



XDT's catapult® explained

Introduction

catapult was designed specifically for the shared storage of uncompressed film frames. Unlike other storage technologies that have a more general-purpose heritage, catapult exploits the sequential nature of film and the high throughput capability of today's computing platforms.

Preconceptions originating from existing SAN, NAS and Network technologies often cause difficulties in understanding catapult's unique advantages, and how it can compliment and extend the investment that facilities have already made into their existing SAN and NAS infrastructure.

Uncompressed film frames are still the standard for maintaining quality and compatibility between systems. Whether these frames originate from scanned film, 3D rendering or are some other intermediate, they are large and cumbersome. Time consumed by data wrangling alone is a serious expense. Compression technologies are emerging, though benefits seem limited as frames need to be decompressed for VFX and compositing, and at very least for compatibility.

catapult provides mechanisms to minimise the number of times frames need to be copied in a DI pipeline. The ultimate long-term goal is to reduce the number of copies to zero, except where a copy may be made as part of the frames processing (e.g. rendering composites).

This document has been designed to help understand what catapult is, how catapult is different to existing storage solutions, and how catapult can make the DI process more efficient. I hope that you find it informative.

Gavin Stewart

Head of Research and Development, XDT.

catapult facts

catapult's efficiency extends from dealing with uncompressed frames as sequences. SAN and NAS storage solutions are required to work at the block level (many blocks per file). This microscopic view cannot provide the same efficiency. Efficiency, in this case, being the concurrent streaming IO per disk. e.g. Two concurrent 2K streams are easily supplied by catapult with 16 disks in RAID-5. An equivalent SAN filesystem, to cope with the random head seeks, would need closer to 100 disks.

Although catapult can utilise InfiniBand, it is not a requirement. Gigabit Ethernet (GigE) is very cheap, readily available, and may be used to stream real-time uncompressed, full frame 2K over three cables. catapult uses TCP/IP and is able to saturate multiple GigE connections to their theoretical maximum¹.

The maximum streaming speed for delivery of uncompressed film sequences is 1.5 Gigabytes/sec when utilising multiple 10GigE or 20Gig InfiniBand. To provide a wide

¹ See <http://www.xdt.com.au/Products/catapult/> for graphs.



safety margin, and for purposes where the slowest locations of the disk platters are in use, the system is rated for 2 x 4:3 2K 10bit sustained streams, or approximately 600 Megabytes/sec.

catapult only deals with frame sequences, and only “talks” the catapult protocol, this keeps efficiency high, but limits connectivity to catapult-enabled client software. Note, however, that catapult bridge extends the catapult protocol to common CIFS and NFS clients. (i.e. Windows, Mac and Unix.)

ballista[®], included with the purchase of catapult, provides the same high performance Ethernet streaming technology to copy files between any two network connected systems. This allows high-speed point-to-point file replication between systems with fast local storage (direct attached or SAN). ballista provides a subset of catapult's functionality, but may be used to provide catapult protocol services on any compatible system running Linux.

catapult and SAN shared storage

SANs with SAN file-systems are the current top performing shared storage systems in use today. Compatible with multiple Operating Systems, scalable to large capacities, and capable of high levels of streaming throughput, SANs initially appear to be the only solution necessary. Unfortunately SANs are terribly inefficient at handling concurrent streams of sequential frames. A large investment is made in high-performance storage and high-throughput network fabric infrastructure, yet only a small fraction of it's potential is ever realised.

To use the ever popular car analogy: It's like creating a courier company from a fleet of 1,000 mph Ferrari's, and your own roads, only to find congestion at the loading depot for two or more simultaneous Ferrari's is slowing the average speed down to 100 mph. The solution is scalable, so purchasing more vehicles and loading depots will increase total performance almost linearly, although overall efficiency remains low.

SANs use hard disk drives with mechanical heads, just as any disk based storage system. To maximise streaming read efficiency, frames must be copied to the SAN in the correct order. This minimises disk head movement when reading the sequence in that same order. When multiple sequences are read at the same time, the disk head must move to read a subset of blocks for one frame in one sequence, before being repositioned to read a subset of blocks from another frame in another sequence. This block-level view of data causes the disk heads to spend more time seeking than reading, this is the underlying cause of the gross inefficiency of SANs.

In contrast, catapult understands frames at a sequence level. When reading sequences concurrently, many frames may be read for the same sequence before the head needs be repositioned for an alternate sequence. Very little time is wasted due to disk head movement, efficiency is kept high.

catapult is not intended to replace SANs in a facility, it's specialised design is not as a general purpose storage system. However, catapult can make an existing SAN investment stretch further by offloading work that couldn't be offloaded previously. Incoming frame sequences, such as from film scanners and render farms, can easily be reviewed, in real-



time, from catapult over common Gigabit Ethernet networks. Utilising slingshot, frames can later be copied, in order, over multiple Ethernet interfaces to and from SAN connected systems, at SAN speeds.

By extending SAN level performance into existing Ethernet networks, and providing mechanisms to offload unnecessary IO from SANs, catapult can greatly improve and extend the efficiency of existing SAN infrastructure in a facility.

catapult and NAS shared storage

NASs are the workhorse of storage systems. Immediately usable by Windows, Mac and Unix clients without additional software or HCAs, NASs are the default choice for general purpose storage.

While the same hard disk drive inefficiencies apply to NASs as they did for SANs, other limitations appear much sooner when using NASs for high throughput or high IO workloads.

No single data transfer between NAS and client may exceed the speed of the connecting Ethernet pathway. Ethernet link aggregation (a.k.a channel bonding or trunking) does not increase throughput to a client with more than one Ethernet interface. However, it does allow multiple client connections to be spread over several NAS server Ethernet interfaces.

Network filesystem (CIFS, NFS) overheads add latency to the data exchange, and throughput is typically limited to less than 50% of the available bandwidth of GigE at best. In real-world terms, a file copy at 50MB/s over is considered to be very good, even though the underlying connection is capable of twice that. SAN filesystems were designed to overcome this very issue, with an additional cost of specialist hardware and software for each client.

Extending the car analogy: Your fleet now uses common roads that have a speed limit of 100 mph, but averaging traffic rules only allow you to reach 50 mph. More roads may be added to loading depots to increase aggregate throughput, but any single destination is still limited to 50 mph.

The catapult protocol allows catapult to utilise all available bandwidth² in a single GigE channel, and by using multiple channels simultaneously to a single client, speeds of up to 600 MB/s are possible.

catapult bridge and rendering

catapult is fast, it can be copied to and from and provides real-time review. So how can a render-farm write a frame sequence in order without copying ?

² Minus ethernet frame and TCP/IP header overhead.



catapult bridge[®] appears as a NAS to CIFS and NFS clients, only with reordered write-back, and frame level read-ahead caching. Low latency and high throughput non-volatile RAM provide SAN-like IOPS³.

As frames are rendered to catapult bridge, emerging (sub)sequences are identified, and committed back to catapult once reaching a sufficient length. This occurs while the render process is taking place, and committed subsequences may be reviewed immediately. The typical file fragmentation and out of order frames associated with rendered frame sequences are no longer an issue.

Frame data read through catapult bridge is cached along with frames included by reading ahead into the frame sequence. This function also extends to existing NAS systems, so as to offload render-farm read load to catapult bridge. i.e. textures, models and other meta data referenced by the rendering process are cached onto catapult bridge from an existing NAS, thereby offloading IOPS from that NAS. Multiple catapult bridge systems may be used depending on render-farm size and workload characteristics.

catapult is protected from the “chatty” and random-IO nature of network filesystems, allowing it to remain dedicated to handling frame streams efficiently.

Further Information

Diagrams, flash media and other documents are available on the XDT website (<http://www.xdt.com.au>). Please visit for further updates on catapult technology.

³ Non-volatile RAM hardware specification rates as 100,000 IOPS