DESIGN OF A SCALABLE VIDEO ON DEMAND ARCHITECTURE

Philip Machanick
Department of Computer Science, University of the Witwatersrand
2050 Wits, South Africa
philip@cs.wits.ac.za

Abstract

The SAVoD (Scalable Architecture for Video on Demand) approach to video on demand exploits the fact that end-user latency goals are relatively modest, while increasing bandwidth is much easier than improving bandwidth. Instead of classic approaches which add latencies, making system-level latency goals hard to achieve, the SAVoD approach shares information across multiple users. Unlike other video on demand (VoD) approaches, SAVoD broadcasts a fixed number of replications of a given stream. VCR-like operations are implemented by switching streams (to another instance of the same stream or to a different stream). The result is a system that scales up well: additional network traffic is only generated by an increase in either the video standard or the number of simultaneously available movies, and the number of users has no impact on system requirements. Instead, increasing users makes an improvement in quality of service more affordable. With a modest budget of 32Kbit/s per user, 69,000 users are required to justify a new HDTV movie, or 23,000 users for a broadcast-quality service.

1. Introduction

A widespread problem in many computer applications is meeting latency goals. Often, the most important latency goals relate to response times seen by users, which seem slow by computer standards. However, scaling up to large numbers of users presents a problem.

A classic example of this problem is video on demand: a service where any user can run any movie from a library when they want to, with VCR-like operations. Acceptable latencies for actions like fast forward or changing to a new movie are many seconds, even a minute or more (given slowness of the competition, a VCR), yet achieving these latencies with a very high number of users is problematic, as witnessed by the complexity of proposed architectures for large-scale VoD systems [Taylor *et al.* 1995, Jadav and Choudhary 1995, Dan *et al.* 1994].

The approach proposed here is to blow away large amounts of bandwidth, generally easier to scale up than latency, to achieve the desired response times. The proposed approach is to stream each movie continuously at 1-minute intervals which, for a typical 2-hour movie, requires 120 times the bandwidth of a single stream. At the user end, a relatively simple set-top box sends signals to an interface to the high-speed interconnect in a local station remote from the user's residence to perform actions like fast forward (skipping to another version of the same movie which is ahead in time) or to another movie. The worst-case latency for operations is 1 minute, and can be improved by caching the movie or portions of the movie on locally.

A high-capacity interconnect is required to carry the full capacity of the overall service. However, a local station containing interfaces to the high-speed interconnect can tune to a desired channel, so a relatively low-speed link is needed to each user (sufficient bandwidth for one movie plus control signals). While the amount of bandwidth in total is high (for 1,000 simultaneously available HDTV-standard movies, 2Tbit/s), the bandwidth is shared between many users. If a budget of 32Kbit/s is allowed per user, 69,000 new users have to be added to justify adding a single new HDTV movie. For broadcast-quality service, only 23,000 new users would be needed to justify adding one more movie to the service with this bandwidth budget.

Contrast requirements with a full video on demand system where any user can be at any stage of viewing any movie: 120 times the bandwidth of all available movies is a relatively modest total requirement.

To phase such a system in, a lower-specification service can be offered at first (e.g., MPEG-1 with a modest number of movies). The attractive feature of the design is that the cost per user goes down as users are added, making it possible to increase the number of movies or video specification, or both, as the user base increases. Since there is no traffic back to the server, server and network requirements are purely a property of the number and quality (in terms of video specification) of movies on offer. This contrasts with other VoD proposals, where an increase in the number of users requires either or both of an increase in the complexity of the server or interconnect.

The remainder of this paper presents an overview of approaches to video on demand, a more detailed description of the design principles of the SAVoD approach, a design of a SAVoD system with alternatives for different levels of service, and a discussion of a roll-out model for the technology. In conclusion, the major points of the design are summarized, followed by a discussion of future work.

2. Video on Demand

The general video on demand (VoD) problem is to provide a library of movies where multiple clients can view movies according to their own requirements in terms of when to start and stop a movie. A VoD server implements a virtual VCR, with all the usual functions: fast forward, rewind, pause, switch to a new movie.

In practice, a full VoD system is not achievable except on a modest scale because the bandwidth an latency requirements do not scale well. Each user adds extra requirements onto the server and the network.

Some compromise solutions attempt to reduce the requirements of the server and network.

Near Video on Demand (NVoD) groups sessions of the same movie together as far as possible in multicast sessions. To do so requires delaying (even sometimes rejecting) requests. Fast forward and rewind are implemented by jumping between multicasts [Abram-Profeta and Shin 1997].

Adaptive Video on Demand (AVoD) is another variation. Like NVoD, AVoD attempts to batch requests, but also treats the decision as to whether to run a given movie as a scheduling problem [Aggarwal *et al.* 1994].

Unlike with the proposal presented here, any attempt at batching requests (as in NVoD and AVoD) has the property that it pays off to delay requests for a popular movie [Dan *et al.* 1994], a strategy which makes little sense from the perspective of marketing and ensuring a high-quality user experience.

In general, other approaches to VoD require much more complex servers [Chen *et al*. 1994, Jadav and Choudhary 1995, Taylor *et al.* 1995] and networks [Byun and Lee 1994] than the SAVoD approach, and do not scale up as well, particularly in terms of number of users.

3. Principles for SAVoD

The most important design principles if the SAVoD are:

- *scale up and down*—aim to scale up to millions of users and hundreds of movies but design for a small system as well to allwo for a roll-oout model and a range of applications from the mass market to small-scale systems

- *push bandwidth requirements back to the server*—storage media are generally much more efficient at streaming data continuously than moving small randomly located data items, so this capability should be exploited as fully as possible

- *push latency requirements to the client*—low, particularly low and predictable, latency is hard to achieve on a server, especially as it scales up; latency goals are much easier to achieve on non-shared equipment, so a design principle is to put aspects of the system that require responses to end-user requests at the client side

These principles, the latter two of which are designed to separate latency and bandwidth goals—which are usually in competition [Hennessy and Patterson 1996]—result in the following goals to guide the design:

- *keep the server simple*—anything in the server design that requires scaling up with the number of users spells trouble; focus on getting the data out as fast as possible, and leave all user-interaction to the client side

- *maximize mileage from the common case*—if 1-million people want a movie at almost the same time with variations in times they may want to take breaks, design for this case and make sure it scales up

- *focus on the user experience*—rather than starting with technology, focus on what a user will expect and design to that, even if the technology ends up less complex than might otherwise be expected

- *blow away bandwidth*—if the goal is scalability to very high numbers of users, the cost of extra bandwidth to meet user-experience goals is widely shared, so don't worry about it

The following section shows how these principles and goals lead to a relatively simple design which provides a user experience comparable with a VCR, yet allowing a large selection of movies as the number of users scales up.

4. Design of a SAVoD System

4.1 introduction

The SAVoD proposal is to broadcast 120 copies of each 2-hour movie at 1-minute intervals, which allows VCR operations to take place by skipping backward or forward either within a stream or to another stream. Maximum latency for any operation is 1 minute, with the possibility of local buffering to reduce delays.

The architecture consists of five major components:

- server

- high-speed interconnect

- local station

- local interconnect

- set top box (STB)

The remainder of this section examines the role of each component.

4.2 server

The server for a single movie consists of little more than a high-speed disk (possibly RAID: see Section 5 for requirements), and a multiplexer. The high-speed disk streams out a movie at 120 times its real-time speed, and a movie is divided into frames (F_i) by a multiplexer, corresponding to 120 streams. On a given pass over the movie, if F_i belongs to stream j, then F_{i+1} belongs to stream j+1, i,j [0..119], all additions modulo 120.

The effect of this streaming is that 120 copies of the movie are simultaneously broadcast on the high-speed interconnect, each 1 minute apart in time (assuming a 120-minute movie).

4.3 high-speed interconnect

The high-speed interconnect does not require any complex routing: the same data is sent over the entire interconnect. This part of the system should go from the server to near enough to the final user that a relatively low-cost interconnect to the user can be used, with low enough latency for interaction.

All that is required at this level of the system is:

- sufficient bandwidth for the required number of movies which are simultaneously available, times 120
- high enough reliability that any loss of data will not visibly affect movie quality
- accurate enough timing within and across streams to avoid artifacts like jitter

By comparison with other multimedia proposals, the interconnect is extremely simple. For a high-end system, a high-bandwidth fiber-optic interconnect is needed, but should not be difficult to design.

4.4 local station

The local station connects to the high-speed interconnect, and allows the user to perform VCR-like operations over the local interconnect, which carries video and control signals.

The local station requires an interface per subscriber to the high-speed interconnect, capably of synchronizing to any required stream, demultiplexing the required stream, and skipping within or between streams to perform actions like fast forward, rewind or select a new movie. Unlike the server, the local station interface does not need to keep up with a 120-times speedup.

To smooth out VCR-like operations, the local station could buffer sections of a movie (even a whole movie). Whether this was done per user or on a shared server is not critical to the proposal, as this end of the system does scale per user (unlike the rest of the system, which introduces the relatively novel notion of a system where the number of users has no impact on load).

4.5 local interconnect

The local interconnect only needs to support the bandwidth of one movie, plus control signals. Conventional cable television technology would be acceptable though to allow for longer-term planning, fiber could be considered. A higher bandwidth would also be useful if the same local cable were used for other purposes like internet or other connectivity.

4.6 set top box

The set top box (STB) in this model need not be very complex, since most of the complexity is at the local station, where it is necessary to have reasonably sophisticated hardware to interface to the high-speed interconnect. The STB needs to be able to convert VCR-like commands to control signals; the incoming video signal could go straight to the television.

5. A Roll-Out Model

5.1 introduction

An attractive feature of the design is that, at least with relatively low-specification video, it is possible to justify the cost with a relatively low number of users.

Since much of the technology is new or has not been applied in this way before, a simple model is to allow a bandwidth budget per user. If a relatively modest budget of 32Kbit/s is allowed per user (less than is achievable on an analog phone line, a technology brought close to almost all homes in the industrialized world), it becomes possible to estimate how many users are needed before the cost of adding a new movie is justified. This is a rather modest bandwidth budget; more detailed costing would likely result in a less pessimistic roll-out model than is presented here, particularly in high-density applications (cities, within a building, or in-flight entertainment, where the high-speed interconnect is short).

Here, three levels of service are considered, an entry-level MPEG-1 system at 1.5Mbit/s, a broadcast-quality 6MBit/s service and an HDTV 18MBit/s service, followed by a discussion of how to roll out SAVoD to the market.

5.2 entry-level system

For an entry-level system, the bandwidth per movie is 180Mbit/s, or 22.5Mbyte/s—which is achievable with a high-end disk, provided no bandwidth is lost to overhead. If 32Kbit/s is allowed per user, 5760 users are required to justify the cost of adding each new movie.

Such a system might, however, given the relatively modest bandwidth per movie, be deployable in low-resolution environments with many fewer users, like in-flight entertainment, where the shortness of the high-speed interconnect would justify a higher bandwidth budget per user.

5.3 broadcast-quality service

For broadcast-quality service at 6Mbit/s, the total bandwidth per movie is 720Mbit/s, or 90Mbyte/s, achievable with RAID though not a single disk in 1998.

On the 32Kbit/s per user calculation, 23040 users are required to justify the cost of the bandwidth for a new movie.

To put this into perspective, a user base of 100,000 would pay for 4 movies; 10-million would pay for 400 simultaneously available movies.

5.4 HDTV service

HDTV requires 18Mbit/s per movie, or 2.1Gbit/s for 120 replications. This kind of bandwidth is achievable with a wide array of disks, or a wide DRAM (potentially affordable by the time HDTV becomes common, especially if the movie cost is shared across many subscribers).

On the 32Kbit/s per user calculation, 69120 users are required to justify each new movie. A population of 10-million would pay for 145 movies; 69,120,000 would pay for a 1000-movie library, requiring a high-speed interconnect of a total of over 2Tbit/s of bandwidth.

5.5 deployment

How technology should be rolled out is an interesting issue. Early adopters will buy something because it does something they couldn't do before, and may be enchanted by the technology, even if it has imperfections, whereas late adopters—who are the majority of the market—expect to buy something that works seemlessly, and don't care too much about the technology. The hard trick is to effect a transition between the two models [Norman 1998].

What the different levels of service offer is a way of partitioning the roll-out, so that it can be sold relatively quickly to the small market of early adopters, who will buy it even if the technology still needs some work. As the problems are ironed out and the user base grows, it becomes possible to switch to one of the higher-specification video models, in concert with a drive to sell the system to a much larger user base.

For example, if the original approach is to sell the entry-level system to about 200,000 users (a small number by consumer standards), the budget bandwidth calculation will support about 35 simultaneous movies, enough to sell the system to the enthusiast.

When the system is pitched to a wider market, it could be upgraded to broadcast standard, which would require an additional 600,000 users. At this stage, it would be a good idea to deploy the system to high-density areas where a higher bandwidth budget per user could be tolerated. Such a service would start to compare reasonably with cable: although the number of choices would be smaller, the convenience of being able to start and stop movies at will, and do fast forward and rewind, should provide sufficient value to justify the system to a wider market. As the number of users exceeded the number budgeted for the available bandwidth, the number of movies could be increased until there were sufficient to avoid short-term demand for more. At this stage, the focus could change on selling HDTV, which would require another transition of selling better service to more users without adding more movies.

As with the move to broadcast-quality, the move to HDTV-quality service could start in high-density areas where a higher bandwidth budget per user could be tolerated.

6. Conclusion

The SAVoD approach is promising in that it could scale up to millions of users, unlike other proposals where the number of users is a significant factor in both the load on the server, and on the network.

The fact that there is a roll-out model that allows the technology to be introduced in phases makes it practical to introduce without a very large amount of capital up front.

Future work on SAVoD includes further investigation of details of the architecture, including the interface to the high-speed interconnect, and the components required in the local stations. Other parts of the system are relatively simple.

Acknowledgments

This work was done while on sabbatical hosted by Trevor Mudge at Department of Electrical Engineering and Computer Science, University of Michigan. Financial support was provided by the University of the Witwatersrand and the South African Foundation for Research Development.

References

[Abram-Profeta and Shin 1997] E.L. Abram-Profeta and K. G. Shin. Scheduling Video Programs in Near Video-on-Demand Systems, *Proc. Conf. on Multimedia '97*, Seattle, November 1997, pp 359-369.

[Aggarwal *et al.* 1994] S. Aggarwal , J.A. Garay and A. Herzberg, Adaptive video on demand, *PODC '94: Proc. 13th Annual ACM Symp. on Principles of Distributed Computing*, Los Angeles, August 1994, p. 402.

[Byun and Lee 1994] J.W. Byun and T.T. Lee. The Design and Analysis of an ATM Multicast Switch with Adaptive Traffic Controller, *IEEE/ACM Transactions on Networking*, vol. 2 no. 3, June 1994, pp. 288-298.

[Chen *et al.* 1994] M.-S. Chen, D. Kandlur and P. Yu. Support for Fully Interactive Playout in Disk-Array-Based Video Server, *Proc. 2nd ACM Int. Conf. on Multimedia '94*, San Francisco, October 1994, pp. 391-398.

[Dan *et al.* 1994] A. Dan, D. Sitaram and P. Shahabuddin. Scheduling Policies for an On-Demand Video Server with Batching, *Proc. 2nd ACM Int. Conf. on Multimedia*, San Francisco, CA, October, 1994, pp. 15-23.

[Hennessy and Patterson 1996] J.L. Hennessy and D.A. Patterson. *Computer Architecture: A Quantitative Approach* (2nd ed.), Morgan Kauffmann, San Francisco, 1996.

[Jadav and Choudhary 1995] D. Jadav and A. Choudhary. Designing and Implementing High-Performance Media-on-Demand Servers, *IEEE Parallel & Distributed Technology*, vol. 3 no. 2, Summer 1995, pp. 29-39.

[Norman 1998] D. A. Norman. *The Invisible Computer*, MT Press, Cambridge, MA, 1998.

[Taylor *et al.* 1995] H. Taylor, D. Chin, S. Knight and J. Kaba. The Magic Video-on-Demand Server and Real-Time Simulation System, *IEEE Parallel & Distributed Technology*, vol. 3 no. 2, Summer 1995, pp. 40-51.