IPMI Member Newsletter

Published for the Members of the IP Multicast Initiative. **August 1998**

New Member Corner

PSINet is one of the largest and most experienced Internet service providers in the world. With more than 30,000 corporate customers in the United States and abroad, our solutions have been proven again and again in demanding real-world environments -- airlines and financial services, media and manufacturing, retail, energy, telecommunications, and more.

Our services include on-demand and dedicated Internet connectivity, remote access for mobile workers, corporate Intranet and security services, and Web site hosting and development. We deliver these solutions over an owned-and-operated Frame Relay network, complete with value-added services like 24-hour network monitoring, outstanding customer support, and free training.

PSINet is a growing presence in the international business community, with offices in eight countries in North America, Asia, and Europe. Our clients include some of the largest corporations in the world, as well as thousands of small and mid-sized businesses.

ViewPoint

The Broadcast.com IPO

INTERNET IPO INSANITY CONTINUES

The Internet stock frenzy continued unabated Friday. Broadcast.com (BCST), an unprofitable and little-known aggregator of Internet audio sites, became America's best-ever initial public offering when its shares opened at 68 1/4, more than 279% above the offering price of $18.

From ZDNET, Monday July 20, 1998

The Company believes that to be a successful Internet broadcaster it also must successfully deploy multicasting or a similar broadcasting technology that can deliver streaming media content to many users simultaneously through one-to-many Internet connections.

From BROADCAST.COM's IPO Prospectus (SEC S-1/A filing on July 16th)

BROADCAST.COM is an internet broadcasting/content aggregation company. Many people are interested in the business models and justifications for IP **Multicast** deployment so following this IPO, it's interesting to dissect the company's SEC filings and consider its business model and dependency on IP **Multicast-enabled** infrastructure. The company prospectus makes many references to **multicasting,** especially in the risk factors section.

Here are some extracts:

1. Factors that may affect the Company's quarterly operating results include ...
(xii) the cost to acquire sufficient bandwidth or to integrate efficient broadcast technologies, such as **multicasting**, to meet the Company's needs,
(xiii) the mix of unicasting and **multicasting** from the Company's Web sites (see "--Scalability of Number of Users

BROADCAST on page 6

IPMI Wrap Up

The Internet Bandwidth Management Summit has a new handle- **IBAND 98!** It's still November 15 - 17, 1998 at the DoubleTree Hotel in San Jose.

The second **interoperability test event** is set to follow IBAND, 18 - 20 November, 1998 at the DoubleTree Hotel. Exhibit space is going fast. Call Suzanne Brock at 408.879.8080.

Guest ViewPoint - Joe Touch, USC/ISI describes an interesting use of IP Multicast. Page 2

ViewPoint - Martin Hall, Stardust Forums CTO, gives his viewpoint about Broadcast.com going public and its effect on Multicast deployment. Page 1

IPMI@NAB rescheduled. The Member Meeting scheduled for this month in Washington D.C. has been postponed. With all the activity and interest between the IPMI & NAB, we would like to reschedule a meeting at NAB HQ!

Stardust to host panel at NAB Radio. NAB Radio produces the worlds largest gathering of radio professionals. The 1998 NAB Radio Show will be held in Seattle, Washington, October 14 - 17. Information about IPMI & IP Multicasting will be presented by Martin Hall & Dan Fortune.

TWG meeting - back by popular demand! The **Technology Working Group** is meeting this month, 8/12. Regular meetings are planned and a schedule will be discussed at the meeting.

Check out the **IPMI Calendar** on page 8.

Guest ViewPoint

Multicast web push using interest-groups the LSAM Proxy Cache (LPC)

By Joe Touch
USC/ISI
(touch@isi.edu)

1. Preface

The LSAM project at USC/Information Sciences Institute (ISI) is developing an open system for scalable multicast web services, called the LSAM Proxy Cache (LPC). The LPC is a multicast proxy cache, adapted from the Apache proxy cache[1]. It uses the AFDP reliable multicast transport protocol for file transfer[2], although the interface would support any command-line accessible equivalent. The system currently runs on FreeBSD 2.2.x, and is designed to provide a self-adapting multicast push of web content based on "popular page groups". It provides a transparent automatic system, or can be controlled via web pages both at the client and server. It can also be accessed remotely, allowing external systems, such as group work management and AI-based access analysis, to control the multicast. This document presents a brief summary of that architecture, outlined in detail in [6], and focuses on the multicast aspects of the system.

2. Avoiding network gaps

Caches are used to make data at a distant server appear close to the client, by keeping a local copy nearer the client. A cache can be used by a single client, where data is reused. We refer to this as temporal caching. A cache can allow a number of clients to use each others' responses; we refer to this as ensemble caching. The greatest benefits in web caches have been in ensemble caching, where proxy caches are used to serve a group of clients, a shared proxy cache (Figure 1).
The difficulty with ensemble caches is that they work only when they are near to all the clients, and far from the server. There are many examples of data sets for which the client set is dispersed, e.g., the Olympic Games, the Academy Awards, and national elections.

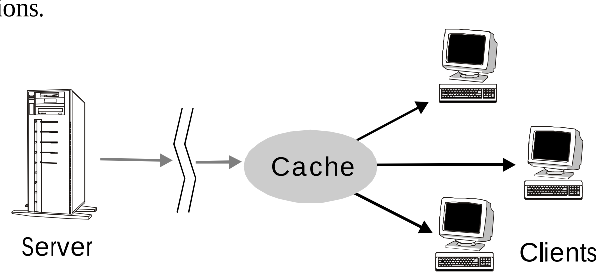

Figure 1. Shared proxy cache

In the LPC, a set of caches, called filters, emulates the behavior of this single, shared cache. Another cache, known as a pump, nearer the data, pushes data to the set of filters, which together effectively form a single, virtual cache (Figure 2). A request travels from a client up to the filter, to the pump.

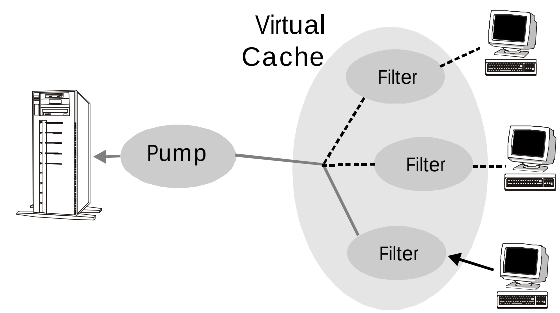

Figure 2. Request travels through virtual cache, to pump

The first response to a request is multicast to the members of the virtual cache, the set of filter caches (Figure 3). This response is forwarded by the filter where the request originated, back to the client.

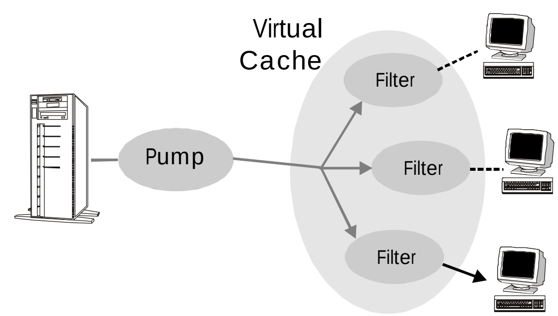

Figure 3. First response ismulticast to the group of filter caches

Subsequent requests are served locally, from the copies in the filter caches. The filter caches give the behavior of a single, shared proxy cache, even where no such centralized proxy could exist. The LPC is, in effect, a distributed shared proxy cache.

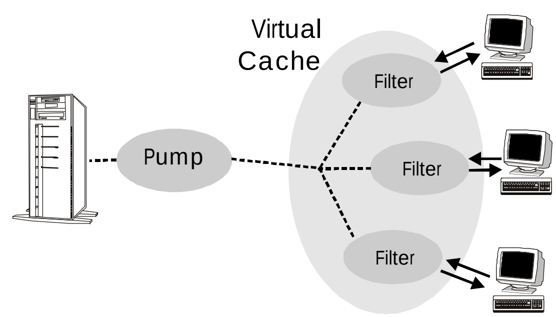

Figure 4. Subsequent responses are served by the local filter caches

VIEWPOINT on page 3

VIEWPOINT from page 2

The LPC relies on Apache's use of the file system for caching. When a filter listens to a channel, files are received directly into the caching directory of the filter proxy. A modified Redirect is used to cause the filter to re-check its local cache; this m-Redirect is returned by the pump after the multicast has been sent to the entire group of filters (Figure 5, step #6). As a result, a single multicast can be used both to pre-load the neighbor filters and to give the immediate response to the waiting client.

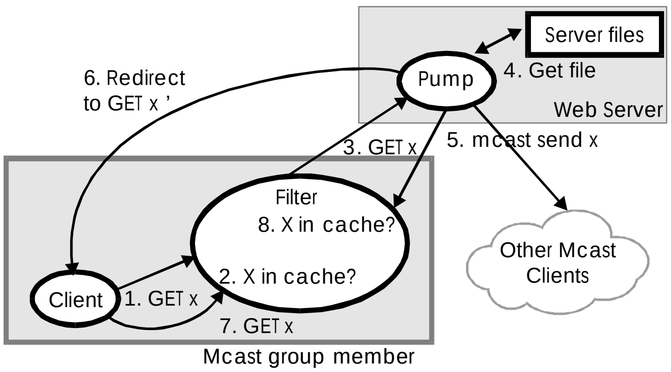

Figure 5. Walk-through of a multicast push

3. Self-organization

An important design feature of the LPC system is that these pumps and filters are actually the same device, a modified Apache proxy cache. These caches are placed throughout a network, and act as filters or pumps where necessary. A proxy becomes a pump when it notices that some subset of its web pages are "popular enough", usually automatically by analyzing its logs, though the LPC supports manual (web forms-based) and remote programmatic control. A proxy becomes a filter when it notices popularity in its request stream.
The LPC uses a two-level multicast announcement system (Figure 6), similar to that used in *sd* (the multimedia teleconferencing management tool) [4]. There is a single, global multicast announcement channel on a well-known address, and new pump channels are advertised on this channel. The TTL for the announcement is related to the TTLs of the incoming requests that, over time, caused the log entries that made that channel popular. If lots of local clients ask for USC pages, the USC channel is announced with a small TTL; the Olympics are likely to be announced with a much larger, global scope.
The announcements indicate the address and content of the other channels, created by individual pumps. A filter listens to the announcements to determine when a relevant channel exists, and uses the address information to configure a partition of its cache to listen for files.

This system of automatic filter joins is self-organizing, so that the filters join the channels at network aggregation points. The LPC assumes that the filters are organized in a hierarchy, using the conventional proxy redirection mechanism. An automated routing protocol allows a set of LPCs to organize themselves

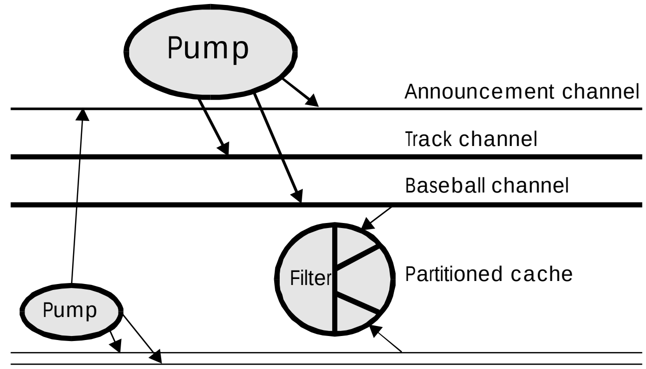

Figure 6. Multicast channels in the LSAM system

this way, without much external intervention. Only the proxy nearest the network egress (e.g., ISP) need be 'flagged' by the network administrator.
The remaining proxies form a tree, rooted at this egress point. There may be many of these trees, within each organization. The egress proxies may be manually configured to be part of another tree, e.g., within the ISP. The result is a set of hierarchies for the unicast queries and responses. The hierarchy provides a way to aggregate requests.
Consider a tree, as in Figure 7. Upstream request load is indicated by the thickness of the lines connecting the proxies. The request load shown here is all related to a single, advertised group. Midway up the tree, some of the proxies see enough requests to warrant joining the related multicast channel, so they do. These proxies are shown filled-in. The result is that a proxy, and typically all its upstream neighbors, join the group.

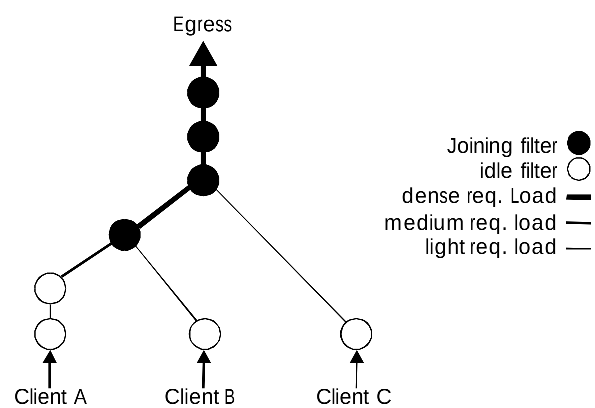

Figure 7. Filters join where traffic is dense enough

At this point, all these 'joining filters' are receiving multicast pre-loads from the pump, for pages related to this channel. Responses that come from clients A and B are served locally, from their nearest filter; other requests continue to travel up the tree, e.g., for client C. Because, in this case, clients A and B are responsible for the bulk of the request traffic related to this channel, most of the upstream proxies will cease to see further requests. They automatically leave the channel, since they no longer have sufficient traffic to warrant staying tuned-in. The

VIEWPOINT on page 4

VIEWPOINT from page 3

result is a lone proxy, at the natural network aggregation point (Figure 8).

There are a few interesting properties to this self-organizing feature. First, the behavior is emergent, a function of the local join/leave rules of the proxies, not any global communication. Second, it aggregates exactly at the network aggregation points. These are the places below which unicast is more effective, and above which multicast is more effective. Which leads us to our third point that the LPC is thus an effective unification of unicast and multicast techniques, each for its own strengths, where each makes sense.

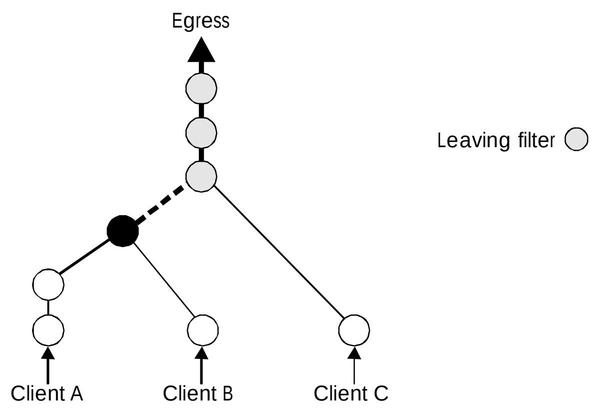

Figure 8. Further upstream filters leave

There are a wide variety of both research and commercial development of web cache systems. The LSAM proxy's main distinction is its use of multicast push to reduce the first-hit cost of retrieval throughout the system. It is based on source preloading of a receiver cache, a multicast version of an earlier unicast scheme [5]. It anticipates requests of individual clients by multicasting pages to the channel. Other related systems are discussed in greater detail in [6].

The current implementation of the LPC, v0.8, supports multicast push for a number of channels, based on manual configuration or scripted program control. In this release, the automated control is limited to 'suggest' manual actions (via highlighting on the control page). The final release, available at the end of August 1998, contains automated channel control for both pumps and filters. It supports dynamic auto-configuration of the unicast proxy hierarchy, which can be exported to other proxy caches and clients. Six different cache replacement algorithms have been implemented, selected in the configuration file at proxy boot time. Several different object scheduling mechanisms have also been implemented, and compared in network-limited and processor-limited environments. This release, v0.8, is currently available on the LSAM web pages [3]. A demonstration of the web page interface is available therein.

The current system has been demonstrated in a lab, using artificial bandwidth limiters and delay inducers. A demonstration is also available, implemented in the *ns* network simulation tool. In both cases, client access is equivalent to a local cache hit, even for pages not yet accessed locally.

4. Additional mechanisms for supporting multicast web push

The channel mechanism represents the bulk of the primary functions of the LPC; in it, servers create and delete channels, and push files, and clients tune and de-tune channels, and receive files. The bulk of the implementation lies elsewhere, in support for these mechanisms.

The LPC relies on automatic detection of interesting channels (groups of web content), both at the client and the server, and files to push (at the server). This relies on on-line analysis of the Apache logs, indicating what files and file prefixes have been most popular over varying windows of time. This analysis must be efficient and incrementally updated.

The cache was modified to support partitions, where each partition corresponds to a channel. The Apache cache was chosen for the LPC because it relies on a disk-based directory for its hash table. Various ways of partitioning it included replicating the directory for each partition, and creating a separate set of directories of hard links into the main hash. Each partition is capable of enforcing a separate space limit, using a separate replacement algorithm. This would allow replacement policies to reflect the needs of the partition: a stock market partition to use the "oldest first" replacement, whereas an image partition might prefer a "largest first" policy.

The LPC assumes the unicast proxy redirection is deep, from the client to the network egress point. There are several problems to be overcome here. First, users are unlikely to participate in the management of a deep proxy hierarchy. To that end, the LPC has implemented an automatic proxy hierarchy configuration, which also uses multicast to find the nearest neighbor. The proxy nearer the egress point remains as found; the other proxy redirects to the first. Egress proxies are 'anointed' by the network administrator, but no other configuration of the system is required.

The resulting hierarchy is likely to be somewhat deep. Recent analysis of Squid cache performance indicate that deep proxy hierarchies can lead to unacceptable forwarding delays [7]. The request is stored at each proxy hop in the path, and checked against its cache (and its siblings, in the case of ICP) before proceeding to the next hop. These delays can be overcome by using an analogy to hardware caching, used for memory lockups. In hardware, it is common to check an address in the cache and main memory in parallel; the first response is used, the second (typically the main memory one) is aborted while still pending. In the web, this corresponds to having the first hop in the path generate a parallel request to the true URL source, bypassing the proxy path. Whichever path responds first is used, the other (typically path through the remaining proxies) is aborted (Figure 9).

The LPC uses this cut-through to reduce the penalty of a deep proxy hierarchy. Cut-through requires additional mechanisms to avoid overloading servers, however. Web caching provides three primary benefits: it reduces response latency, and reduces both server and network load. Without additional mechanisms, cut-

VIEWPOINT on page 5

VIEWPOINT from page 4

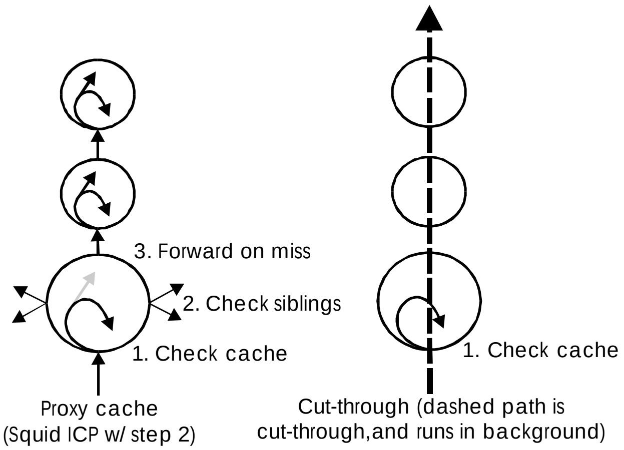

Figure 9. Store-and-forward checking vs. cut-through

through would defeat these load reduction benefits. The LPC treats cut-through requests as lower priority than non cut-through request. This requires scheduling support in the proxy and the server, as well, as in the network, such that background (cut-through) traffic is ignored (or dropped) in preference to foreground (non cut-through). This foreground/background mechanism here applies to unicast traffic, because the request and responses are unicast. It also applies to multicast traffic, in a slightly different way.

Pump rules

There is a set of rules that govern how a server responds to a request, that depend on whether the server is currently announcing a related channel, whether the request is part of that channel, and whether the client has already tuned into the channel. These rules govern the use of both unicast and multicast foreground/background, and ensure timely response with a minimum of network disruption:

Legend:
FG = foreground
BG = background
Italics = condition
Bold = action

1. An FG-GET is generated when all proxies on the path from client to server fail to 'hit' on a request one of three cases applies:

   1.a. *A request IS in a channel that the pump is serving*
   AND client IS tuned to that channel
   AND response HAS NOT recently been sent on that channel

   There are two different ways to handle this. The first relies on the multicast channel to send the response to the request, which could be problematic (requires FG multicast):
   stall the response to the GET until multicast finishes
   multicast an FG response to the channel
   respond to the GET with an m-REDIRECT unicast to the client

   The second variant is the same as the next case, having the advantage of using only BG multicast:
   unicast a FG response to the client
   multicast a BG response to the channel

   1.b. *A request IS in a channel that the pump is serving*
   AND client IS NOT tuned to that channel
   AND response HAS NOT recently been sent on that channel

   unicast a FG response to the client
   multicast a BG response to the station

   1.c ALL OTHER CASES, i.e.,
   if response IS NOT tuned to the channel
   OR response HAS recently been sent on that channel

   unicast a FG response to the client

2. A BG GET is generated when exactly one intermediate proxy decides, in which case:

   2.a *if response IS in a channel that the pump is serving*
   AND client IS tuned to that channel
   *AND response HAS NOT been recently sent to the chan*nel

   multicast a BG response to the channel

   2.b *ELSE*

   unicast a BG response to the client

5. Summary

The LPC shows one way in which multicast can be used to provide enhanced capability for web caching. LPC has been optimized for a particular domain, of sets of popular web pages. In the process of designing and implementing the LPC, several issues have been raised regarding the use of multicast for web access. Notably, multicast requires special attention to the use of foreground vs. background network and server capacity, because in our case it involves speculative push of files. It also requires attention to the overall routing of requests, to provide a natural way to combine the unicast and multicast delivery mechanisms into an effective, seamless whole.

6. References

[1] Apache HTTP Server Project,
http://www.apache.org
[2] J. Cooperstock,., S. Kotsopoulos, "Why use a fishing line when you have a net? an Adaptive Multicast data Distribution protocol," Usenix '96 Proceedings.
http://www.ecf.utoronto.ca/afdp/
[3] LSAM proxy release, v0.8, August 1998

VIEWPOINT on page 6

VIEWPOINT from page 5

http://www.isi.edu/lsam/proxy/

[4] *sd* tool

ftp://ftp.ee.lbl.gov/conferencing/sd/

[5] J. Touch, "Defining `High Speed' Protocols: Five Challenges & an Example That Survives the Challenges," IEEE JSAC., special issue on Applications Enabling Gigabit Networks, Vol. 13, No. 5, June 1995, pp. 828-835.

http://www.isi.edu/touch/pubs/jsac95.html

[6] J. Touch and A. Hughes, "The LSAM Proxy Cache - a Multicast Distributed Virtual Cache," to appear in Computer Networks and ISDN Systems, also in Proc. 3rd International WWW Cache Workshop, Manchester, U.K., June 15-17, 1998.

http://www.isi.edu/touch/pubs/wc98/

[7] D. Wessels, K. Claffy, ICP and the Squid Web Cache (August 13, 1997),

Http://www.nlanr.net/%7ewessels/Papers/icp-squid.ps

This work is supported by the Defense Advanced Research Projects Agency through FBI contract #J-FBI-95-185 entitled "Large-Scale Active Middleware". The views and conclusions contained in this document are those of the author and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Department of the Army, the Defense Advanced Research Projects Agency, or the U.S. Government. Portions of this document are condensed from [6].

This project is the result of a team effort, summarized here by the project leader, Joe Touch. The team: Theodore V. Faber, Gregory G. Finn, Steve Hotz, John Heidemann, Anne Hutton;. grad students: Amy S. Hughes & Stephen Suryaputra

ViewPoint - cont.

BROADCAST from page 1

2. SCALABILITY OF NUMBERS OF USERS

The Company's success depends on its ability to broadcast audio and video programming to a large number of simultaneous users. Until recently, the Company only deployed unicasting (one user per Company originated stream) technology to broadcast audio and video programming to users over the Internet. The Company has deployed another broadcast technology, **multicasting** (multiple users per Company originated stream), on a trial basis since September 1997 and has begun to deploy this technology on a broader commercial basis only recently. The Company anticipates that unicasting will continue to be used to distribute its archived and on-demand programming and that **multicasting** or a similar broadcasting technology will be used for live and other events where a large audience for the content is expected. To increase the Company's unicast capacity, the Company will be required to successfully expand its network infrastructure through the acquisition and deployment of additional network equipment and bandwidth. There can be no assurance that the Company will be successful in such expansion. The Company believes that to be a successful Internet broadcaster it also must successfully deploy **multicasting** or a similar broadcasting technology that can deliver streaming media content to many users simultaneously through one-to-many Internet connections. To this end, the Company has deployed **multicasting**, but has not yet tested its full capacity during an actual broadcast. The Company will be required to test, deploy and successfully scale its **multicast** network infrastructure to serve mass audiences. There can be no assurance that the Company will be successful in doing so, that **multicasting** will be able to support a substantial audience or that an alternative technology will not emerge that offers superior broadcasting technology as compared to **multicasting**. In the event that **multicasting** technology is not successfully deployed in a timely manner or such an alternative technology emerges, the Company would likely be required to expend significant resources to deploya technology other than **multicasting**, which could have a material adverse effect on the Company's results of operations during the period in which the Company attempts such deployment. If the Company fails to scale its broadcasts to large audiences of simultaneous users, such failure could have a material adverse effect on the Company's business, results of operations and financial condition.

3. EXPAND NETWORK INFRASTRUCTURE

Broadcast.com intends to expand its network infrastructure through the acquisition and deployment of additional network equipment, bandwidth and broadcast scaling technologies. As part of its network expansion strategy, the Company is deploying its **multicast** network which is designed to provide streaming media content to hundreds of thousands of users simultaneously through one-to-many Internet connections. The Company has entered into agreements with over 30 ISPs and UUNET and is building the first large-scale commercial **multicast** network, which provides the Company with access to over 400,000 dial-up **multicast** ports. The Company is developing software to more efficiently handle the broadcast of hundreds of simultaneous live events and has developed proprietary software to handle broadcast blackouts, remote monitoring and remote server access. Although streaming video over the Internet does not currently offer broadcast television equivalent quality to a broad base of users, the Company believes that the quality of and demand for Internet video broadcasts will continue to improve as broadband Internet access technologies such as xDSL and cable modems become more commonly available. Further, the Company believes that video is an important and essential element in the future of Internet broadcasting. Accordingly, the broadcast.com network has been video enabled and supports multiple leading video streaming technologies including RealNetwork's RealVideo and Microsoft's NetShow.

BROADCAST on page 7

Ink Report

The IPMI Short Ink Report contains press on IP Multicast found by: IPMI staff searching the Internet, IPMI Member submissions and PR agencies. This represents only the most recent data. To view the full report, go to the IPMI web site, http://www.impulticast.com/press/ink.html/

7/15/98	America's Network	Pay Attention To Bandwidth
5/15/98	Network Computing	StarBurst Communications StarBurst Multicast
5/11/98	EE Times	News
5/8/98	PC Magazine	On the Horizon
5/5/98	PC Magazine	Multicasting Tunes In
5/4/98	Internetweek	The Fastest Networks On Earth
5/--/98	Boardwatch	Multicast Your Fate To The Wind
4/27/98	Internet Computing	IP Multicast: Real Bandwidth Relief Right Now
4/21/98	Network Computing	Multicast's Coming! It's Really Coming!
4/14/98	Internet Computing	IP Multicast Builds Momentum
4/3/98	PCWeek	StarBurst Uses MFTP for Reliable Delivery
4/2/98	Lantimes	Controlling bandwidth time
4/1/98	America's Network	Locking up new networks
4/--/98	Lantimes	Risks and rewards in videoconferencing
3/30/98	Internetweek	Blazing Fast Routers Set To Come Out Of The Gate
3/22/98	EE Times	Net Task Force Drafts TCP Upgrade For Satellite Links
3/16/98	PC Magazine	Tuning In to IP Multicasting
3/9/98	Infoworld	IP Multicast: Martin Hall, Stardust Technologies founder brings IP Multicast to market
3/5/98	PCWeek	PointCast turns to IPMulticast

ViewPoint - cont.

BROADCAST from page 6

4. TECHNOLOGY AND BANDWIDTH PROVIDERS
In order to scale with future audience growth, broadcast.com has entered into an agreement with UUNET under which the Company acquired access to 155Mbps of bandwidth for unicasting and additional bandwidth for **multicasting** on UUNET's network which can be configured to allow up to 500 simultaneous live events. In addition, UUNET is part of the Company's multicastingbuildout strategy.

5.DISTRIBUTION
Currently, the Company employs both unicasting (one user per Company originated stream) and **multicasting** (many users per Company originated stream) technologies to distribute streaming media content to users over the Internet. The Company's unicast network can provide content to tens of thousands of simultaneous users through 580multimedia servers which support multiple streaming technologies. These servers are linked through direct 45Mbps and 155Mbps connections to major Internet backbone providers including GTEI, MCI, Sprint and UUNET, which, in turn, connect to over 80% of the downstream ISPs. The Company believes that direct connections to these major backbone providers enhances the user experience by avoiding the congestion of public peering points which can cause transmission delays or packet loss. Although the Company anticipates that unicasting will remain essential for archived and on-demand applications, it believes that **multicasting,** or similar scaling technology, is essential to the future of large-scale Internet broadcasting to a mass audience. The Company believes **multicasting** is especially suited to audio and video broadcasting and will be increasingly used in the delivery of streaming media content. Currently, the Company is deploying its **multicast** network which is designed to provide streaming media content to hundreds of thousands of users simultaneously through one-to-many Internet connections. The Company has entered into agreements with over 30 ISPs and UUNET and is building the first large-scale commercial multicast network which provides the Company access toover 400,000 dial-up multicast ports.

We are planning additional work on IP Multicast-based *businessmodels as part of the suggested new IPMI programs we will be sharing with the Advisory Council over the next few weeks. In the meantime, if anyone needs contact information for BROADCAST.COM, please send email to me,* martinh@stardust.com

MARTIN HALL

IPMI Events Calendar

	August 98	September 98	October 98
IPMI Member Relations	NewsBrief (Twice weekly) IPMI Member Newsletter Technology Working Group Meeting - 12th	NewsBrief (twice weekly) IPMI Member Newsletter Advisory Council Meeting - 23rd Monthly TWG Meeting - TBD	NewsBrief IPMI Member Newsletter IPMI Member Meeting & Social @ N+I - 20th Monthly TWG Meeting - TBD
Stardust Forums Educational Events	Deployment Seminar	**The SOCKS Summit September 15-18 1998 TechMart, Santa Clara, CA**	Deployment Seminar
Industry Tradeshows ** Stardust attending*	IETF Meeting Chicago 23-28th *	ComNet, San Francisco, 9/30 - 10/3	N+I Atlanta 19 -23rd *
Publications	State-of-the-Art Reports Multicast Network Management, Monitoring & Reporting	IETF 42 Report IPMI Multicast Informational RFC	The IPMI State-of-the-Art Report calendar will be discussed at TWG meeting - Aug 12th