deployment of CNs.

We hope that this paper contributes to establishing a better understanding of the issues and challenges in CN and encourages researchers to address those problems.

Acknowledgements

This work has been supported by E-Next, a Network-of-Excellence in the 6th Framework of the European Commission. We would like to thank especially all E-Next members that filled out the questionnaire and returned all the valuable information to us.

References

[1] D. Wu, Y.T. Hou, W. Zhu, Y. Zhang, J.M. Peha, Streaming video over the Internet: approaches and directions, IEEE Transactions on Circuits and Systems for Video Technology 11 (3) (2001) 282–300.

[2] A. Hu, Video-on-demand broadcasting protocols: a comprehensive study, Proceedings of IEEE INFOCOM, Anchorage, Alaska, USA 1 (2001) 508–517.

[3] S. Ramesh, I. Rhee, K. Guo, Multicast with cache (Mcache): an adaptive zero-delay video-on-demand service, Proceedings of IEEE INFOCOM, Anchorage, Alaska, USA (2001).

[4] L. Gao, D. Towsley, Supplying instantaneous video-on-demand services using controlled multicast, Proceedings of IEEE Multimedia Computing Systems (1999) 117–121.

[5] D. Verma, Content Distribution Networks, Wiley, New York, 2002.

[6] M. Rabinovich, O. Spatscheck, Web Caching and Replication, Addison-Wesley, Reading, MA, 2002.

[7] J. Dilley, B. Maggs, J. Parikh, H. Prokop, R. Sitaraman, B. Weihl, Globally distributed content delivery, IEEE Internet Computing 6 (5) (2002) 50–55.

[8] A. Vakali, G. Pallis, Content delivery networks: status and trends, IEEE Internet Computing 7 (6) (2003) 68–74.

[9] I. Lazar, W. Terrill, Exploring content delivery networking, IT Professional 3 (4) (2001) 47–49.

[24] M. Izal, G. Urvoy-Keller, E.W. Biersack, P.A. Felber, A. Al Hamra, L. Garces-Erice, Dissecting BitTorrent: Five Months in a Torrent's Lifetime, Fifth Annual Passive and Active Measurement Workshop (PAM'2004), Antibes Juan-les-Pins, France (2004).

[29] A. Mauthe, P. Thomas, in: Wiley (Ed.), Professional Content Management System: Handling Digital Media Assets (2004).

[30] R. Prasad, C. Dovrolis, M. Murray, K. Claffy, Bandwidth estimation: metrics, measurement techniques, and tools, IEEE Network 17 (6) (2003) 27–35.

[31] R. Sherwood, R. Braud, B. Bhattacharjee, Slurpie: a cooperative bulk data transfer protocol, IEEE Infocom (2004).

[32] A. Keromytis, V. Misra, D. Rubenstein, SOS: an architecture for mitigating DDoS attacks, IEEE Journal on Selected Areas in Communications (JSAC), Special Issue on Service Overlay Networks 22 (1) (2004).

[33] A. Stavrou, D. Rubenstein, S. Sahu, A lightweight, robust, P2P system to handle flash crowds, IEEE Journal on Selected Areas in Communications (JSAC), Special Issue on Service Overlay Networks 22 (1) (2004).

[34] E.W. Biersack, A. Al Hamra, G. Urvoy-Keller, D. Choi, Cost-optimal Dimensionig of a Large Scale Video on Demand Server, Quality of Future Internet Services COST Action 263 Final Report Lecture Notes in Computer Science, vol. 2856, Springer-Verlag, Berlin, 2003.

[35] O. Christ, Eine Architektur für das Content-Management, PhD Thesis, University St. Gallen, 2001.

[36] F. Pereira, T. Ebrahimi (Eds.), The MPEG-4 Book IMSC Press Multimedia Series, Prentice Hall, Englewood Cliffs, NJ, 2002.

[37] The Standard Media Exchange Framework (SMEF™), British Broadcasting Cooperation, http://www.bbc.co.uk/guidelines/smef/, 2002

[38] B. Manjuhabath, P. Salembier, T. Sikora (Eds.), MPEG-7: Multimedia Content Description Interface, Wiley, New York, 2002.

[39] About the Dublin Core Metadata Initiative, DCMI—Dublin Core Metadata Initiative, http://dublincore.org, 2004.

[40] K. Ahmed, D. Ayers, M. Birbeck, J. Cousins, D. Dodds, J. Lubell, M. Nic, D. Rivers-Moore, A. Watt, R. Worden, A. Wrightson, Professional XML Meta Data, Wrox Press, UK, 2001.

[41] SMPTE Metadata Dictionary, Society of Motion Picture and Television Engineers, RP210.2 (including RP210.1), Merged Version, post rail publication of RP210.2, http://www.smpte-ra.or/mdd/RP210v2-1merged020507b.xsl, White Plains, NY, 2001.

[42] Media eXchange Format (MXF), 7 Parts: File Formats, OPI, DV, GC, Format, GC D10, Mapping, GC SDTI-CP Mapping, Version 10, SMPTE/Pro-MPEG Forum, http://www.g-fors.com/mxf.htm, 2002.

[43] Specification of the Broadcast Wave Format, A Format for Audio Date Files in Broadcasting, Supplement 1: MPEG Audio, European Broadcasting Union, EBU Technical Document 3285, 1997.

[44] M. Day, B. Cain, G. Tomlinson, P. Rzewski, A Model for Content Internetworking (CDI), Internet RFC 3466, <http://www.faqs.org/rfcs/rfc3466.html>, February 2003.

[45] IETF Content Distribution Internetworking (CDI) Working Group, http://www.ietf.org/html.charters/cdi-charter.html.

[46] M. Green, B. Cain, G. Tomlinson, S. Thomas, P. Rzewski, Content Internetworking Architectural Overview, Internet Draft ⟨draft-ietf-cdi-architecture-00.txt⟩, February 2002.

[47] L. Amini, S. Thomas, O. Spatscheck, Distribution Requirements for Content Internetworking, Internet Draft, ⟨http://www.ietf/org/proceedings/03Mar/draft-ietf-draft-ietf-cdi-distribution-reqs-00.txt⟩, December 2002.

[48] M. Day, D. Gilletti, P. Rzewski, Content Internetworking (CDI) Scenarios, RFC 3570, ⟨http://ops.ietf.org/lists/cdn/cdn.2003/msg00034.html⟩, July 2003.

[49] D. Gilletti, R. Nair, J. Scharber, J. Guha, Content Internetworking (CDI) Authentication, Authorization, and Accounting Requirements, Internet Draft, ⟨http://www.ietf.org/Proceedings/02nov/I-D/draft-ietf-cdi-aaa-reqs-00.txt⟩, June 2002.

[50] Mauthe, A. Content Management and Delivery-Related Technology Areas, Technical Report MPG 03–04. Lancaster University, 2002.

[51] Coding of Moving Pictures and Audio, N4801: MPEG-21 Overview version 4, ISO/IEC JTC1/SC29/WG11, Fairfax, USA, 2002.

[52] C. Shirky, Listening to Napster, in Peer-to-Peer—Harnessing the Power of Distruptive Technologie, O'Reilly, New York, 2001.

[53] The Gnutella Protocol Specification v0.4, http://www9.limewire.com/developer/gnutella_protocol_0.pdf, 2000.

[54] FastTrack. http://www.fasttrack.nu.

[55] Kazaa. http://www.kazaa.com.

[56] Overnet. http://www.overnet.com.

[57] P. Maymounkov, D. Maziéres, Rateless codes and big downloads, Proceedings of Second International Workshop on Peer-to-Peer Systems (IPTPS03) (2003).

[58] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, H. Balakrishnan, Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, ACM Press, New York, 2001. pp. 149–160.

[59] J. Byers, J. Considine, M. Mitzenmacher, Simple load balancing for distributed hash tables, Proceedings of the Second International Workshop on Peer-to-Peer Systems (IPTPS03) (2003).

[60] A. Rao, K. Lakshminarayanan, S. Surana, R. Karp, I. Stoica, Load balancing in structured P2P systems, Proceedings of the Second International Workshop on Peer-to-Peer Systems (IPTPS03) (2003).

[61] V. Darlagiannis, A. Mauthe, R. Steinmetz, Overlay Design Mechanisms for Heterogeneous, Large Scale Dynamic P2P Systems in Journal of Network System Management, 12(3) 2004.

[62] B. Traversat, A. Arora, M. Abdelaziz, M. Duigou, C. Haywood, J.-C. Hugly, E. Pouyoul, B. Yeager, Project JXTA 2.0 Super Peer Virtual Network, http://www.jxta.org/project/www/docs/JXTA2.0protocols1.pdf, 2003.

[63] J. Kubiatowicz, D. Bindel, Y. Chen, S. Czerwinski, P. Eaton, D. Geels, R. Gummadi, S. Rhea, H. Weatherspoon, B. Zhao, Oceanstore: an architecture for global-scale persistent storage, Proceedings of the Ninth International Conference on Architectural Support for Programming Languages and Operating Systems (2000).

[64] D. Xu, M. Hefeeda, S. Hambrusch, B. Bhargava, On peer-to-peer media streaming, Purdue Computer Science Technical Report, 2002.

[65] Cui, K. Nahrstedt, Layered Peer-to-Peer Streaming in Proceedings of International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'03), 2003.

[66] X. Jiang, Y. Dong, D. Xu, B. Bhargava, Gnustream: a P2P media streaming system prototype, Proceedings of the International Conference on Multimedia and Expo (ICME) 2 (2003).

[67] M. Hefeeda, A. Habib, B. Botev, D. Xu, B. Bhargava, PROMISE: Peer-to-Peer Media Streaming Using CollectCast in Proceedings of ACM Multimedia, Berkeley, CA, 2003.

[68] R. Rejaie, A. Ortega, PALS: Peer to Peer Adaptive Layered Streaming in Proceedings of the International Workshop on Network and Operating Systems Support for Digital Audio and Video Monterey, California, 2003.

[69] M. Zink, A. Mauthe, P2P Streaming using Multiple Description Coded Video in Proceedings of the 30th EUROMICRO 2004, Rennes, France, 2004.

[70] H. Deshpande, M. Bawa, H. Garcia-Molina, Streaming Live Media over a Peer-to-peer Network, Technical Report 2001-31, Stanford University, 2001..

[71] H. Deshpande, M. Bawa, H. Garcia-Molina, Efficient Topology-Aware Overlay Network in Proceedings of the 1st Workshop on Hot Topics in Networks, Princeton, New Jersey, USA, 2002.

[72] Y. Chu, A. Ganjam, T. Seng, S.G. Rao, K. Sripanidkulchai, J. Zhan, H. Zhang, Early Experience with an Internet Broadcast System Based on Overlay Multicast, Technical Report CMU-CS-03-214, Carnegie Mellon University, 2003.

[73] S. Banerjee, B. Bhattacharjee, C. Kommareddy, Scalable Application Layer Multicast in ACM SIGCOMM, Pittsburgh, PA, USA, 2002.

[74] D.A. Tran, K.A. Hua, T.T. Do, Zigzag: An Efficient Peer-to-peer Scheme for Media Streaming in Proceedings of IEEE INFOCOM, 2003.

[76] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, A. Singh, SplitStream: High-Bandwidth Multicast in Cooperative Environments in ACM SOSP, Bolton Landing, NY, USA, 2003.

[77] V.N. Padmanabhan, H.J. Wang, P.A. Chou, K. Sripanidkulchai, Distributing Streaming Media Content Using Cooperative Networking in ACM/IEEE NOSSDAV, Miami, FL, USA, 2002.

[78] A. Nicolosi, S. Annapureddy, P2PCast: A Peer-to-Peer Multicast Scheme for Streaming Data in IRIS Student Workshop, MIT, Cambridge, MA, 2003.

[79] D. Kostic, A. Rodriguez, J. Albrecht, A. Vahdat, Bullet: High Bandwidth Data Dissemination Using an Overlay Mesh in ACM SOSP, 2003.

[80] Cohen, B. Incentives to Build Robustness in BitTorrent, http://bitconjurer.org/BitTorrent/bittorrentecon.pdf, 2003.

[81] D. Qiu, R. Srikant, Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks in SIGCOMM, Portland, OR, USA, 2004.

[82] E.W. Biersack, P. Rodriguez, P. Felber, Performance Analysis of Peer-to-Peer Networks for File Distribution in QofIS'04, Barcelona, Spain, 2004.

[83] L. Cherkasova, J. Lee, FastReplica: E_cient large file distribution within content delivery networks, Proceedings of the Fourth USENIX Symposium on Internet Technologies and Systems (2003).

[85] L. Rizzo, pgmcc: A TCP-friendly single-rate multicast congestion control scheme, Proceedings of SIGCOMM, Stockholm, Sweden, August (2000).

[86] A. Chaintreau, F. Baccelli, C. Diot, Impact of network delay variation on multicast session performance with TCP-like congestion control, Proceedings of IEEE Infocom, Anchorage, Alaska, USA (2001).

[87] K. Sripanidkulchai, B. Maggs, H. Zhang, An analysis of live streaming workloads on the internet, Internet Measurement Conference, Taormina, Sicily, Italy (2004).

[88] http://www.planet-lab.org.

[89] http://www.cs.bu.edu/brite/.

[90] http://www.hpl.hp.com/research/iii/projects/medisyn.html.

[91] http://pages.cpsc.ucalgary.ca/~carey/software.htm.

[92] https://producerapps.helixcommunity.org/cmdproducer/.

[93] http://www.camtasia.com.

[94] http://www.isi.edu/nsnam/ns/.

[95] P. Lougher, D. Shepherd, The design of a storage server for continuous med, The Computer Journal 36 (1) (1993) 32–42.

[96] L. Mathy, C. Edwards, D. Hutchison, The Internet: a global telecommunications solution?, IEEE Network 14 (4) (2000) 46–57.

[97] Inktomi, http://www.inktomi.com.

[98] N. Race, D. Waddington, D. Shepherd, A dynamic RAM cache for high quality distributed video, Proceedings of Interactive Distributed Multimedia Systems and Telecommunications Services (IDMS) LNCS 1905 (2000) 26–39.

[99] M. Hofmann, E. Ng, K. Guo, S. Paul, H. Zhang, Caching Techniques for Streaming Multimedia over the Internet, Bell Labs Technical Memorandum, London, 1999.

[100] S. Acharya, B. Smith, MiddleMan: a video caching proxy server, Proceedings of Network and Operating Systems Support for Digital Audio and Video (NOSSDAV') (2000) 114–123.

[101] R. Rejaie, M. Handley, H. Yu, D. Estrin, Proxy caching mechanism for multimedia playback streams in the Internet, Proceedings of the International Web Caching Workshop (1999).

[102] A. Dan, D. Sitaram, A generalized interval caching policy for mixed interactive and long video workloads, Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE) 2667 (1996) 344–351 [chapter 40].

[103] B. Ozden, R. Rastogi, A. Silberschatz, Buffer replacement algorithms for multimedia storage systems, Proceedings of the International Conference on Multimedia Computing and Systems (1996) 172–180.

[104] S. Sen, J. Rexford, D. Towsley, Proxy prefix caching for multimedia streams, Proceedings of IEEE INFOCOM'99 (1999) 1310–1319.

[105] Z. Miao, A. Ortega, Proxy caching for efficient video servers over the Internet, Proceedings of the Ninth International Packet Video Workshop (PVW 99) (1999).

[106] Y. Wang, Z-L. Zhang, D. Du, D. Su, A network-conscious approach to end-to-end video delivery over wide-area networks using proxy user, Proceedings of the IEEE INFOCOM'98 660 (667) (1998).

[107] B. Wang, S. Sen, M. Adler, D. Towsley, Proxy-based distribution of streaming video over unicast/multicast connections", Proceedings of the IEEE Computer and Communications Societies (ICCS) (2002).

[108] C. Diot, B. Levine, B. Lyles, H. Kassem, D. Balensiefen, Deployment Issues for the IP Multicast Service and Architecture, IEEE Network 14 (1) (2000) 78–88.

[109] A. El-Sayed, V. Roca, L. Mathy, A survey of proposals for an alternative group communication service, Special Issue of IEEE Network on Multicasting: an Enabling Technology 17 (1) (2003) 46–51.

[110] Xenoserver: http://www.cl.cam.ac.uk/Research/SRG/netos/xeno/.

[111] T. Plagemann, V. Goebel, P. Halvorsen, O. Anshus, "Operating System Support for Multimedia Systems", Computer Communications Journal, Elsevier, Vol. 23, No. 3, February 2000, pp. 267–289.