Video-on-Demand Networks: Design Approaches and Future Challenges

Abstract

IP network-based deployments of Interactive Video-on-Demand (VoD) systems are today very limited in scope, but there is a strong belief among telecommunication companies that this market will expand exponentially in the next few years. In this paper, we outline the components of VoD architectures and survey the current approaches to their design. We strive to identify the research challenges that must be addressed in the development of design tools that can determine how to expand upon an existing network infrastructure in order to support Video-on-Demand. The 'long-tail' of content and extensive usage growth are expected to have a major impact on the streaming and storage requirements of such systems. Hybrid VoD architectures that incorporate peer-to-peer exchange are an extremely promising paradigm, but there are many challenges in developing operational and economically-feasible peer-to-peer systems. VoD networks generate sufficient traffic that their impact should be considered in planning general network infrastructure expansions.

I. Introduction

True or Interactive Video-on-Demand (iVoD) dedicates a single channel to each user and allows the video to be started at any time with VCR-like controls (pause, rewind, fast-forward, etc.). IP network-based deployments for services such as iVoD are today very limited in scope, with the largest deployed libraries achieving a mere 0.7% (5,000 hours) of the global movie and TV-series catalog (DVD by mail services such as Netflix offer up to 10%) and peak utilization reaching only 10-15% of broadcast TV. There is a strong belief among telecommunication companies that this market will expand exponentially in the next few years [1]. Service providers are intensely interested in scalable design methodologies for the deployment of large-scale IP content delivery networks that provide content propagation, storage, streaming, and transport.

Extensive video-on-demand (VoD) systems have the potential to consume enormous bandwidth. The interactivity requirement and lack of network support restrict the widespread application of multicast streaming, so many concurrent unicast streams must be supported, and even when state-of-the-art compression methods such as MPEG-4 are employed, hundreds of Gigabits of streaming capacity are required. The high bandwidth requirements motivate distributed architectures with replication of content and localization of network traffic, but such architectures imply a substantial increase in storage requirements, a non-negligible factor given the large size of video files.

Many existing design tools ignore existing network infrastructure and focus on greenfield deployments, a situation that is rare in practice. There is an urgent need for design tools that can determine how best to expand upon existing network infrastructure and where to deploy storage and streaming devices in order to support the additional demand imposed by VoD systems. The tools must strive towards generating *scalable* architectures that can be expanded incrementally to handle order-of-magnitude increases in library size, rate of ingest of new content, and peak demand.

In this article, we outline the components of VoD network architectures and survey the current approaches to the design of the these components. For each of these aspects of the VoD design challenge, we identify issues that are yet to be addressed or have received insufficient attention. Some of the key open research questions we raise are the following: (i) how should peer-to-peer content exchange operate in VoD systems and how should it affect pricing mechanisms? (ii) how should VoD network design methods take into account 'long-tail of content' effects that emerge as libraries and user-groups expand? (iii) how can network designs meet current demands at low cost yet remain suitable for expansion? (iv) how can a VoD network adapt to changing patterns of user behaviour and evolutions in the nature of accessed content?

In Section II we present the architectures and topologies that have been considered for content distribution (delivery) networks (CDNs) and VoD deployments as well as the mechanisms required to build a network cost model (content allocation and distribution, request routing, and content delivery). In Section III we summarize the models used to describe file access patterns, traffic, and network cost. In Section IV we present the optimization strategies (replica location, video placement and resource allocation) used to produce cost-efficient solutions during the planning phase of video-on-demand network deployment. Finally, in Section V we discuss open research challenges that exist in the design and development of VoD networks. The reader is invited to refer to a review on content delivery networks for a more detailed description of the mechanisms and models we discuss [2].

II. VoD Network Architecture and Mechanisms

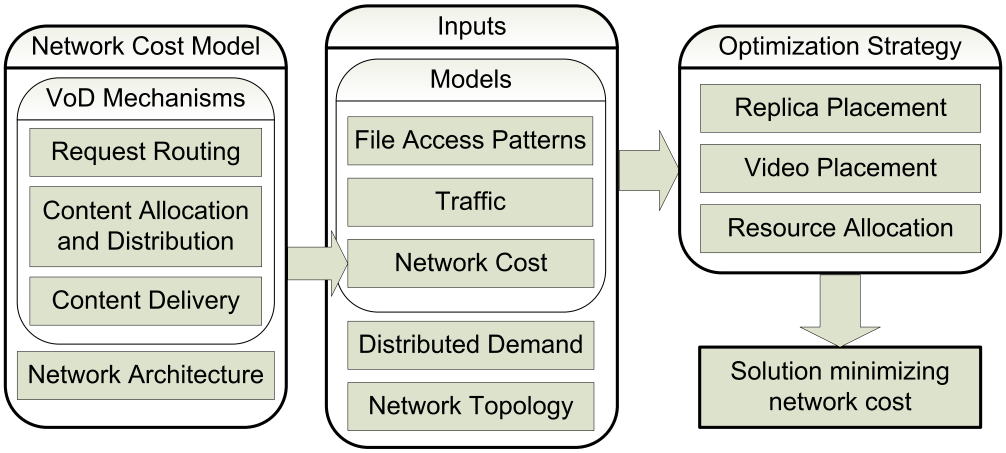


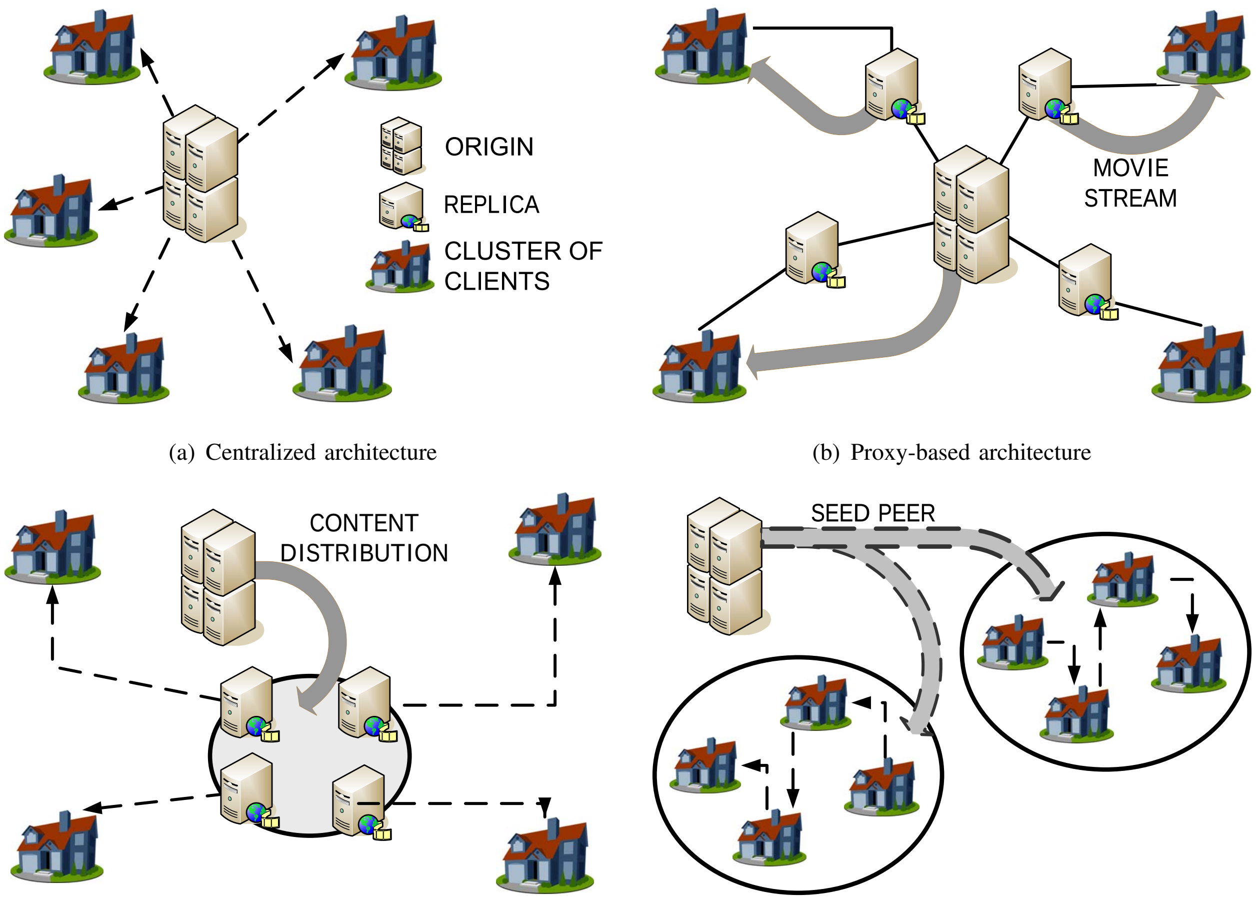
Fig. 1. Block diagram showing the the interaction of video-on-demand networks mechanisms, models and optimization strategies involved in the planning phase of a VoD network deployment. Based on the mechanisms and network architecture, a network cost model maps a solution to one of the optimization problem (replica placement, video placement or resource allocation) to an actual network cost. The models for network cost, traffic and file access patterns as well as information about the distribution of demand and the network topology are typically required to solve these problems.

When a user wants to watch a video, he simply has to make a selection from a list of available titles and within a few seconds, the video is ready to be started. The provision of such a simple interface requires many complicated network mechanisms that remain invisible to the users. Most VoD network designs use proxy servers (replicas) to cache content in order to minimize latency at the user-end and reduce bandwidth consumption and load at the server hosting the movie library (the origin), as depicted in Fig. 2(b). The *request routing* mechansim redirects each user's request to the most appropriate server according to a variety of metrics such as distance, network load or content availability. Depending on the technique used for *content delivery*, the server transmits the chosen content to the user either via a dedicated unicast stream or through a multicast connection where many clients receive the same stream. The approach adopted for *content allocation* (where storage/streaming devices and video files are placed in the network) influences both the request routing and content delivery. The large size of video files makes it expensive and impractical to replicate the entire library at each site and generally only a fraction of the most popular titles or file prefixes are stored at each replica. Distributing these files to the replicas consumes significant bandwidth, so the mechanism for *content distribution*, the process of sending content from the origin server (library) to the replicas, must be carefully designed; it is either triggered by a user's request or executed periodically to update the cache. At a higher level in the network hierarchy, there must be mechanisms for *content ingestion*, which govern how and when new content is added to the network. The rate of content ingestion can vary significantly depending on the nature of the content provided by the VoD system. A library consisting predominantly of movie and documentary content might be expected to grow slowly, with a few titles added every day, whereas a library with significant TV content can grow at a very rapid rate. It is highly likely that the usage and content of a VoD network will evolve over time, so a design must be robust to changes in usage patterns and ingestion rates.

Various optimization strategies exist to produce designs that minimize network cost whilst respecting quality constraints. A classical formulation is the replica placement problem, which consists of determining the location of replica servers such that the infrastructure cost is minimized. Alternatively, the video placement problem involves the identification of the location and number of copies of each movie file in the network that minimizes bandwidth consumption. Finally, resource allocation problems explicitly consider the number of streaming, storage and switching devices required to meet quality requirements while minimizing network cost. The inputs required to solve these problems include the distributed demand, the network topology (inter-nodal distances) and models for file access patterns, traffic and network cost. File access patterns are used to determine the fraction of requests for a given group of objects, traffic models are used to estimate the bandwidth requirements of a movie stream, and network cost models take into consideration the VoD mechanisms described above and the network architecture to determine the projected deployment cost of a VoD network solution (see Fig. 1).

A. Architecture and Topology

In a centralized architecture, the origin server is responsible for serving all the clients (Fig. 2(a)). Although simple to implement, this approach has serious weaknesses: a single point of failure and high load on both the origin server and the surrounding network links and switches. These shortcomings have prompted the development of distributed architectures with proxy servers installed at strategic locations in the network (closer to the clients). The proxy servers, located close to the user-end, cache content to reduce the load on the origin, as shown in Fig. 2(b). A request from a client is served by a proxy if it has a copy of the requested file; if it does not, the



(a) Centralized architecture

(b) Proxy-based architecture

(c) CDN approach

(d) Hybrid architecture

Fig. 2. Architectures used for media delivery. (a) In the centralized architecture, all the requests from the clients are handled at the origin server. (b) Proxy servers located close to the user-end reduce the load on the origin server by caching content to serve a fraction of the clients' requests. (c) Content delivery networks (CDNs) are third-party solutions that deploy proxy servers in the core of the network (close to the edge) to serve a fraction of the clients' requests. (d) In a peer-to-peer (P2P) approach, the peers share their resources to distribute the media. Powerful peers help in routing the requests and searching for content.

file is streamed from the origin. An extension of the proxy server approach is the use of content distribution networks (CDNs); in these networks, requests can be referred to other CDN servers (replicas or surrogates), which are generally located at the edge of the network core (Fig. 2(c)).

Hefeeda et al. argue that proxy-based approaches simply shift the bottleneck from the origin to the proxy servers and are not cost-effective solutions for streaming media [3]. Optimization exercises using reasonable, although necessarily approximate, network cost models contradict this argument, indicating that there is potential for substantial savings by deploying proxy servers at local network exchanges provided there is sufficient demand [12]. A major reason behind these design results is the rapid reduction in the cost of memory in recent years, which has led to a shift in the cost trade-off between replica deployment and bandwidth consumption on network trunks. Hefeeda et al. propose a hybrid architecture based on the peer-to-peer (P2P) paradigm to distribute the files to the users (Fig. 2(d)) where the origin acts as a seed peer to help in routing the requests and searching for content [3]. Hybrid architectures are an extremely promising paradigm, because the storage and streaming resources scale (approximately) with the number of users. The architectures thus have the potential to adapt to growth in library size and usage. The details of such systems, including numerous operational issues and pricing considerations, remain to be resolved.

B. Content Allocation and Distribution

Server replication consists of placing entire copies of the origin server at strategic places in the network, thereby partitioning the network, localizing network traffic, and reducing bandwidth requirements at the expense of increased server and storage cost. It is generally impractical to have complete replicas of the origin server because the large size of multimedia objects leads to an excessive storage cost. An alternative is *program caching*, a scheme where only popular objects are stored at the surrogate servers; storing the most popular thirty percent of files can often accommodate up to eighty percent of requests.

The popularity of an object changes over time and an initially hot (popular) file frequently becomes cold (the number of requests falls below a given threshold). In order to ensure that the replicas can serve the majority of requests, it is important to regularly replace objects that have become cold with new hot objects; this is the problem of content distribution, and there are two main approaches. In the client-initiated approach, or *pull-caching*, the replica retrieves the copy of an object in the case of a cache miss. In the server-initiated approach, or *push-caching*, content is distributed to replicas before any requests for this data have been made. If we anticipate that a specific object will be very popular (e.g., a blockbuster movie release), it is advantageous to distribute the object prior to any requests in order to avoid cache misses and reduce bandwidth consumption on the origin-replica path.

Several approaches exist for improving the performance of the surrogate servers. Storing file prefixes rather than full files at the replicas (*prefix caching*) improves the hit ratio of the caches, reducing delivery cost and traffic on the origin-replica path but preventing clients from experiencing delays and jitter. Sharing the content of different surrogate servers by grouping them into clusters can also improve performance; this follows the philosophy of content distribution networks. The extent of the benefit it provides is dependent on the underlying network topology, increasing when there are direct paths between surrogates. Clustering avoids the duplication of content at servers that are close to each other according to some distance metric (spatial, temporal, session clustering or popularity-based). Push-caching, prefix caching and clustering increase the complexity of the network, but improve the scalability of VoD designs.

As the number of users and videos in the library grows, it becomes more difficult to maintain a good hit ratio at replicas without dramatically increasing the storage capacity. Beyond the clear expectation that the size of the cache must scale with that of the library, there is the potential for 'long-tail of content' effects to influence the effectiveness of caching in large-scale VoD systems [4]. These effects, which have been observed in on-line music and book stores and mail-based video rental companies, arise from the tendency for a significant percentage of requests to be directed towards less popular objects. Therefore, it is important to consider the heavy tail of the popularity distribution when modeling user-demand patterns. This has a direct impact on the storage and streaming requirements at each location and on the choice of content allocation (program caching vs. prefix caching) and distribution (push-caching vs. pull-caching) mechanisms. It enhances the attractiveness of peer-to-peer transfer, which allows users with similar interests to share content and better exploits the available storage in the network.

C. Request routing

Request routing is the process of directing the client requests to the best server in order to avoid the servers that provide low performance without overloading the others. It is applicable only in VoD systems where the underlying network topology makes it beneficial for multiple surrogates to service a client. This is not the case in many metropolitan-area networks where a tree or ring topology implies that service from another replica involves a higher bandwidth cost than service from the origin server (and concentrates the load over the same links).

When request routing is beneficial, one of the simplest approaches is the Round-Robin (RR) algorithm [5]. This algorithm selects which surrogate serves a specific request in a cyclic mode without considering the state of the network. There are many schemes which use various metrics to make a better decision than the RR algorithm. Metrics include the response time the user previously experienced with a particular server and the client-replica path's current utilization.

When surrogate servers do not host all files in the library, content availability must also be considered when routing requests (see [6] and references therein). A simple method is the query-based scheme, in which a proxy broadcasts a query to other nodes in its cluster if it does not have the requested content locally. In the digest-based scheme, each proxy maintains a list of the information stored at other replicas (alternatively, the list can be centralized on a directory server). Jian et al. propose an approach that uses the content's URL, the address of the proxies and a hashing function to redirect the request to a designated proxy [6]. Their enhancement consists of reserving a portion of storage at each proxy for local popular content to improve performance. The formation of clusters and presence of a proper routing mechanism are not essential for current deployments, but their implementation is necessary for scalable Video-on-Demand systems.

D. Content Delivery

Content delivery consists of transmitting objects from the surrogate servers (or origin servers) to the clients. VoD is generally unicast in nature (there is a dedicated stream to each user), which imposes significant bandwidth pressure on the network, but permits the provision of interactive VCR functions. In multicasting schemes, the video is transmitted with a pre-defined schedule on a dedicated channel that supports any number of clients with a constant amount of bandwidth. An example of multicasting is batching, which collects requests that arrive within a given time interval and then multicasts the stream to the clients [7]. In patching (stream tapping), if there is no stream for a video, then one is initiated when a client requests it. If it already exists, then the client simultaneously listens to the multicast stream of the video and retrieves, from a proxy server, the part of the video that was streamed before the client joined the broadcast. The problem with aforementioned techniques is that they all require the path between the server and the client to be multicast-enabled, but this capability is far from being fully deployed. Alternative solutions when the end-to-end network provides only unicast service is to use proxy-assisted transmission schemes (one-to-one connection between the server and the client), application-layer multicast (end hosts maintain a data forwarding path) or peer-to-peer delivery.

Although VoD is typically considered a unicast-based application, implementing some form of multicast delivery is crucial to reducing the load on the network and the servers in large-scale deployments. As mentioned previously, using peer-to-peer delivery is a solution that decreases bandwidth requirements while maintaining the interactive VCR functions that VoD provides.

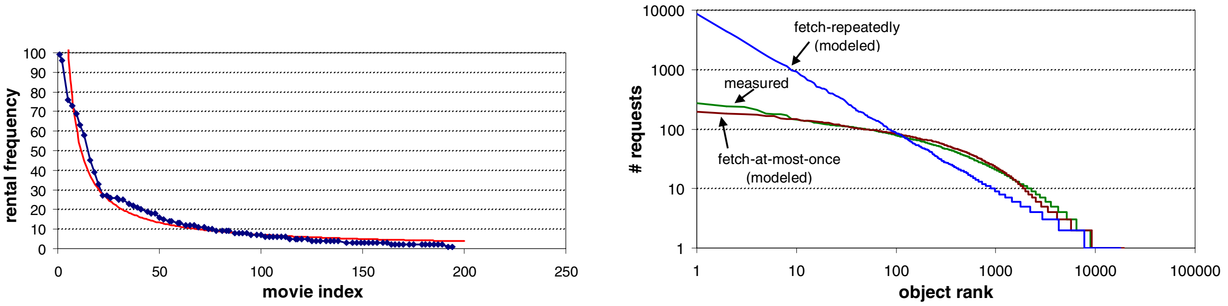
III. VoD Network Models

A. File Access Models

When allocating content with a program caching scheme, only the most popular files are stored, with the aim of minimizing the storage and bandwidth needs. By using an appropriate file access model, we can predict the hit ratio at a replica site given the set of files it is hosting. The hit ratio represents the probability that a user's request is served at a given replica.

Previous studies exploring the distribution of multimedia files in CDNs have used Zipf's Law to characterize the popularity of the different files [6]–[8]. In Zipf-like distributions, the access frequency for a file of popularity rank i is equal to C/i^{α}, where C is a normalization constant and $\alpha > 0$ is the distribution parameter. In the VoD context, models have been proposed for file access patterns based on the long-term life cycle of videos and the variations in user population sizes and user behaviour throughout the day (children's interests dominate during the afternoon whereas adults' interests dominate later in the day). However, based on dated video store rental statistics, many researchers still adopt the Zipf approach to model popularity in video-on-demand applications. Although this data seems to fit a Zipf curve (Fig. 3(a)) on a linear scale, Fig. 3(b) shows that the part of the curve for the most popular files is flattened and does not fit the Zipf linear curve on a log-log graph.

Gummadi et al. explain this behaviour by analyzing the characteristics of video object access [9]. VoD system users rarely access the same file twice because the files are not modified (fetch-at-most-once) and new files are often added to the system. The Zipf distribution is a better fit for retrieval of Web objects which are accessed more than once because they are updated regularly (fetch-repeatedly). Since the popularity of a video diminishes in time, new videos become the most popular titles when they are added to the system. Models that are driven by Zipf's Law but take into account the "fetch-at-most-once" and "new arrivals" factors provide a much better fit to the video rental statistics distribution than standard Zipf models (see Fig. 3(b)) [2], [9].



(a) linear scale

(b) log-log scale

Fig. 3. (a) The popularity distribution from a 1992 video rental data set used to justify Zipf's law for file access models used in many Video-on-Demand system designs, along with a Zipf curve fit with $\alpha = 0.9$, and (b) the same data set and curve fit plotted on a log-log scale. Contrary to the assumption of many papers, video rental data does not appear to follow Zipf's law. Gummadi et al. show that their fetch-at-most-once model fits the data more accurately than a fetch-repeatedly model. (Reproduced from [9])

B. *Traffic Models*

VoD network design requires models for the traffic generated by high quality video because they indicate the bandwidth that a system must provide to support a set of streams. Compression methods like MPEG-4 can substantially reduce bandwidth requirements while maintaining good quality. MPEG videos are encoded using a variable bit rate (VBR) which generates bursty traffic, making it difficult to predict the required resources. Reserving resources based on average rates results in long delays when the source is transmitting at peak rate, but if the system is designed to support the peak rates, it is significantly under-utilized outside the high-usage periods. One way to eliminate the peaks and maintain a constant rate is to deliver videos ahead of time but this requires users to have a device that can store the content at home. An important consideration when predicting the required bandwidth is that when several videos are transmitted simultaneously on the same link, there is a multiplexing gain. A set of video streams does not display the same burstiness as a single stream, because alignment of the peak rate periods of the video streams is rare.

C. *Network Cost Models*

The network cost model typically includes the cost of the equipment required for transporting (switches and fiber), transmitting (streaming devices) and storing (storage devices) the data. The resources required for each of these tasks depends on the architecture and mechanisms comprising the VoD network, as described in Section II. The request routing and content allocation and delivery schemes determine the amount of traffic on each link and the streaming and storage capacity required at each replica (as a function of the distributed demand, file access pattern and traffic model).

When determining transport cost, it is important to consider the internodal distances between clients, replicas and origin servers. Many metrics are used to represent distance such as network latency, number of hops, or link cost (also called bandwidth cost). Routing to the closest server does not necessarily lead to lowest cost. Instead, to calculate the delivery cost, total network bandwidth can be expressed as the (possibly weighted) sum of the bandwidth required for each hop on the delivery path. Another key parameter is the storage server cost, or replication cost, of keeping a copy of an object at a given location. Server cost also includes a fixed start-up cost (installation cost) and the cost of the required bandwidth, which is proportional to the popularity of the files it stores (where the content allocation mechanism and file access model play a role). For example, a server that hosts very large files (high storage cost) which are not popular (like archives) has low bandwidth requirements.

Most VoD design approaches consider the scenario where the VoD service provider does not own any network equipment or infrastructure prior to the deployment. However, this is generally not the case; for example, the service provider might be able to transport data for free, i.e., there may be no need to install additional fiber, network interfaces, switches, or amplifiers. Even if there is no network installation cost, there are still fees incurred by the usage and maintenance of the equipment and the resources, which have to be considered when generating solutions for this scenario. In the case of a peer-to-peer architecture, cost models must account for the upstream traffic generated by the customers.

An important issue that has been largely neglected is the design of (and development of cost models for) incremental deployments. It is unclear how best to gradually expand a design and iteratively add resources to a network. In such evolutionary designs, should include as extra parameters the bandwidth, streaming and storage capacity currently available. It should also take into account projected growths in library size and usage as well as price variations of equipment over time. Such designs would make possible multi-stage deployments that could adapt over many years to growing needs in storage and streaming capacity.

IV. Optimization Strategies

Several different optimization strategies have been adopted with the aim of generating optimal VoD network designs; the most prevalent three formulations are the replica placement, video placement and resource allocation problems. The *replica placement problem* and *video placement problem* consist of determining the location and number of replicas (or video files) in the network such that the performance is maximized given an infrastructure or that the infrastructure cost is minimized for a given quality of experience (QoE) impairment, such as delay, packet loss, frame loss, or packet jitter [10]. Laoutaris et al. argue that the replica and video placement problems should not be solved independently of the resource allocation problem [11]. Resource allocation is an important and complicated task that consists of determining the location and number of networking resources to deploy such that cost is minimized whilst certain conditions are respected.

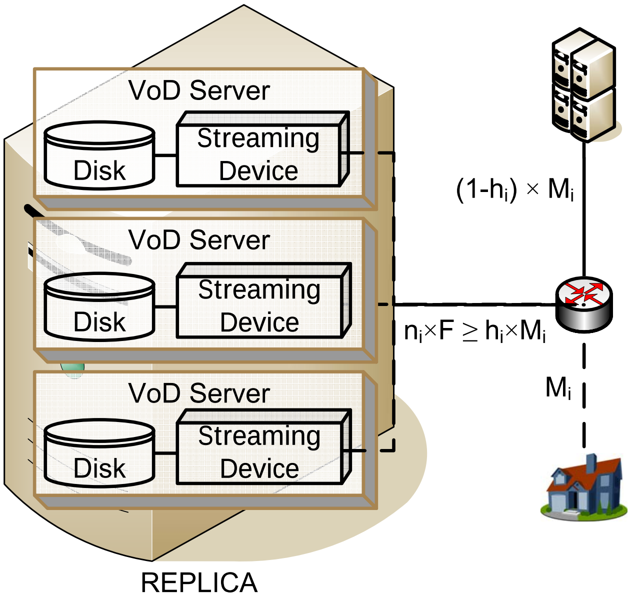


Fig. 4. A replica with $n_i = 3$ VoD servers each with storage capacity of G_i TB and streaming capacity of F_i Gbps. The total streaming capacity $n_i \cdot F_i$ should be greater or equal to $h_i \cdot M_i$ where h_i is the hit ratio at site i given the storage capacity $n_i \cdot G_i$ and M_i is the worst-case demand from the attached group of clients.

An example of the resource allocation formulation is the *storage capacity allocation problem*, which involves identifying how to best distribute storage capacity (from an available storage capacity budget) to the nodes of a hierarchical content distribution system, given known access costs and client demand patterns. Wauters et al. addressed the resource allocation problem from a different perspective by explicitly determining the equipment required for transport (the number of ports at each server and the number of multiplexers and switch ports at each network node) [8]. In [12], Thouin et al. defined the *VoD equipment allocation problem* as the task of determining the number of VoD servers (see Fig. 4) to deploy at each potential location in a network topology such that the total demand is satisfied and the deployment cost is minimized. Solving the VoD equipment allocation problem determines the location of the replicas, the amount of storage available to cache content and the streaming capacity available to serve clients at each location. The case studies in [12] indicate that for scenarios where the demand is unevenly distributed amongst the clusters of clients serviced by each potential replica location, it is beneficial to cache content in only a small fraction of locations. Furthermore, the type of equipment available for installation at each location has a significant impact on the optimal design and the deployment cost (in particular the ratio between storage and streaming capacities). Even if a wide variety of VoD server models is available, it is difficult to select a single model that matches the requirements of all locations. A natural extension of the problem thus involves identifying the best type of equipment to install at each location.

Better solutions can be achieved if VoD network topology design is integrated into the optimization strategy; deciding upon the *potential* sites for replicas, which effectively partitions the demand, is as important as deciding the eventual locations. The joint design of topology and VoD network architecture is a fascinating and challenging task that remains to be addressed satisfactorily. The problem consists of choosing a topology which allows an allocation of resources that minimizes the deployment cost of the network. Whereas previous optimization approaches assume a given set of inter-nodal distances, potential replica location positions and distributed worst-case demands, in this joint optimization problem these variables are unknown; the number and position of potential replica locations become part of the set of optimization parameters.

V. CONCLUSIONS AND THE FUTURE

We have reviewed VoD network architectures (centralized, proxy-based and peer-to-peer) and mechanisms (request routing and content allocation, distribution, ingestion and delivery) as well as relevant models (file access, traffic, network cost) and optimization strategies (replica placement, video placement, resource allocation) involved in the design of low-cost solution during the planning phase of network deployment.

Proxy-based architectures offer important savings in deployment cost, but their scalablity is questionable. As it is expected that usage and library size will grow exponentially in the next few years, designs must have the capacity to support significant increases in traffic and streaming and storage capacity. In particular, it becomes difficult to maintain the hit ratio without increasing storage capacity because a larger fraction of file requests are for less popular objects, i.e., objects in the heavy tail of the distribution, known as 'the long-tail'. This not only affects storage and streaming requirements at each location, but makes prefix caching and push-caching more desirable approches and motivates the development of new file-access models.

Hybrid peer-to-peer architectures have much better scaling capabilities because the available resources increase with the number of users. They offer a solid alternative to multicasting delivery to reduce high bandwidth requirements of unicast delivery while maintaining the interactive VoD features. While there has been excellent preliminary work in this area, there are substantial challenges in developing practical mechanisms for peer-to-peer transfers that allow users with common interests to share content. Network cost models in such architectures must take into account the upstream bandwidth that is consumed; peers must presumably be rewarded for the resources they are providing.

There exist key challenges in developing VoD network design tools that take into account the existing resources and network infrastructure. The tools should be capable of producing evolving VoD network designs, wherein resources are iteratively added to the network as demand grows. New network cost models that include the available resources and take into account growth factors and price variations are required to underpin these design tools. Finally, VoD networks are capable of producing a sufficient volume of traffic that their impact should be considered in planning where to add general network infrastructure. The integration of VoD system design and network expansion planning is a promising area of future research.

REFERENCES

[1] K. Couch, "Raising the bar for triple play with VoD," *Converge! Network Digest*, January 2005, http://www.convergedigest.com/blueprints/ttp03/2005nortel1.asp?ID=189&ctgy=Headend.

[2] F. Thouin and M. Coates, "A review of content delivery networks," McGill University, Montreal, Canada, Tech. Rep., Apr. 2005, http://www.tsp.ece.mcgill.ca/Networks/projects/pdf/thouin_content_delivery.pdf.

[3] M. M. Hefeeda, B. K. Bhargava, and D. K. Y. Yau, “A hybrid architecture for cost-effective on-demand media streaming,” *Computer Networks*, vol. 44, pp. 353–382, 2004.

[4] C.Anderson, “The long tail,” *Wired Magazine*, October 2004, http://www.wired.com/wired/archive/12.10/tail.html.

[5] M. Masa and E. Parravicini, “Impact of request routing algorithms on the delivery performance of content delivery networks,” in *Proc. Int. Performance Computing Communications Conf. (IPCCC)*, Phoenix, AZ, Apr. 2003.

[6] N. Jian, D. Tsang, I. Yeung, and H. Xiaojun, “Hierarchical content routing in large-scale multimedia content delivery network,” in *Proc. IEEE Int. Conf. Communications (ICC)*, Anchorage, AK, May 2003.

[7] B. Wang, S. Sen, M. Adler, and D. Towsley, “Optimal proxy cache allocation for efficient streaming media distribution,” in *Proc. IEEE Infocom*, New York, NY, June 2002.

[8] T. Wauters, D. Colle, M. Pickavet, B. Dhoedt, and P. Demeester, “Optical network design for video on demand services,” in *Proc. Conf. Optical Network Design and Modelling*, Milan, Italy, Feb. 2005.

[9] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan, “Measurement, modeling, and analysis of a peer-to-peer file-sharing workload,” in *Proc. ACM Symp. Operating Systems Principles (SOSP)*, Bolton Landing, NY, Oct. 2003.

[10] W. Tang, E. Wong, S. Chan, and K. Ko, “Optimal video placement scheme for batching VoD services,” *IEEE Trans. on Broadcasting*, vol. 50, pp. 16–25, Mar. 2004.

[11] N. Laoutaris, V. Zissimopoulos, and I. Stavrakakis, “On the optimization of storage capacity allocation for content distribution,” *Computer Networks Journal*, vol. 47, pp. 409–428, Feb. 2005.

[12] F. Thouin, M. Coates, and D. Goodwill, “Video-on-demand equipment allocation,” in *Proc. IEEE Network Computing and Applications (IEEE NCA)*, Boston, MA, July 2006.