VOD SERVERS – EQUATIONS AND SOLUTIONS

Glen Hardin[1] and W. Paul Sherer[2]
[1]Time Warner Cable, [2]Arroyo Video Solutions

Abstract

Video-On-Demand (VOD) is now a widely deployed product with a ready audience. No longer a "trial" product, it is a cornerstone offering for the cable industry - generating revenues, reducing churn and setting MSOs solution apart from satellite.

Yet the technology underpinning VOD services is still in its infancy, and, as new VOD services are developed, the VOD infrastructure must continue to evolve if the potential of these new services is to be realized to its fullest.

This paper seeks to provide context for VOD server technology - where it has been, where it is and where it might be going. This discussion is presented in context of the changing VOD server equation. Understanding this equation is paramount to understanding the solution going forward.

THE ORIGINAL EQUATION: CONTENT + STREAM = VOD

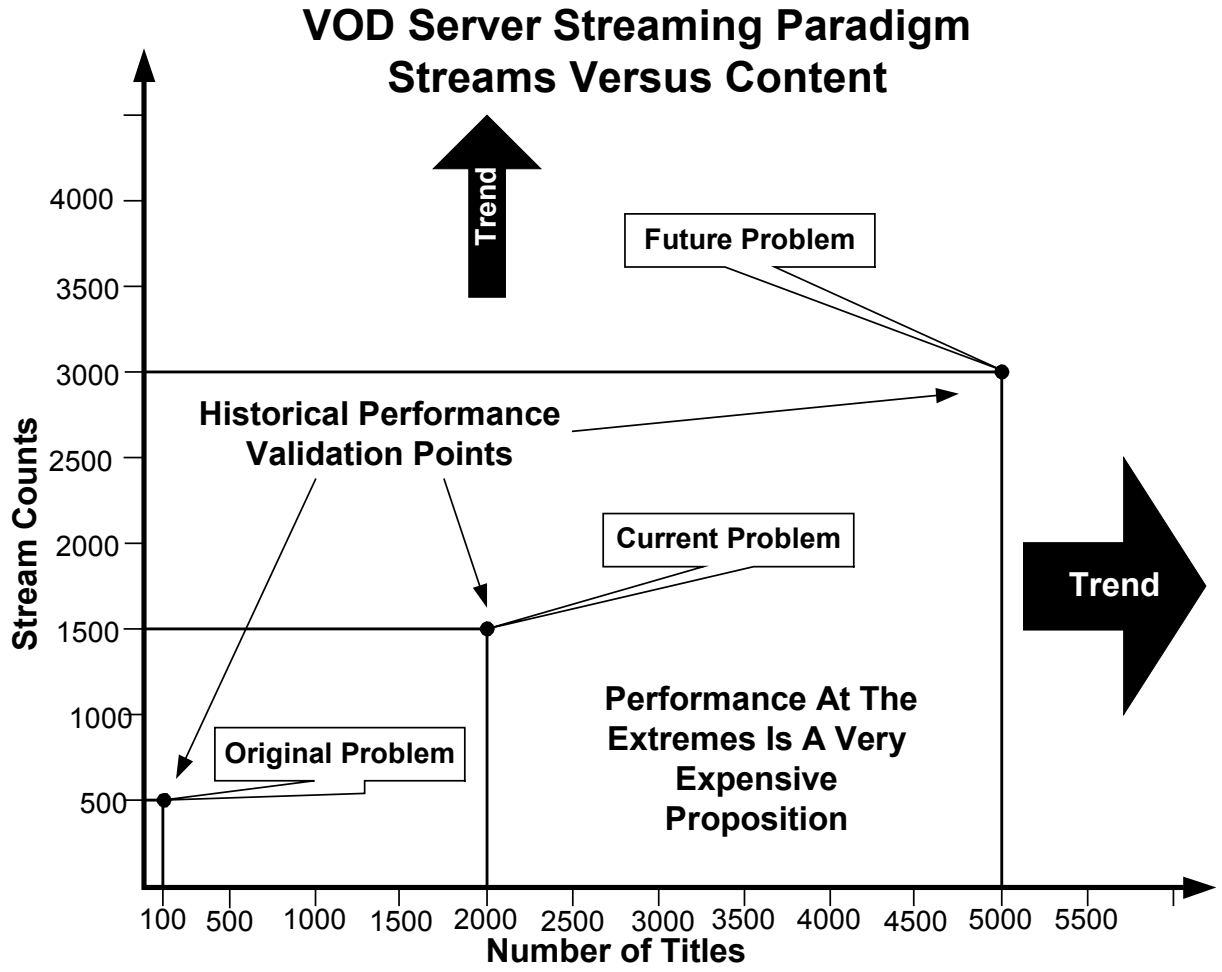
Historically there have been two basic variables to the VOD equation: the content variable and the streaming variable. Each VOD server solution has attempted to understand and resolve the relationship between content and streaming. All vendors in the marketplace work to optimize performance and price as they tackle the basic problem of how to access the stored content, transfer it across the bus architecture and pump it out of the video server without interruptions. Some do it with brute force and others with complex elegance. At first glance, it is seemingly a basic problem to solve, but, as the multipliers in front of each of these variables scale independently and infinitely, the solution quickly becomes complex. The physical solution to the equation can be based in either proprietary or commodity hardware and bonded together with plenty of custom software

Content:

The old real-estate adage "location, location, location" has its analogy in VOD and it is "content, content, content". Content is the main driver for the success of VOD. Add additional compelling content and the stream use rates will increase.

The amount (and type – High Def content is 4 times as resource consuming as Standard Def) of content drives the total amount of storage the system requires. In the original VOD services, the content variable was limited to the top 100 "hit" titles – requiring perhaps 250 content hours of storage. As VOD technologies proved themselves, new services such as subscription video–on-demand were added and the total number of storage hours grew to support them. The hours of storage grew from a few hundred hours to 800 hours. With the increase in the number of subscription services and recent new services such as Free-On-Demand, Music-On-Demand and High-Definition-On-Demand (HDVOD), the storage requirement quickly has quickly grown to thousands of hours.

Depending on a server's architecture, scaling the content storage may be as easy as adding more drives or an additional disk array to the system or as complicated as replacing all the drives in the system. The one thing that is for sure, if the VOD service is to be successful, the multiplier to content variable can go in only one direction, ever increasing.

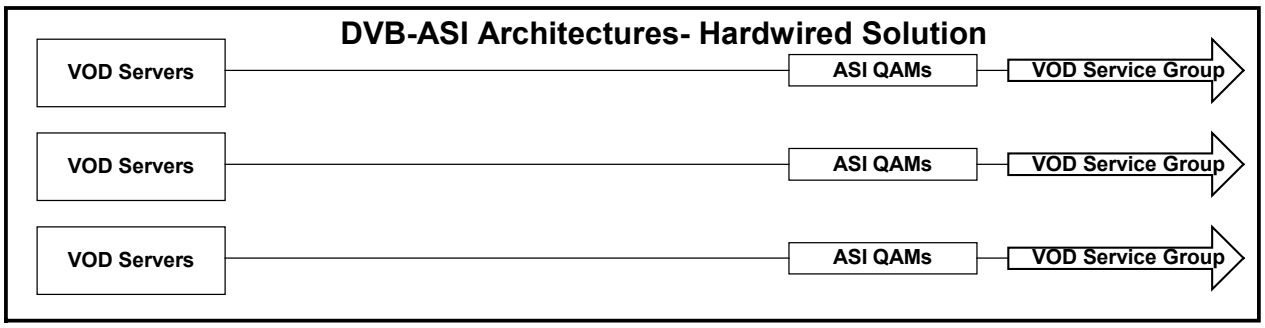


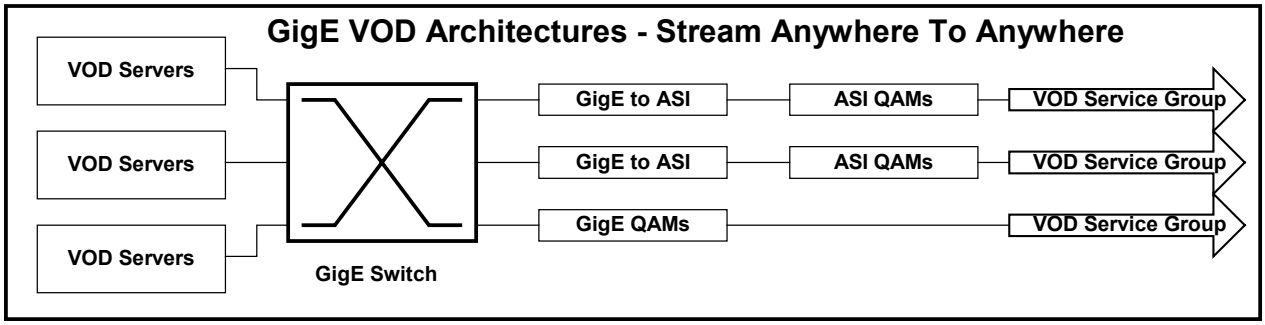
Streaming:

Streaming needs to access the content stored on disk and route it across a bus or interconnect and pump it out of the server. Since the sole purpose of streaming is to deliver content, all other functions may need to operate at a lower priority including the reception of new content. Content delivery is everything and it must be accomplished flawlessly without interruptions.

Scaling streaming is a very complicated proposition and different vendors have approached the problem in different ways. Historically, VOD server vendors relied on core disk Input/Output (I/O) subsystem performance to attain their stream performance. Some vendors chose to implement complex interconnect and RAID architectures to gain the efficiencies of parallelism and thereby increase streaming performance, while others scaled through simple server replication. In either case, validating a server's streaming performance was accomplished by taking a single piece of content and streaming it out at the server's

max stream capacity (the easy way) and taking a unique piece of content per unique stream to the server's max capacity (a much harder problem to solve). Testing at both extremes guaranteed that the server could deliver the content in any way the customer could ever order it. This performance at both extremes came at a relatively high price, but was reasonable with a relatively small amount of content.





THE NEW EQUATION: INGEST + CONTENT + STREAM = ON DEMAND SERVICES

As discussed above, server architectures have historically focused on optimizing the output capabilities of their servers at the expense of their input capabilities. However, increasingly a new factor is changing the original server performance equation. The new factor is ingest.

Ingest:

Servers are increasingly required to receive MPEG files in real-time, ingest them into the VOD server platform, and immediately (within seconds) allow all customers to stream them. This rapid increase in ingest requirements is a natural outgrowth of the increase in content offered in VOD form, but it also seems to be a universal in the various next generation On-Demand services under development - including Network Personal Video Recording (NPVR), broadcast "Start-Over" and client applications like Weather-On-Demand.

These real-time acquisition-based services greatly impact VOD servers and in multiple ways. Content storage requirements are growing tremendously as the number of networks offering On Demand content grows. Instead of supporting 1200 titles, the VOD servers increasingly need to support multiples of that number. Streaming is also impacted, both because the wealth of new content must be written to non-volatile storage (i.e. disk), and because of the increase in the quantity of streams as subscribers access the new content. Additionally, since the quality of service must be maintained both for content ingest

and for streaming, VOD servers will have to work within even tighter performance tolerances as both these variables scale. This equation is far more complicated equation than what was originally required in the early days of VOD.

Architecturally, real-time acquisition-based services favor more centralized content storage solutions that allow single ingest points to serve all customers. The ingest server must have interconnectivity to all service groups. Supporting such features in highly distributed server architectures is overly complicated and almost impossible.

With the broader width of content offering and the advances in parallel technologies, the current VOD server architecture paradigm needs to be re-examined. Furthermore, the market now has the historical experience to evaluate the necessary performance requirements against the usage patterns of the On-Demand-Services offered.

REMOVING THE COMPLEXITY FROM THE VIDEO SERVER

Advancement in other technologies, including software technologies, has allowed the complexity of the VOD server to be simplified.

ASI to GigE

The most important shift in complexity of the VOD server was the removal of the DVB-ASI interface and replacement with the GigE interface. As a result, the VOD vendor no longer had to develop and support custom DVB-ASI cards within the server, which was a huge cost reduction for the server companies. VOD servers with DVB-ASI also require the video server's streaming capacity to be in parity with the edge capacity as the video servers are physically tied to the edge devices. This shift to GigE also reduced the barriers to entry, allowing new vendors and innovation into this market.

The Edge

As a result of the shift to GigE within the video server, the requirement and costs for DVB-ASI moved further out into the network. To maintain compatibility with the existing QAM devices, new devices were developed to translate the GigE back to ASI to interface to the existing QAMs. Now, native GigE interfaces are available from every QAM manufacturer, negating the requirement to convert the GigE signals back into ASI prior to the QAM. This will further reduce the cost and complexity of the VOD solution.

Transport

Advancements within the transport technologies have greatly facilitated the shift from highly distributed VOD architectures to more centralized architectures. Transport technologies have gone from inefficient ASI transports to single GigE pipe on a pair of fibers to 40 times 1G, 40 times 2.5G and finally 40 times10G on a single pair of fibers. Highly distributed architectures also required multiple instances of storage arrays and copies of the content. Centralizing storage and/or the servers has the added benefit of allowing for greater efficiencies through sharing of the storage arrays across many streaming devices. As a result, fewer storage arrays are required as fewer copies of the content are needed.

Core Switch

The most desirable and advantageous method of connecting the GigE video server into the cable plant is through a GigE core switch. Advancements in switching

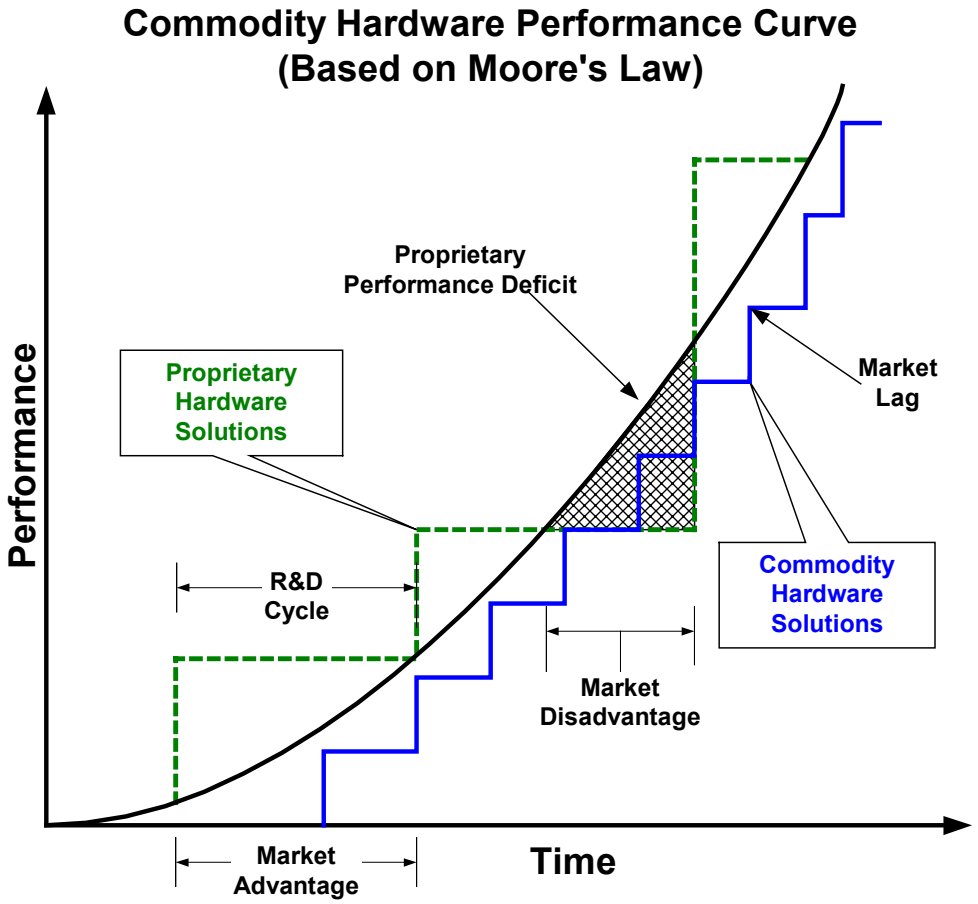
technologies now allow for fully meshed non-blocking delivery of video.

The integration of a core switch enables video servers to stream from anywhere to anywhere. That is to say, any video server streaming port can service every VOD service group. Since content is no longer tied directly to storage at the edge, but is now centrally available to all streaming devices, the content is no longer bound to a particular video server component. The core switch also has the added benefit of reducing the overall streaming requirement of the server. The server's streaming capacity no longer needs to be in parity with the edge QAM capacity, but only with the max stream utilization. This allows the video server to scale independently from the edge QAM capacity. For example, a DVB-ASI server with the capacity of 3000 streams serving a given customer base can now be served by a 2000 stream GigE server with the same blocking factor.

Furthermore, the addition of the core switch between the VOD server platform and the plant also minimize the requirement to develop methods of interconnecting various discrete storage arrays though some backend back-end switching fabric.

Software Infrastructure

Finally, advancements in open software standards that allow interoperability between VOD vendors have greatly influenced the marketplace. MSO's are no longer held captive to a particular vendor once the initial purchase is made. Each time the system expands or major features are added and new server capacity is required the MSO can



choose the best of breed among the vendors. This ensures that VOD vendors remain competitive in terms of price and performance.

PROPIETARY HARDWARE SOLUTIONS VS. COMMODITY

The graphic attempts to illustrate the relationship between video server performance and commodity hardware performance based on Moore's Law, an industry-accepted concept that hardware performance will double every 18 to 24 months. In general, the hardware commodity performance curve increases due to parallel advancements in all the technologies within the PC market: faster and multiple processor machines and bus infrastructures, faster and denser DRAM, drive technology, and network interfaces.

The increased performance described above results in several secondary benefits for the constrained environment of the cable headend: less space, power, cooling, and wiring are required. The newer solutions are much more dense and efficient in terms of Mbps per rack unit and the number of Mbps per watt of power consumed. Less power consumed infers lower cooling requirements. Additionally, as the outputs of the video server become denser, fewer wires are required to integrate these servers into the plant. It has been demonstrated that a server of 5000 streams @ 3.75 Mbps can now be wired into the plant with just a couple of 10 G interconnects. Historically, this interconnect would require upwards of 31 DVB-ASI wires or even 21 GigE connections. Less wiring substantially simplifies the integration work.

Both proprietary and commodity server vendors try to optimize their server costs because the stream price is determined by competition within the market. A vendor can only control its server costs. If, for a given set of hardware, a video server can sustain n number of streams, then the minimum per stream cost equals $ / # streams per unit of hardware (not including the cost of development for the necessary software and other associated costs). Regardless of the hardware solution, the software is the valuable component of intellectual property of any vendor.

It is prophesized that this curve cannot sustain its exponential growth forever but in the near-term it provides guidance and insight into the future capabilities of the market.

Proprietary Hardware Solutions:

Many VOD server vendors have developed a proprietary solution by creating and integrating custom hardware components and/or custom interconnection technology. If accomplished effectively, the resultant solution should outperform what is available in the commodity market using a similar generation of technology.

The difference between the performance of proprietary solution and the commodity curve determines the performance advantage of the proprietary solution. The performance advantage translates into a market advantage for a period of time until the commodity curve catches up with the proprietary performance. Server companies offering proprietary solutions must exploit this finite time of market advantage through sales to recoup their investment in hardware R&D. At the same time they must also continue to invest in the next generation server solution lest they fall below the commodity performance curve. It is a never-ending race to stay ahead of the commodity curve and a risky business proposition. It is easy for vendors with proprietary solutions to fall

below the commodity performance curve if they do not carefully time their adoption of the newer higher performing hardware. There is a high cost to develop performance gains above the commodity performance curve leading to expensive R&D cycles. Those R&D costs must be re-cooped before commodity performance catches up, otherwise, sales opportunities will evaporate as the performance advantage disappears. In sum, it is possible to develop a proprietary solution that exceeds the commodity curve, and the more is invested the longer this advantage will remain. However, the commodity market has proven time and time again that the commodity curve will eventually catch up regardless of the technology.

Since MSOs cannot be expected to perform forklift upgrades enthusiastically or frequently, proprietary hardware solutions also face the challenge of integrating newer higher performing hardware into an existing lower performing solution. Typically, proprietary solutions rely on symmetric server performance with all machines within a server complex operating at the same performance level. But it does not make sense to integrate new high performing hardware and operate it only at the existing performance levels. Therefore in addition to constant efforts to keep up with the commodity hardware curve, vendors of proprietary solutions must undertake the development of many lines of custom code in order to have older and newer generation hardware interoperate (if at all possible) at their respective levels of performance as one integrated seamless solution.

Hence, even in proprietary hardware solutions the software is as important as the hardware and is, in fact, the key intellectual property within the VOD server platform.

Commodity Hardware Solutions:

In the emerging VOD industry back in the early 90's, the raw commodity server market barely was able to eke out enough performance from a given platform to justify the costs of VOD. The market price for streams was magnitudes higher than it is today.

There are several enabling factors that allow for commodity hardware solutions to be competitive today. First the content equation has changed dramatically as described above, i.e., 10,000 hours of content versus the historical 100 hours of content. Second, the base hardware available in the commodity market has the necessary off-the-shelf performance required to deliver dense VOD streaming. Video servers supporting multiple GigE and 10 GigE pipes per two or three rack units. And finally, the MSO market has accepted the premise of caching based on its historical content use patterns and the cost/performance trade-offs associated with cache-based servers.

VOD vendors with architectures based on commodity off-the-shelf servers abstract the hardware solution from the software solution and, at a minimum, develop loosely coupled systems. The VOD delivery solution is software–based, which makes the hardware choice an independent decision. As such, the vendor is able to choose the best-of-breed within the commodity market.

A potential drawback to working solely with commodity hardware is that the performance of commodity platforms must lag slightly the commodity performance curve, due to the need to re-qualify new hardware platforms as they become available in the market. Best-in-class solutions abstract out the software from the hardware allowing

performance to more closely follow the commodity performance curve.

Vendors using commodity servers must, like those with proprietary architectures, address the problem of integrating advancements in hardware into their solution. To address this issue, a commodity based solution needs to be developed in a manner that supports asymmetric server performance within the solution.

By achieving solution independence from the underlying hardware, commodity vendors allow MSO's to utilize existing procurement and maintenance contracts for the underlying hardware. This allows the MSO to leverage its volume purchase agreements with commodity hardware vendors. Further, internal expertise can be more readily leveraged across hardware platforms which are used for multiple service solutions.

Another reason to focus on commodity hardware is enhancing the utility of the intellectual property which must be created by the vendor. Simply put software intellectual property is readily reapplied across multiple generations of underlying commodity hardware with little to no redevelopment or retraining across generations. This allows the commodity vendor to focus on continual enhancement of system robustness and functionality without the need for continual investment in long lead time hardware development cycles in order to keep pace with the performance created by the overall computing market.

All servers, even the historical servers, stream from RAM. The difference with the new caching servers is that the RAM is used to capture the "working set" of the cache. That is, the set of content which is active at this given moment and likely to be active in the near future. The goal is to minimize the size of the working set so the minimal amount of expensive components may be utilized to achieve the desired level of performance.

CACHING SERVERS

Overall caching server vendors try to optimize the right mix of components and costs to get the greatest return on performance

Caching components vs. Costs
RAM
~$350 / GB
High Performance Drives
~$5 / GB
Standard Performance Drives
~$1 / GB

When a caching architecture is part of any system be it a microprocessor or a VOD server one of the first questions which must be asked is how to determine what to keep in the cache and for how long. All such systems use a mixture of predictive and reactive algorithms to decide what to cache.

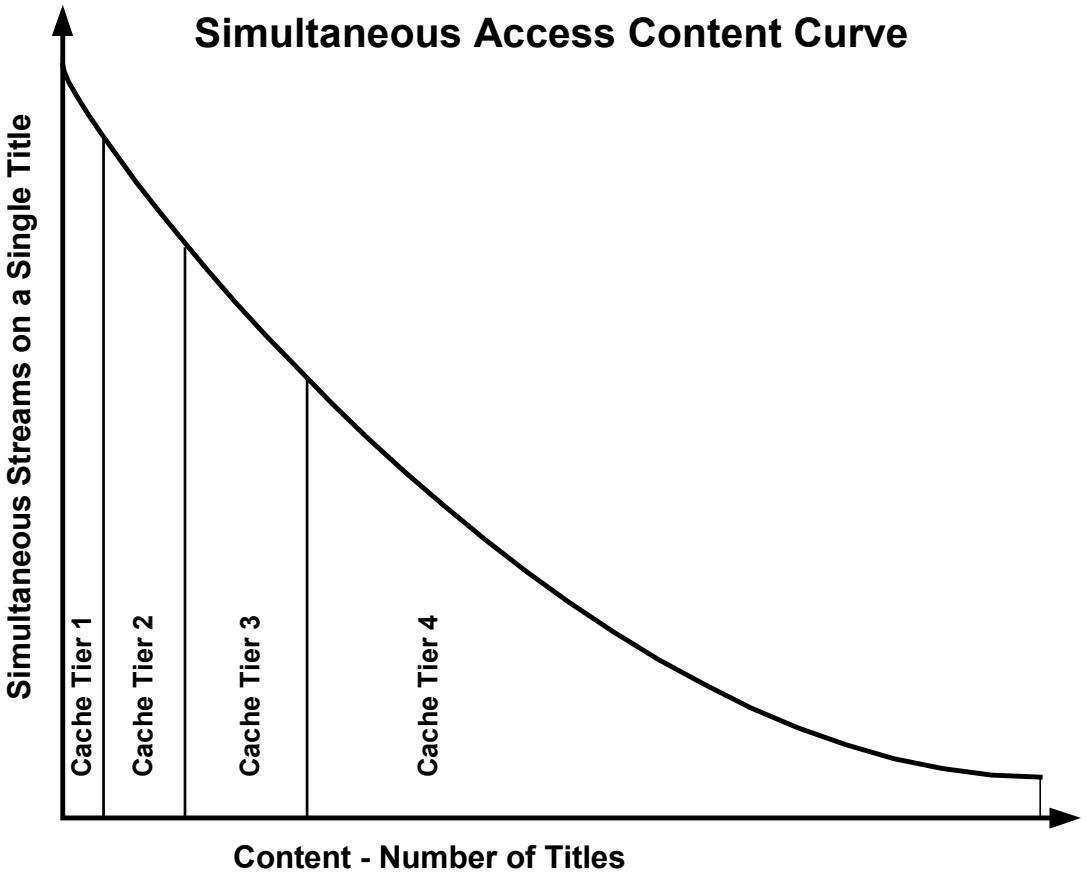
The most common predictive algorithm is the "next obvious thing". That is based on whatever is happening now the next obvious thing will most likely happen next. In a microprocessor, this usually means the next instruction after the current one – in a VOD server this usually means the frame after this one. Some seek to predict events at a much higher level. In a microprocessor this might be to predict which program will be run or in a VOD server which title will be played. The problem with this approach is the decision at this level depends on factors which are beyond reasonable prediction a priori. Johnny

Carson can die and the "Best of Carson" titles can suddenly become very popular. Janet Jackson can suffer a wardrobe

malfunction and a particular sequence from the Super Bowl can see extremely high utilization.

Most effective caching systems do not rely on accurate prediction at this level but rather rely on reactive algorithms. The approach is to make observations at a slightly broader level than the low level predictions described earlier. But to then assume that these higher level decisions will be tend to be self-similar. That is, for example, if 50 of the last 100 plays have been for a certain sequence of a Super Bowl then is it is likely that many of the upcoming plays will be for the same sequence. In this way a cache can adjust quickly to decisions which it cannot predict accurately beforehand but it can observe accurately and react to.

Traditional VOD servers tended to treat a piece of content as whole but the "wardrobe malfunction" example illustrates a portion of a piece of content may have radically different usage patterns associated with it than other portions. A more efficient cache can recognize that this portion has radically different usage and treat it differently than the rest of that piece of content. These usage patterns can happen for many reasons in addition to an event in the content. New forms of navigation such as chaptering can allow entry into a piece on content at a set of locations. The chapters in effect become mini-pieces of content with a larger whole. Another trend is the creation of virtual assets. These are logical assets composed of components from other assets which are perceived as single asset by someone viewing



them. This could be an edited topical news update or a play list of music videos.

Another reason to no longer view a piece of content as whole is responsiveness. Even if a user has held a bookmark for resume for a long enough time that the local cache has flushed the content, it is desirable that the VOD system should be able to "resume" very quickly and easily. To do this, the caching system should retrieve content starting from the point where the resume occurred, rather than from the opening scenes of the piece of content.

As has been noted earlier, VOD is now a cornerstone revenue generating service. As such it must be robust and available 24 X 7. This means MSO's should look to vendors to provide automatic resiliency to system faults and to allow for maintenance and upgrade without service outage.

One important consideration is the unit of failure for which the system is resilient. When considering the failure modes which must be compensated for, most would think some hardware fault such a network interface failure. In reality, for all types of video server, whether based on proprietary hardware or commodity hardware, the most common failure mode is a failure in the software not a failure in the hardware. So in this sense the most common unit of failure must be considered to be the server itself. This means the entire server function must be recoverable automatically. That is, the current workload must be recovered intact by other systems without the need for human intervention. This level of resilience has been applied to telephony and data applications but is just now being designed into VOD servers.

In terms of content availability, while there is a definite trend to centralized storage, many MSO's are now considering geographic resiliency in their system planning. Centralized but in at least two locations and interconnected through switching and transport. The idea is that the content storage must survive a natural disaster such as a hurricane or tornado or a manmade problem such as a portion of the power grid going offline. By having content stored in geographically diverse location the odds that such an event would take two or more facilities offline is greatly reduced compared the odds of a single facility going offline.

For many years the resiliency of content was assured via RAID 5 technology. With costs of modern disks becoming so low, in many situations it is simply more cost effective to keep multiple copies. The issue with RAID 5 resiliency structures is that there is an assumed extremely high bandwidth path among the components of a RAID 5 structure. This is reasonably easy to achieve among components in single system but becomes increasingly onerous when resiliency is spread across many systems. In a RAID 5 system of n components when a failure occurs, all n-1 remaining components must participate in recovering the lost information - which must be regenerated through computation. The bandwidth impact of this process will often make it impractical particularly across geographically diverse content storage facilities.

All of the above is leading to the creation of caching tiers – each with a role to fulfill. The exact boundaries of these tiers will to large degree be determined by the cost and reliability of transport between the tiers.

Level 1 – The Edge

This is the tier closest to the service groups. The content kept here will be fairly

active and will have fairly high reuse. The two more expensive caching components of RAM and high performance disk drives will be used in this tier. The key here is to capture the "working set" with the minimum amount storage capacity. The goal of this tier will normally be to satisfy 90 to 95% of the stream requests within this tier to provide sub-second responsiveness. However, while handling the bulk of the stream, this tier would have very little of the overall cache storage capacity - typically only a few percent. This creates a very efficient usage of the high cost caching components in this tier.

Level 2 – Local Storage

The tier behind the edge serves to decouple the higher instantaneous performance of the edge from the much more modest performance of the local library. This tier can be seen as a performance matching tier which uses greater cache storage capacity to only allow a small number of the total stream requests to have impact on the local library. At this level the storage is still viewed primarily as cache with the implicit assumption that if need be content can always retrieved from the local library and resiliency of content is less important.

Level 3 – Local Library

This tier is the demarcation of the relatively inexpensive and readily available local transport to the relatively expensive and scarce long haul transport. This content has high resiliency so that single failures of devices or servers can be handled without requiring retrieval from the regional or national library. If the area served by the local library is large or prone to problems such as hurricanes the storage may be implemented with geographic resiliency. The percentage of all content accessible from the associated edge systems is very high. Because of the large amount of storage lower cost storage components are used here. It would be expected for every stream play request which accessed the regional or national library that thousands or tens of thousands of stream play requests would have been seen by the edge systems.

Level 4 – Regional or National Library

This tier is the ultimate source of content available to any edge system. All content available to any edge system is resiliently stored somewhere in the library. This tier will have geographic resiliency and multiple points of ingest. The regional or national library tier has the greatest storage capacity and the greatest resiliency of all the tiers.

NEW TESTING PARADIGMS

The advent and adoption of new caching server architectures requires a re-examining of how servers are tested and qualified. The old method of validating a server's streaming performance by taking single piece of content and streaming it out at the server's max stream capacity and by taking a unique piece of content per unique stream up to the server's max will not result in the desired and practical price and performance point. The historical usage patterns must be applied to the testing and validation of the new caching servers.

A new term needs to be defined to help normalize the validation of servers. The term Cache Gain represents the additional steaming capacity above what is available through the core disk I/O subsystem performance. For example, if a given server has a disk I/O performance of 1000 unique streams but can deliver not only the 1000 unique content streams from disk but an additional 500 duplicate content streams from cache, the server would demonstrate a Cache Gain of 50 percent. In that same example, if

a server were able to deliver 1000 duplicate content streams from cache then the Cache Gain would be 100 percent.

Since caching servers vary greatly in the number of cache tiers and performance within the tiers, setting simple easy standards of performance is difficult. The process of normalizing the system performance to core disk I/O performance provides a baseline from which to work.

CONCLUSION

The jury is still out. Although all the best and brightest within the VOD server community agree that there are cache gains to be made as of now, there is not enough empirical data of cache effectiveness to unequivocally say what exactly what the cache gains are for a given type of content and service.

What is a believable cache gain? Is it 10%, 100% or 1000%? Only careful monitoring of live systems in the field will prove out the actual achievable gains.

However, this architectural advance clearly represents the next step in the evolution of the infrastructure for on demand services. With the advent of this architecture the stage is set for a plethora of new services reliant on much greater breadth of content and much more dynamic usage. “Start Over” and network PVR fit well to this architecture. More applications will come.

One can now see the infrastructure coming into being which will enable the efficient delivery of a fully personalized entertainment to every MSO customer on every television.