CHRISTIANITY TODAY INTERNATIONAL

Online Rich Media Presentations, Streaming, & Content Delivery Network

The research for this project began as a simple exploration of how to stream audio and video effectively over the Web, and whether we should host the content in-house. Several issues immediately clarified themselves.

1) **Streaming Media Issues**

Customer valuation of streaming media is extremely dependant upon the successful and rapid delivery of streamed data packets. These packets must **a)** arrive in the proper order, with **b)** few or no packets lost in transition, and with **c)** as high a quality of transmission as possible.

Typical HTML delivery suffers none of these requirements: when a Web page is delivered it is broken down into small packets of data (like streamed media is, also), but the packets may arrive in any order; whatever packets are lost may be easily re-requested with no loss of quality for the customer; and transmission of the Web pages' contents can come through as "fat" or "narrow" a bandwidth pipe as is desired. The only impact is the rapidity of delivery, but since Web pages are typically between 50-150 kilobytes in size, and since users pause several seconds before requesting subsequent pages, the two delivery experiences and requirements are not comparable.

a. To assure minimal packet loss, which is worsened as the "distance" from the server to the recipient increases, content delivery should originate from a server that is either very close to the user's point-of-presence, or the streaming server should be very close to the user's ultimate connection to the Internet Backbone. (Thus, an AT&T user would be best served by a streaming media provider that was delivering content from another AT&T server or, failing that, delivering content from a server situated close to AT&T's own backbone gateway.)

b. To assure that the streaming media packets arrive in their proper order, there should be a minimal number of "hops" between the streaming server and the user. The greater the number of servers packets must pass through to get to the user, the more likely various packets will be re-routed through other servers or suffer lag due to random network usage fluctuations. If the packets don't arrive in their proper order the user's media client must re-request packets to be re-sent by the streaming server, forcing the user to wait for the buffer to catch up or to miss the data altogether if the point where the packet belonged in the presentation is already passed. As above, these problems are minimized if the content streams from a server either located close to the user or is close to the user's network backbone.

c. A high quality of transmission includes minimal packet loss (above), proper ordering (above), high bandwidth media-types (100-300 kbps streams), minimal lag time between request and delivery (perceived "speed"), and the ability to handle multiple simultaneous users. This is best accomplished with a robust server (multiple fast CPUs, a lot of RAM, a fast hard drive), on a very fat pipe (a T3 is nice, but even more is better, since even a T3 can only accommodate 150 simultaneous users if they all request the high-bandwidth version of a video file), with redundant network connections (not just a single connection to AT&T's backbone, but also AOL's, Sprint's MCI's, and so on, as well as International backbones and network points of presence), with close proximity to networks backbone gateways, further more some sort of edge caching whereby streamed content is stored on a machine closer to the user than the originating server is desirable, though not necessary if the streaming provider has good connections with multiple backbone providers.

2) Media Type Issues

Not all media types are created equal. Based on my recent research, the current best-rated encoding format goes to the Windows Media codec for being more highly compressed (thus allowing us to stream more video information with less data), for being the least intrusive (it's already installed on all PCs, and comes with Internet Explorer for the Mac), and for playing the widest range of media types (as opposed to specializing in one codec as Real, QuickTime, and Flash do). Ultimately, the visual and audio qualities for all the major codecs and players are roughly the same under ideal circumstances. The key is to determine which media type fares best under real-world conditions.

In my analyses, Windows Media offers the best all-round user experience, despite the hype surrounding Flash video encoding and it's seemingly "instant" playability. What is not usually said about the method of Flash streaming is that the video is not buffered ahead of time, and is thus very sensitive to packet collision, packet loss, and frame skips due to packet misordering. Viewing a Flash file on a T3 line minimizes these problems, so evaluation needs to be done under typical user conditions.

Further, most Flash presentations are not truly delivered by a streaming sever: they are delivered via HTTP—which doesn't allow for the Flash client to re-request missing or misordered packets. Only VitalStreams, at the moment, offers true Flash streaming services.

QuickTime and Real lose out, in my opinion, due to the tendency for these players to be more intrusive, to require more explicit user engagement in downloading the client, or associated plug-ins.

Straight MPEG encoding is not yet recommended, in my mind, since players that natively handle MPEG streams are not yet widespread enough to make a good user experience.

In the end the actual video and audio quality of the various major codecs is minimal enough that the decision here needs to be based on the end-user experience. While my recommendation is for the Windows Media codec, we also should keep in mind that we may want to change codecs in the future, for a variety of reasons, so I believe we should ultimately be "agnostic" about the media type and choose business solutions here that allow for long-term flexibility. In other words, let's make a cost effective choice, for now, but not get locked into a particular media-type mindset.

3) Interface Design Issues

In addition to choosing a content distribution network (or streaming provider) and choosing the codec for video encoding, we need to address, separately, the best approach for designing the user interface for our online training courses, and how best to implement that interface. There are several issues that come to mind here:

a. We don't yet have a clear picture of exactly what features the user interface should and should not have

b. We don't yet have a prototype in mind that captures these features in a way that will help us arrive at a cost for creating the interface.

c. While the prototyping and creation of the initial interface needs to be done by a third-party firm at the moment, we should consider intermediate or long-term ownership of the source code. This should include not only the source code of the interface itself, but also of the underlying applications that are driving it. If it's not possible to own the underlying applications (such as is the case with underlying structure of a production system at an ASP like Nine Systems/Accordent, Harmony, etc.), we should at least own the code for the interface.

d. We should keep in mind the long term maintainability of the user's visual interface: we need to choose a solution that doesn't undermine our long-term technology strategy here at CTI. In order to maintain cost effectiveness, we are consolidating our technology expertise onto as few platforms as possible. For this reason, developing a Flash-based GUI increases long-term costs because we must train or hire a Flash developer in order to take over long-term maintenance.

Interface Design issues are largely driven by the purpose of the content and the intended market. If an e-learning solution is what we are driving for, this will dramatically impact the complexity of both the design and the site architecture, and should change the focus of research. However, if we are driving toward a Rich Media delivery system that can double as a training tool, this simplifies things greatly.

4) Production Issues

We need to be very sensitive about choosing a system that has been in active production for at least half a year or more, to already have the bugs and "kinks" worked out of the system. We are considering 1) testing new product to 2) an unknown market with 3) unclear cost and production factors. It would seem to be wise to avoid 4) paying for the development of an interface, structure, architecture, and infrastructure that is equally untested. We should consider only those solutions which have some maturity and market testing behind them at this stage—at the very least we should not fund somebody else's ground-floor development

Based on what I've seen to date, the most well-developed systems always begin at the front end—because this is what sells to customers. However, successful and cost-effective implementation of an end-to-end production-to-delivery system needs to have a mature infrastructure (operating system, reporting, delivery methods) as well as a proven production system. Many companies will show us great user interfaces but will have anemic or nonexistent production interfaces. Partly this is due to maturity in the market, but I suspect it's often true for job security: the companies want our ongoing our production fees.

Not only should we avoid unnecessary production costs associated with outright fees, we should also avoid the hidden costs of laborious and time-consuming production procedures that could be streamlined with better tools. If those tools are not present and accounted for during this round of evaluation, they will not be mature enough to be useful when we launch.

5) Reporting Issues

A truly robust and mature system—whether we're talking about a delivery system, or the learning system—will provide extensive reporting tools. Especially since this is a fledgling market and is unproven here at CT, we need to be sure that, at launch, we have all the business information at our disposal that we can get so that we can fine tune content, structure, and fees right from the very start based on actual usage. However, if reporting structures are weak, we won't have the information we need to make critical decisions in the early days, and we may wind up committing to an inefficient and expensive course of action for too long. For example, suppose we found out that users were opening but never finishing any courses that had streaming video in it, whereas courses with audio only were not only selling as well as video courses, but were being completed twice as often? Wouldn't that impact our business strategy? However, without the kind of utilization reporting this implies, we'll never be able to make that decision.

Further, if there are charges associated with open-ended items such as disk space utilization, bandwidth utilization, simultaneous users, number of users, etc., then we need to have some sort of threshold notification (i.e., send me an email when I reach 95% of account capacity, etc.), or we need to be able to monitor such items in real-time.

6) Miscellaneous

a. **Scalable:** the systems we choose should be scalable, and we should be allowed to enter into contract at the low end of the scale, before we determine whether these projects will actually work

b. **Flexible:** The systems we choose should allow us to own the media we use, should allow us to change media types if we desire, should allow us access to the source code of the design (at the very least), and allow us full ownership of user data.

c. **Unsullied:** The systems we choose should not be hosting or party to hosting pornographic or other morally offensive content.

d. **Portable:** The systems we choose should be portable across platforms and exportable to DVD or CD.

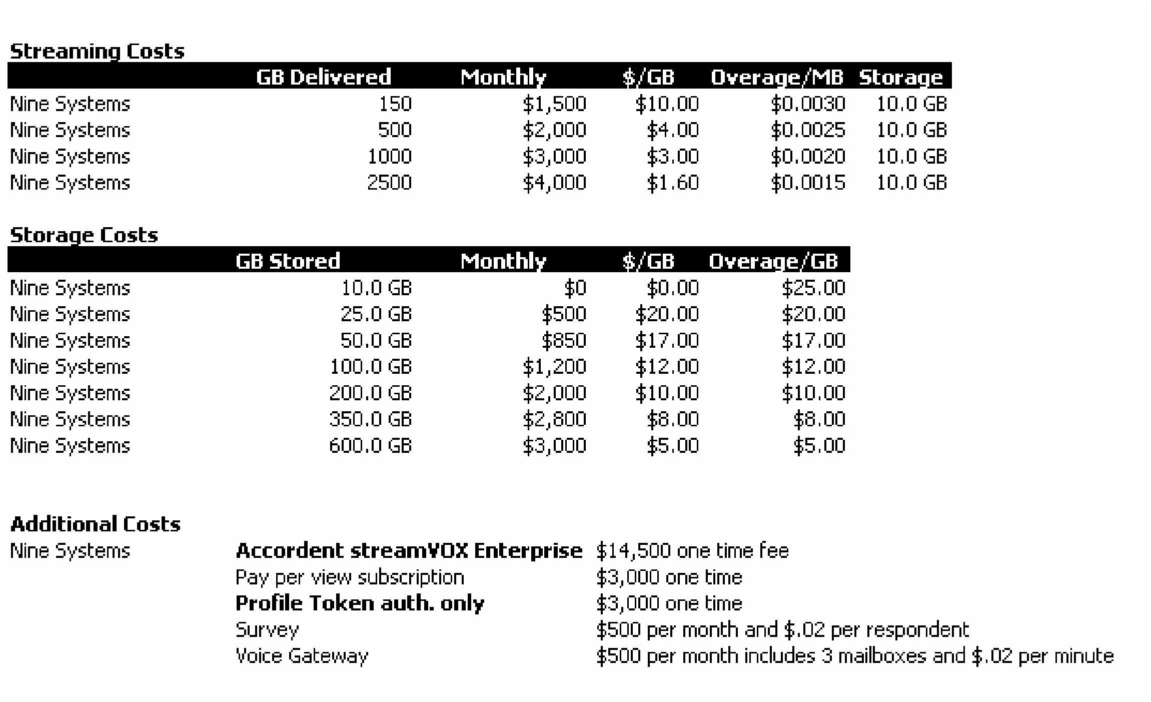
e. **Live-Broadcast Enabled:** I think we should also keep in eye on the long-term potential of needing to help a content partner webcast a seminar, convention, or class in real-time.

Recommendation

In light of the above discussion, I am leaning toward recommending Nine Systems as a content distribution partner to host the streaming content. I would also recommend Accordent as the interface/Rich Media partner for helping us develop the content and providing us the production tools to make it happen.

However, I am not as strong on Accordent as I am Nine Systems. I think that we could use virtually any content production/e-learning firm as long as we could stream the content from elsewhere.

Costs



Streaming Costs

	GB Delivered	Monthly	$/GB	Overage/MB	Storage
Nine Systems	150	$1,500	$10.00	$0.0030	10.0 GB
Nine Systems	500	$2,000	$4.00	$0.0025	10.0 GB
Nine Systems	1000	$3,000	$3.00	$0.0020	10.0 GB
Nine Systems	2500	$4,000	$1.60	$0.0015	10.0 GB

Storage Costs

	GB Stored	Monthly	$/GB	Overage/GB
Nine Systems	10.0 GB	$0	$0.00	$25.00
Nine Systems	25.0 GB	$500	$20.00	$20.00
Nine Systems	50.0 GB	$850	$17.00	$17.00
Nine Systems	100.0 GB	$1,200	$12.00	$12.00
Nine Systems	200.0 GB	$2,000	$10.00	$10.00
Nine Systems	350.0 GB	$2,800	$8.00	$8.00
Nine Systems	600.0 GB	$3,000	$5.00	$5.00

Additional Costs

Nine Systems	**Accordent streamVOX Enterprise**	$14,500 one time fee
	Pay per view subscription	$3,000 one time
	Profile Token auth. only	$3,000 one time
	Survey	$500 per month and $.02 per respondent
	Voice Gateway	$500 per month includes 3 mailboxes and $.02 per minute