

HP StorageWorks XP1 2000 Disk Array best practices for performance guidelines white paper



- Executive summary..... 2
- Peak performance expectations 2
- A general guideline 3
- Overview..... 4
 - Traffic isolation 4
 - Hardware 4
- First best practices 5
- Best practices to optimize performance 5
- Back-end processing 6
- Disk performance..... 8
- Workload balance and contention management of the back-end..... 9
- RAID Group type has an effect 9
- Disk emulation type has an impact..... 13
- ACP considerations..... 13
- Back-end rules of thumb..... 15
- Front-end processing 15
 - Client Host Interface Processors (CHiPs) 15
- Replication software and CHiP processors 17
- Cache and shared memory..... 18
 - Crossbar switch (CSW)..... 18
 - Cache capacity and optional cache boards 18
- Shared memory 20
- Front-end rules of thumb..... 20
- Glossary..... 21
- For more information..... 22

Executive summary

The HP StorageWorks XP12000 Disk Array has the highest availability of any previous generation XP array and clearly outperforms the competition. The long-established design principles for the StorageWorks XP systems are, in priority order, to ensure uncompromising reliability and availability to provide a storage platform enabling advanced features and functions for the most demanding storage area network (SAN) environment workloads.

When configuring to optimize performance, the XP12000 Disk Array has differences from previous XP arrays that must be considered. The XP12000 Disk Array is architecturally different than the HP StorageWorks XP1024 Disk Array. The differences are not obvious from the outside. The way the disk drives are connected to the ACP controllers is different, cache slot size has increased, and the learning algorithms have been improved.

If you need the maximum in connectivity, availability, reliability, and features then follow the recommendations in this paper to get the most performance from the XP12000 Disk Array.

Before reading this document, be sure you understand the architecture or have read the HP StorageWorks XP12000 owner's guide at

http://h20000.www2.hp.com/bizsupport/TechSupport/DocumentIndex.jsp?contentType=SupportManual&locale=en_US&docIndexId=179911&taskId=101&prodTypeId=12169&prodSeriesId=436460.

Peak performance expectations

Several important basic measurements of disk array performance include:

1. The amount of time it takes for an array to respond to a single request for I/O or a READ operation, from cache or from disk
2. The amount of time it takes for an array to record or WRITE a single request to cache and then to disk
3. The aggregate number of READ operations the array can perform in a stated time period (typically one second)
4. The aggregate number of WRITE operations the array can perform in a stated time period
5. The maximum amount of data the array can READ from disk, if the data is ideally or sequentially arranged on disk, measured over a fairly long time period, but often reported over a one second period
6. The maximum amount of data the array can WRITE to disk, if the data is presented to the array in an ideal fashion as large blocks to be written sequentially to disk

There are other measurements, such as peak sequential transfer rate from cache, that are of lesser interest, which are not discussed in this paper.

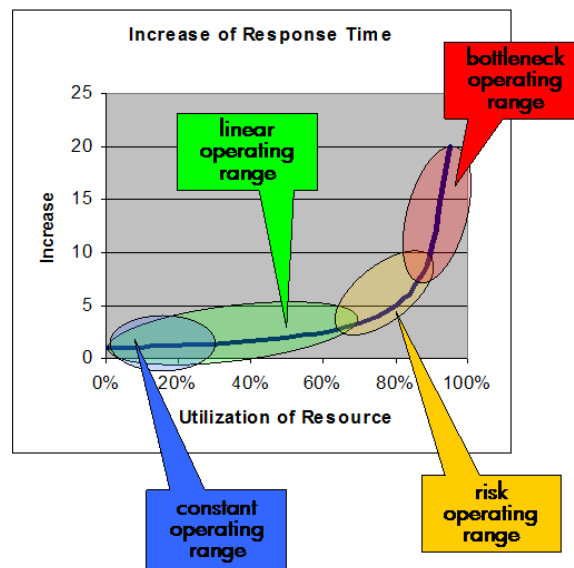
A common misconception about array performance has to do with items 1 and 2 from the preceding list. The amount of time required for a single read or write are typically bounded by performance of host bus adapters (HBAs), memory technology, interface speed, I/O processing code path length, and basic disk performance. In moving from one array (XP1024 Disk Array) to another array (XP12000 Disk Array), these performance bounds do not change much. In other words, the time required to do a single read or write action with an XP1024 Disk Array will be about the same as the time required performing a similar action on an XP12000 Disk Array. The XP12000 Disk Array will not be twice as fast as an XP1024 Disk Array on a single read or write. In fact, it will be about the same. As the number of reads and writes increases, the aggregate number of reads and writes that can be performed at peak rates with fully configured systems does increase. This aggregate number of peak rates is what is referenced in the marketing literature for the XP12000 Disk Array.

To achieve peak performance with measurements 3, 4, 5 and 6, the XP arrays need to be configured at or near its maximum capacity. As the configuration is completed with the maximum number of front end processors, or CHIP processors and Array Control Processors, or ACPs, then the systems become capable of achieving these peak performance numbers.

A general guideline

For reliable day-to-day operation, the system will typically not be operated under peak performance level. The following model may help to define an operating level that is suitable to achieve reliable service levels. The behavior of a storage system and its components can be approximated by the models from queuing theory.

Figure 1. Operating levels



A simple model is the M/M/1 queue, where the behavior is expressed by the following formula:

$$T_R = T_S / (1 - U)$$

Where T_R is the response time for a request, T_S is the time that the resource is busy with a request (service time), and U is the utilization of the resource (0=unutilized, 1=100% utilized). Figure 1 shows the results for $T_S = 1$.

In this curve several ranges can be distinguished. Through the constant operating range (up to 20% to 30% utilization) the response time is approximately equal to the service time. Up to approximately 70% utilization the increase of the response time stays quite flat and linear. At 50% utilization the response times are approximately twice the service time and at 70% utilization approximately three times the service time. In this range load peaks typically will not impact the overall behavior significantly. Beginning at 65% the curve becomes increasingly steeper. In this range, small changes of the utilization—which translates in small changes in the workload characteristics—can lead to significant changes in the response time of the system and the behavior becomes rather unpredictable. In this range the risk exists that service levels cannot be maintained. The final range above 90% utilization is characterized by rocketing response times without any increase in throughput.

If this model is applied to the components of the storage configuration and interpretation of the performance metrics, performance issues can be avoided. The utilization can either be determined by the ratio of the applied I/O rate divided by the maximum I/O rate for that component at the given workload or if available by the utilization metric for that resource.

From this model you can also gather that it is important to distribute the load as evenly as possible over the available resources (processors and disks). In cases where load distribution is not even, the higher loaded resources will add over proportional to the overall latency so that under the same I/O load, the average response time will be higher.

Overview



Traffic isolation

Unlike any other storage array, the XP12000 Disk Array provides excellent traffic isolation. That is, activity on one part of the array typically does not impact other areas of the array. This allows different hosts to have isolated ports. For example, it allows backup servers to use different ports than production hosts. This provides outstanding workload isolation. This traffic isolation greatly improves the predictability of I/O performance. This is extremely important for applications that require low latency for I/O operations, such as trading systems and/or heavy database queries for online transaction processing systems.

Hardware

The XP12000 Disk Array hardware consists of a disk control frame (DKC) that will hold up to 128 disks, and one to four optional disk array frames (DKUs) for the remaining disks. The DKC contains the control panel, service processor (SVP), cache memory, shared memory, crossbar switches, batteries, client-host interface processors (CHiPs), and array control processors (ACPs). The DKUs hold the majority of the disk drives and can hold up to 256 physical disks each. Unlike the XP1024 Disk Array, the XP12000 Disk Array can be used with only the center cabinet. Up to four disk cabinets can be added to each XP12000 Disk Array.

Although the XP12000 Disk Array is designed to work in a wide range of configurations, configurations options can influence performance, usable capacity, or availability. With the necessary information about the configuration options for the XP12000 Disk Array, a storage administrator can optimize its configuration for a specific application.

These configuration choices include:

- Number of disks
- Number of disk groups
- Type and number of disks in a disk group
- Emulation type
- RAID levels (1, 5, and 6)
- Replication SW (HP StorageWorks Continuous Access XP, HP StorageWorks Business Copy XP, or HP StorageWorks Snapshot XP)
- Cache capacity and optional cache boards
- Shared memory capacity and optional shared memory boards
- Number of CHiP pairs and how the ports are connected
- Number of ACP pairs
- Number of CSWs

Each of these topics is detailed in the following sections. Note that it may **not** be possible to simultaneously optimize a configuration for cost, performance, and availability. Conflicting recommendations require one objective to yield to the demands made by another. For example, RAID6 provides the highest availability from a disk failure, but is the slowest performing due to extra parity cycles. Other tradeoffs are more complex by comparison but are worth understanding because there is no best choice in all situations. “Best” depends on the priorities of a particular environment.

First best practices

The first best practices before considering performance fall into the common sense category, but are often overlooked.

- Read the XP12000 Disk Array Owner’s Guide. Always operate the array in accordance with the owner’s guide. In particular, never exceed the environmental operation requirements.
- HP continually improves the performance, reliability, and functionality of the XP12000 Disk Array. Your XP array can benefit from these investments only if it is using the latest FW.
- Deploy the array only in supported configurations. In many case, HP does not support a configuration because it failed its testing. Do not risk the availability of your critical applications to unsupported configurations.

Best practices to optimize performance

Experience shows that high and low cost generally have an inverse relationship to performance. While this continues to be true, there are ways to make tradeoffs in trying to balance between the two directions with the XP12000 Disk Array. This section outlines configuration options for optimizing performance and price-for-performance, although sometimes at the expense of cost and availability objectives.

Array performance management typically follows one of two strategies: contention management or workload distribution. Contention management is the act (or art) of isolating different performance demands to independent array resources (that is, disks, controllers, ports, and so on). The classic example of this is assigning database table space and log files to separate disks. The logic is that removing the contention between the two workloads improves the overall system performance.

The other strategy is workload distribution. In this strategy, the performance demand is evenly distributed across the widest set of array resources. The logic for this strategy is to reduce possible queuing delays by using the most resources within the array.

The preceding strategies assume a lack of intelligence in the array and a simple structure for array architecture. The HP StorageWorks XP12000 Disk Array does not fit either assumption. The XP12000 Disk Array is like having multiple arrays within an array as well as intelligence to determine the type of workload being presented to the front-end (CHiP) processors. For this reason, there will have to be a balance between the two strategies.

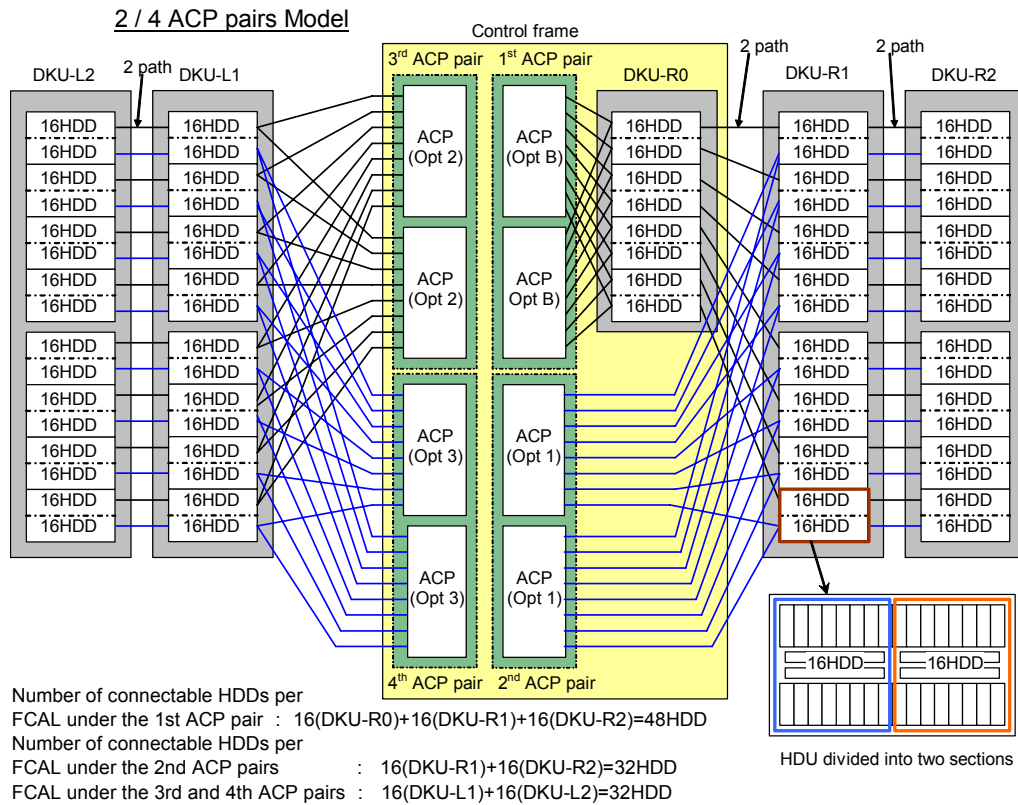
Optimizing performance raises the issue of demand versus capability. Additional performance improvements have very little effect on application performance when the performance capabilities of the array exceed the demand from applications. An analogy would be tuning a car engine. There are numerous techniques to increase the horsepower output of a car engine. However, the increased power has little effect if the driver continues to request that the car travel at only 30 MPH. If the driver does not demand the additional power, the capabilities go to waste.

The best results from performance tuning are achieved when the analysis considers the whole system, not just the array. However, short of a complete analysis, the easiest way to determine if an application could take advantage of array performance tuning is to study the queues in the I/O subsystem on the server and in the array. If there is little or no I/O queuing, additional array performance tuning is unlikely to improve application performance. The suggestions are still valuable, but if the array performance capabilities exceed the demand (as indicated by little or no queuing), the suggestions may yield only a modest gain.

Back-end processing

First, consider what can be done for performance to and from disks within the array. To do this, look at the architectural structure of the back-end. Figure 1 shows the array controller processors (ACPs) and how they are connected to the disk drives. It is important to note that the basic or first ACP pair can connect to 48 disk drives per Fibre Channel link. This means the ACP potentially could access 384 disk drives. From this figure, you can also see two ACP pairs manage the disk drives in the right side frames and two ACP pairs manage the disk drives in the left side frames.

Figure 2. XP12000 Disk Array back-end diagram



Small configurations using only one ACP (opt B) will start loading R0 with disks and then expand to the Frame DKU-R1.

For the highest performance, install all four ACP pairs. Disk drives would be added evenly into the R1 frame and the L1 frame. As you can see from Figure 2, this would provide the most paths to the disk drives. Loading R0 with active drives should be avoided to maintain an even load on the ACP pairs under a symmetrical configuration. Spare drives, archive data, and executables, which are typically only accessed during boot, are good candidates for R0. In all the frames the drives should be spread equally between the top of the frame and the bottom of the frame. As capacity needs grow, R2 and L2 could be added to meet the need.

For a pure capacity configuration, you would have one ACP pair and fill up R0 with drives, then add R1 and fill it, then add R2 filling it. Then you would add ACP (opt 3) and add and fill L1 then L2. If a unit was built up this way and was fully loaded with disk drives, the performance probably would not be acceptable without adding the last two ACP pairs.

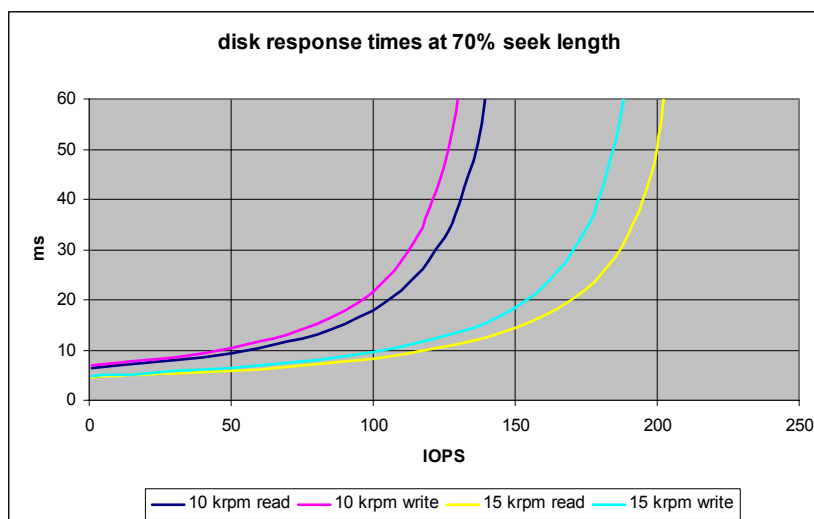
Disk performance

Disk drive specifications	73 GB, 15K	146 GB, 10K	300 GB, 10K	146 GB, 15K
Rotation speed	15k rpm	10k rpm	10k rpm	15k rpm
Mean latency time	2 ms	3 ms	3 ms	2 ms
Mean seek time (Read/Write)	3.5/4.0 ms	4.7/5.3 ms	4.7/5.3 ms	3.3/4.0 ms
Sustained transfer rate (MB/sec)	73 to 125	39 to 80	39 to 80	73 to 125

Seagate Cheetah data sheet

Based on these specifications and using the preceding queuing formula, the disk performance can be approximated as following.

Figure 3. Disk performance characteristic



This behavior does not reflect any optimization by cache.

Some general observations on disks:

- Transfer time on the outside is nearly twice as high as on the inside. This can be significant for sequential I/O loads with big block sizes. It has no effect on random I/O.
- Data density is twice as high on the outside as on the inside. Data on the outside is therefore spread over less tracks, which reduces seek times. This results in some improvement for random I/O.
- Keeping active data on adjacent tracks improves the average seek time.

Workload balance and contention management of the back-end

There are two types of workload that should be separated whenever possible: sequential and random I/O. Examples for these would be log files for sequential and data for random. The highest performance would come if you could separate by ACP pair and RAID Group. The minimum separation would be by RAID Group. The access patterns of sequential and random are very different at the drive level and the ACP. When doing a sequential read access, the ACP will contact the disk for pre-fetching. The disk will also be trying to fill its buffer with pre-fetched data. If a random I/O is done in the middle of this pre-fetching, the benefits of the pre-fetching are not only lost, but slow down both types of I/O as the buffers are dumped. When doing a sequential write, if a random read is done in the middle, the disk will reorder the read to complete after the write (disk optimizes workload for the disk) thus causing a delay potential of 100s on milliseconds for the read to complete.

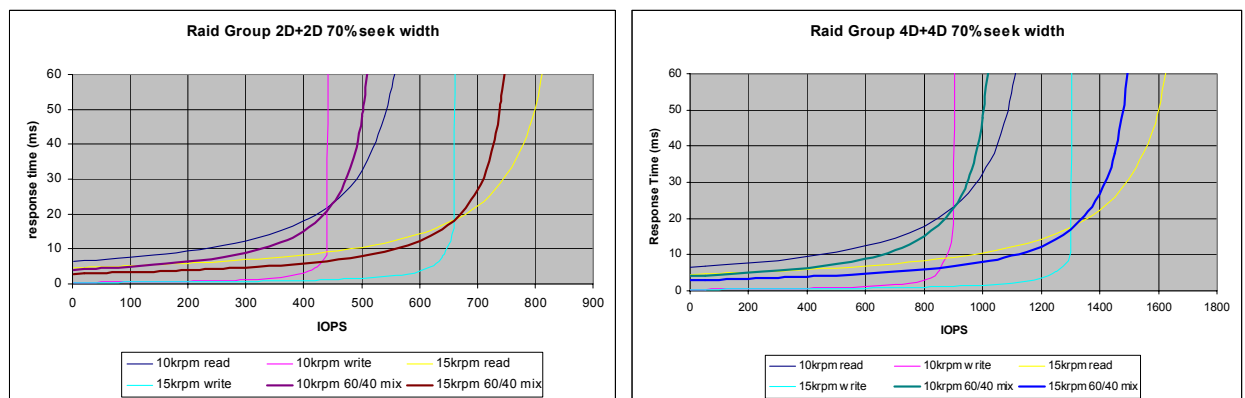
If there is data that is read often and changed very little and data that is written often, putting these on separate RAID Groups would also be advised. Having separate ACPs is not necessary.

RAID Group type has an effect

The RAID Group type will have an effect on I/O transfer rates from disk. RAID1 is 2D+2D and 4D+4D. RAID5 is 3D+1P, 7D+1P, 14D+2P, and 28D+4P where the latter two are a striped or interleave of 2 and 4 7D+1P RAID Groups. Table 1 shows the relative differences between RAID Groups for reads and writes in random access, sequential access, and random mix of 60% read and 40% write. These measurements were done with OPEN V emulation and are meant to show relative differences, as specific measurements on any system can vary. Each RAID Group type has differing availability characteristics, particularly for write performance. Each one also has a different cost characteristic as well.

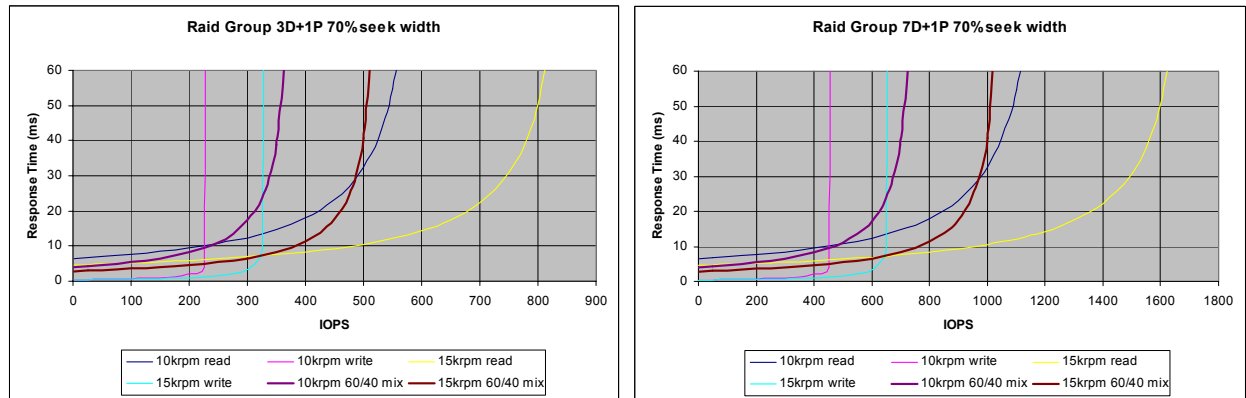
- **RAID1**—Consists of disk mirroring for performance and protection schemes. The set includes either four or eight disks (2+2 or 4+4). A RAID1 set can sustain the loss of one or more disks without loss of service, as long as both disk and its mirror are not lost. RAID1 is often used for database and log LUNs, as well as the SMTP mailroot directory. While RAID1 provides some of the highest performance and availability, it also has the highest cost factor as only 50% of the capacity is utilized for data storage.

Figure 4. RAID1 approximate performance characteristic



- RAID5**—Stripes data and parity across all of the disks, saving the information needed to rebuild the contents of any one disk lost in production when replaced with a spare disk. The set includes either four or eight disks (3D+1P or 7D+1P). RAID5 appears to offer the most efficient means for providing disk capacity, cost and fault resilience, although write performance is lower than RAID1 due to the added overhead of creating the parity data. Read performance is higher than RAID1 due to the higher data density.

Figure 5. RAID5 approximate performance characteristic



- RAID5 “Concatenated”**—The XP1200 Disk Array provides new functionality that allows you to “concatenate” multiple 7D+1P RAID Groups. This actually interleaves the stripes between two and four 7D+1P groups (see Figure 2). This interleaving provides the highest level of disk performance since there are more drives to spread the work across. The write penalty for creating parity is the same as with 7D+1P.

Figure 6. RAID Group concatenation by interleaving

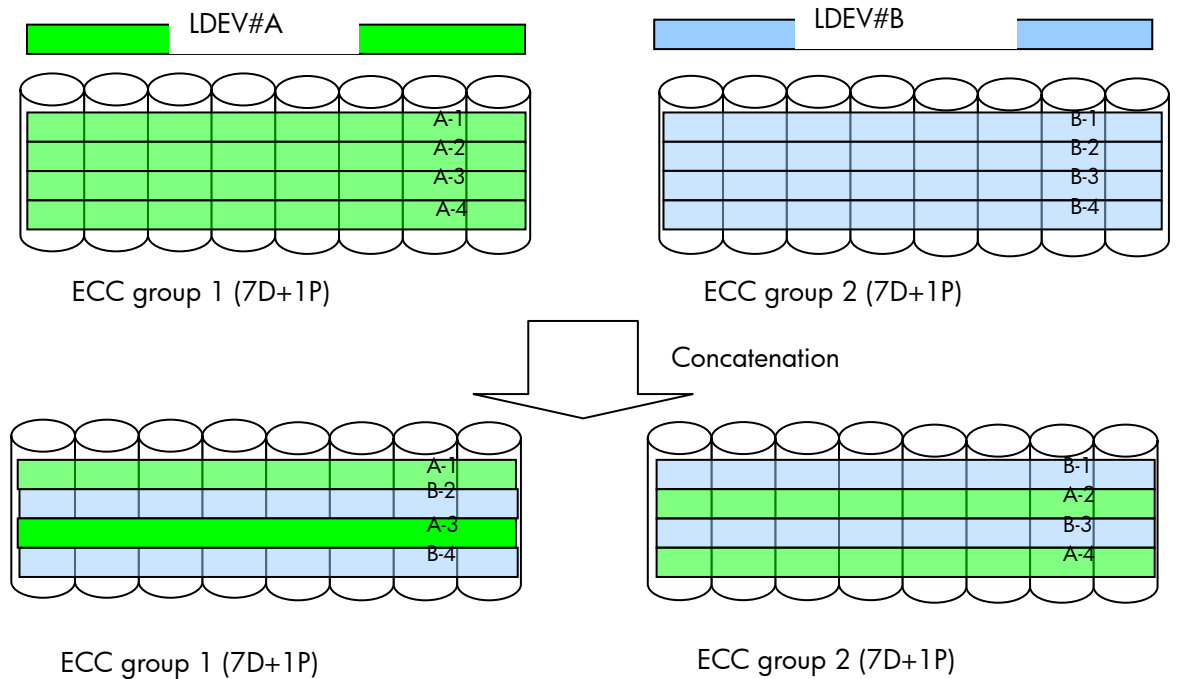
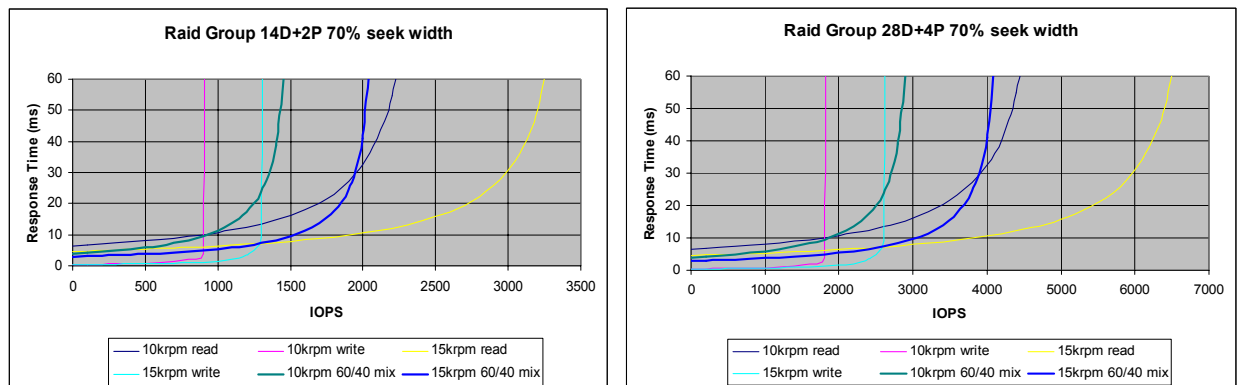
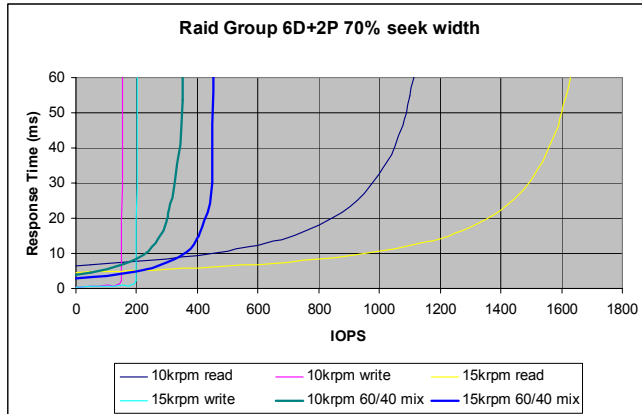


Figure 7. RAID5 concatenation—Approximate performance characterization



- **RAID6**—Similar to RAID5, with an extra set of parity data, supporting the loss of two spindles without complete loss of service to the data on these disks. While RAID6 provides additional fault resilience compared with RAID5, it also requires additional write overhead to generate the second set of parity data. Due to the added overhead of two sets of parity, RAID6 is generally not recommended for write-intensive applications.

Figure 8. RAID6—Approximate performance characterization



Disk emulation type has an impact

The XP12000 Disk Array supports old disk emulation types such as OPEN-3, OPEN-9, OPEN-E, and so on. These old emulation types will be referred to as OPEN-X as they all have similar performance degradation. The native OPEN-V type will provide the highest level of performance. The reason for this is OPEN-V has a slot defined as 256 KB. OPEN-X slot sizes will vary between 48 and 64 KB. The XP12000/XP10000 Disk Array cache slot size is 256 KB, so for OPEN-V it is a one-to-one mapping. For the other types, 256-KB cache slots will be used to hold the smaller 48–64-KB slots making cache less efficient.

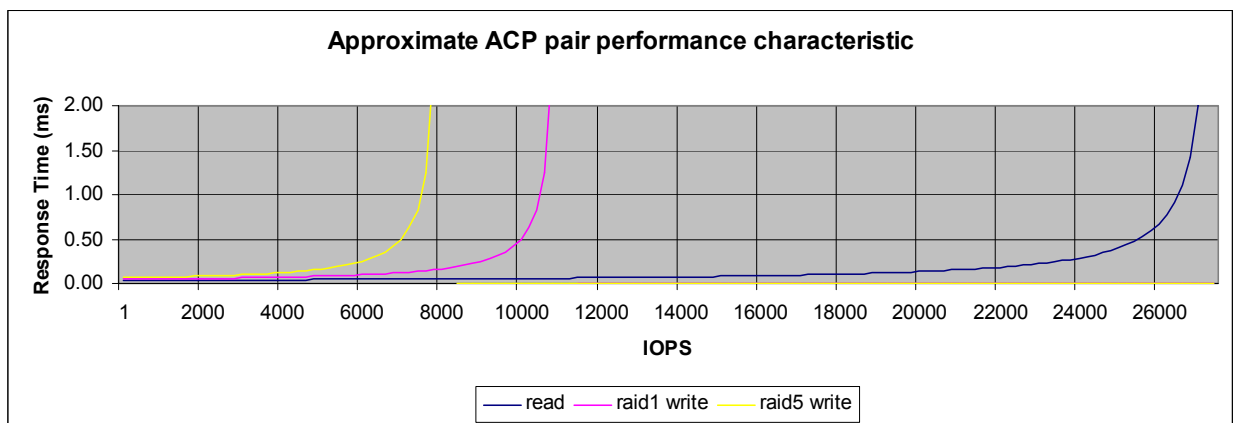
When the disk groups have been created, they need to be partitioned using an emulation type. The emulation type defines the available capacity of the parity or array group. Depending on which emulation type is selected, the array group is partitioned into multiple volumes (LDEVs) or slices. Emulation type has both fixed and user-defined capacity sizes. For performance applications you should use Open-V and define the capacities needed for your LDEVs.

Emulation type	Open-3	Open-8	Open-9	Open-E	Open-L	Open-V
Storage Capacity (LDEV)	2.4 GB	7.3 GB	7.3 GB	14.5 GB	36.4 GB	User defined

ACP considerations

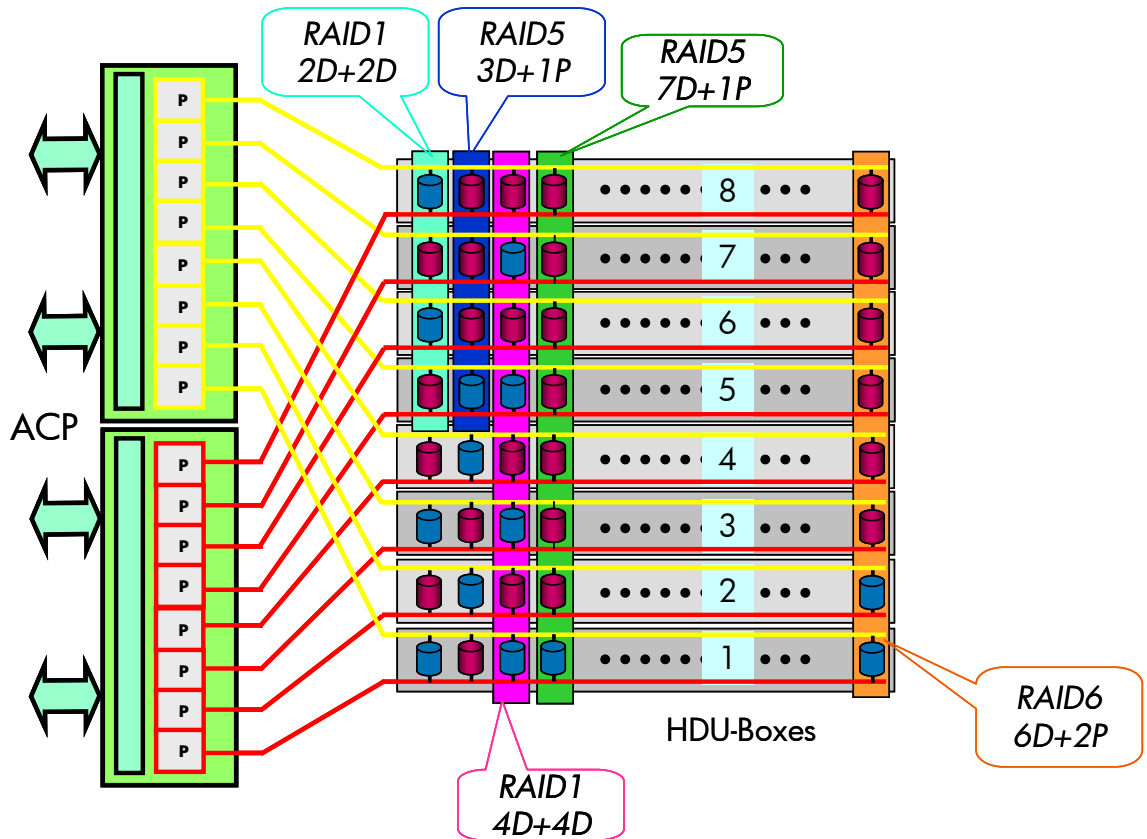
The ACP processors handle the data transfer between the cache and the disk drives. An XP12000 Disk Array can have one to four ACP board pairs. Each board pair services eight dual Fibre Channel loops with up to 48 dual-attached disk drives per dual loop on the first ACP pair and up to 32 dual attached disk drives per dual loop on the other ACP pairs. Each loop is serviced by its own processor. For more details, see Figure 2.

Figure 9. ACP performance characteristic



The different RAID Group types span either four (2D+2D, 3D+1P) or eight (7D+1P, 6D+2P) Fibre Channel loops. The RAID Groups should be distributed evenly across all Fibre Channel loops to achieve an even load distribution across all loops and thereby all ACP processors.

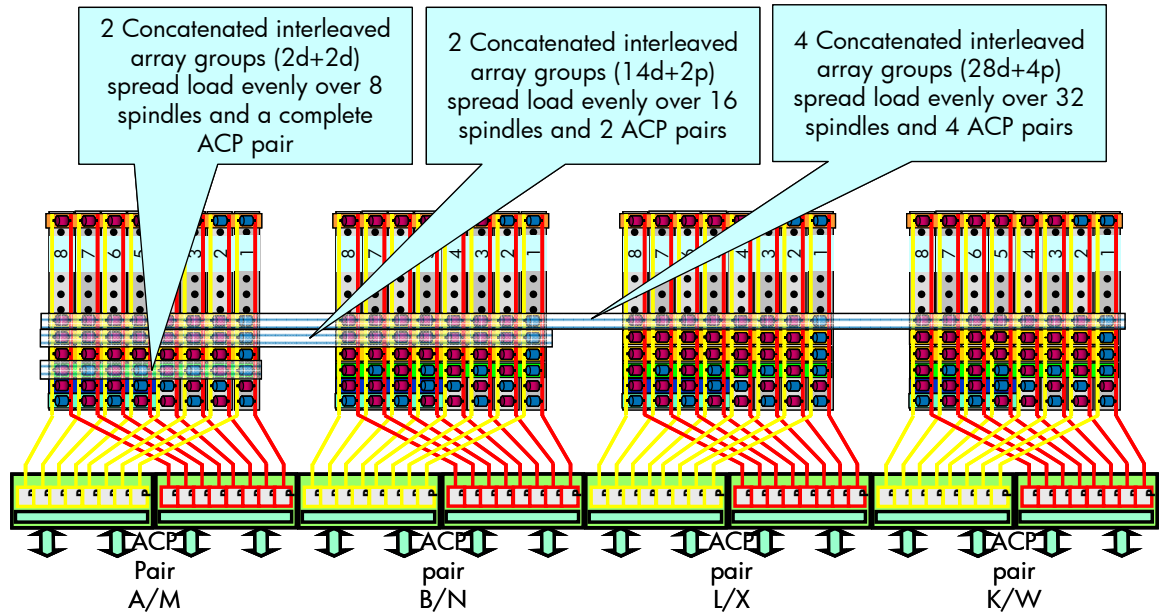
Figure 10. RAID Group layout



On systems with more than one ACP pair, it is a good practice to fill DKU R0 last with disk drives as long as there is still bandwidth left on the first ACP. This helps to maintain an even load distribution over the ACP pairs and eases management due to a symmetrical configuration.

There is no rule or limitation which RAID Groups can be concatenated. But by choosing the RAID Groups for concatenation that do not share Fibre Channel loops helps to distribute the load over the ACP pairs.

Figure 11. Using concatenated RAID Groups to distribute I/O load



In most cases it is better to spread wide and share resources than to segment the resources and assign specific resources to single applications. The configuration will then be more robust against unexpected peaks and workload changes. The resources will be used more efficiently.

Back-end rules of thumb

As a summary the following simple rules for a good configuration should be followed:

- Keep I/O rate less than 50–60 IOPS per 10k-rpm spindle and less than 70–80 IOPS per 15k-rpm spindle for typical workloads.
- Keep ACP utilization well under 70%.
- Spread load wide.
- Avoid populating RO on systems with more than one ACP pair.
- Balance load over spindles and ACP pairs. For the same load the average response times get higher the more unbalanced the system gets.
- Place active data on the outside of the spindles (typically the smaller LDEV numbers).

Front-end processing

Client Host Interface Processors (CHiPs)

Client Host Interface Processors (CHiPs) provide connections to host servers either directly connected (point-to-point) or through SAN switches when using Fibre Channel. The CHiPs are configured in pairs for redundancy. There are various iterations such as Fibre Channel, ESCON, iSCSI, and FICON, depending upon the array's configuration.

Figure 12 is a diagram of one half of a 16-port Fibre Channel CHiP pair (a 32-port pair is similar). It shows there is single processor per two ports. In this diagram two of the processors have their ports set up for a Continuous Access XP environment. For a performance application you want to use the

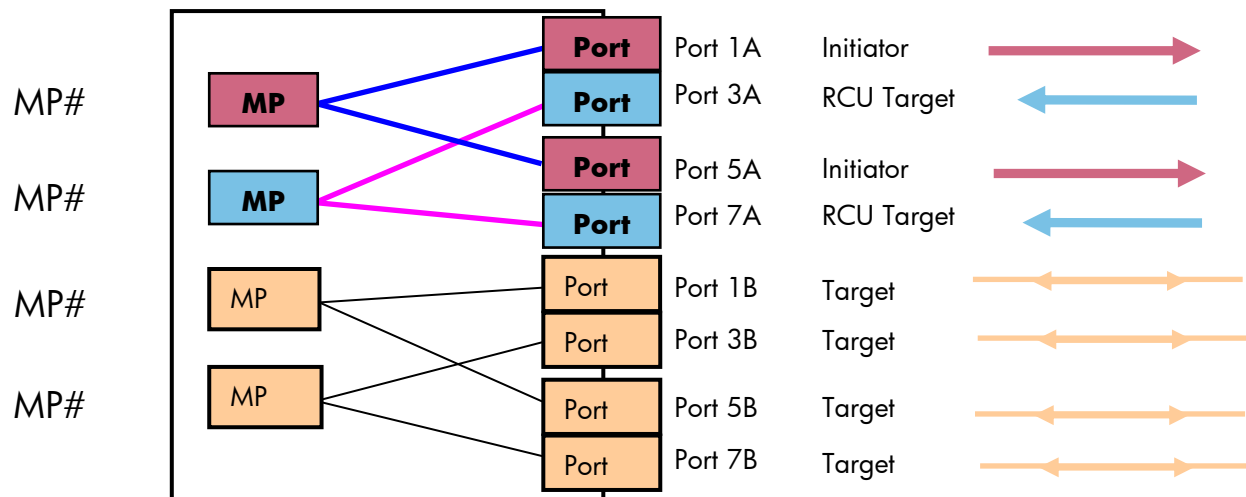
other two processors in high-performance mode, which would mean that only one port on each processor would be used to connect to hosts (if you have a shortage of ports, you could use one port per CHiP as a failover port). This high-performance mode provides the highest processing to a host since it would not share its bandwidth between two hosts. It should be noted that the 8-port CHiP pairs have one processor per port.

The reason processor bandwidth is important is that the CHiP manages the I/O until the I/O completes. The processor checks in shared memory to see if the data is in cache or if a disk access will have to be completed. On a write the process will mirror the write to cache before notifying the host the write is complete. The CHiP also performs the learning algorithms for sequential reads and writes to increase the performance of sequential access. These sequential algorithms also keep cache from filling up because of sequential access allowing for other applications to get good cache hit ratios for random access. CHiP processor utilization should be kept below 70% for the best performance. If CHiP processor utilization reaches 70% or higher, either rebalance the load between ports or add more ports to the XP array.

HP recommends segmenting ports between random and sequential I/O if possible because of the learning algorithms for sequential I/O. On sequential reads the CHiP will do prefetches from disk in order to keep the data needed in cache in advance of the host asking for it. For sequential writes the CHiP will keep full stripes in cache to avoid any penalty for parity creation. It will also manage the amount of cache being used for the writes to avoid starving other I/O of cache hits. The CHiPs can detect sequential I/O across multiple ports if the I/Os are split uniformly across those ports (for example, eight I/Os to one port and then eight to another).

HP does not recommend host port load balancing as many of these applications spread the I/Os in a non-uniform manner making it hard to detect sequential I/O, in which case the sequential I/O will have random I/O performance characteristics. Using host-based load balancing works quite well with random I/O.

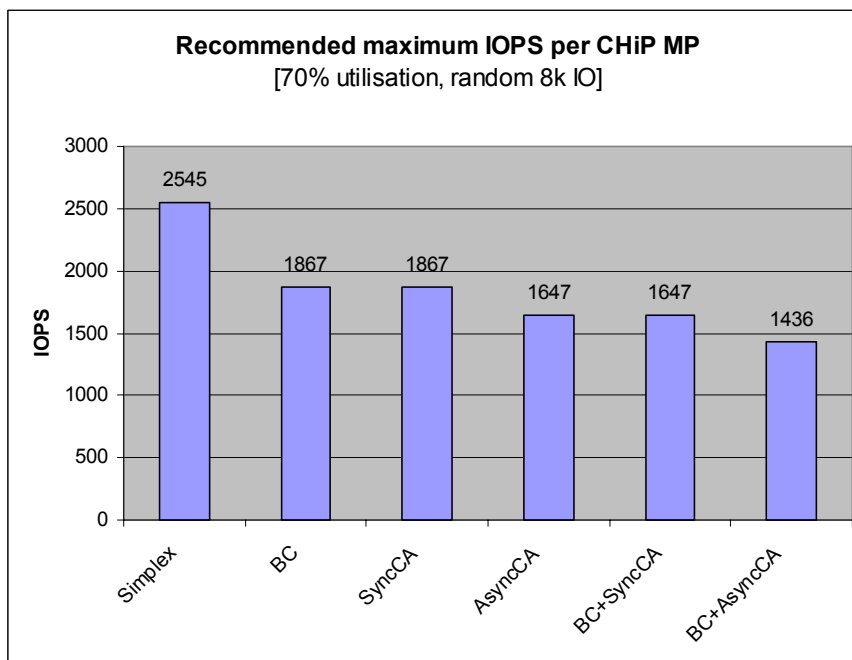
Figure 12. Front-end CHiP processors and ports



Replication software and CHiP processors

Replication software (Continuous Access XP, Business Copy XP, and Snapshot XP) all require extra processing from the front-end processors. Snapshot XP has about a 70% impact on writes due to copy on write technology that is used and should not be used in a performance-sensitive configuration. Continuous Access XP Extension (async) has about a 35% impact and Continuous Access XP Journal has about 25%. Of course Continuous Access XP (sync) is dependent on the speed of the link for writes and reads are the same as for Continuous Access XP Extension. Business Copy XP has the least because it is mostly a background process except when you do a Quick Split, then it is dependent on the size of the copy. Figure 13 can be used as a visual to understand the impact to the processor due to extra overhead that must be managed by the CHiP processor doing the I/O. This graph does not take into consideration assistance from cache hits.

Figure 13.

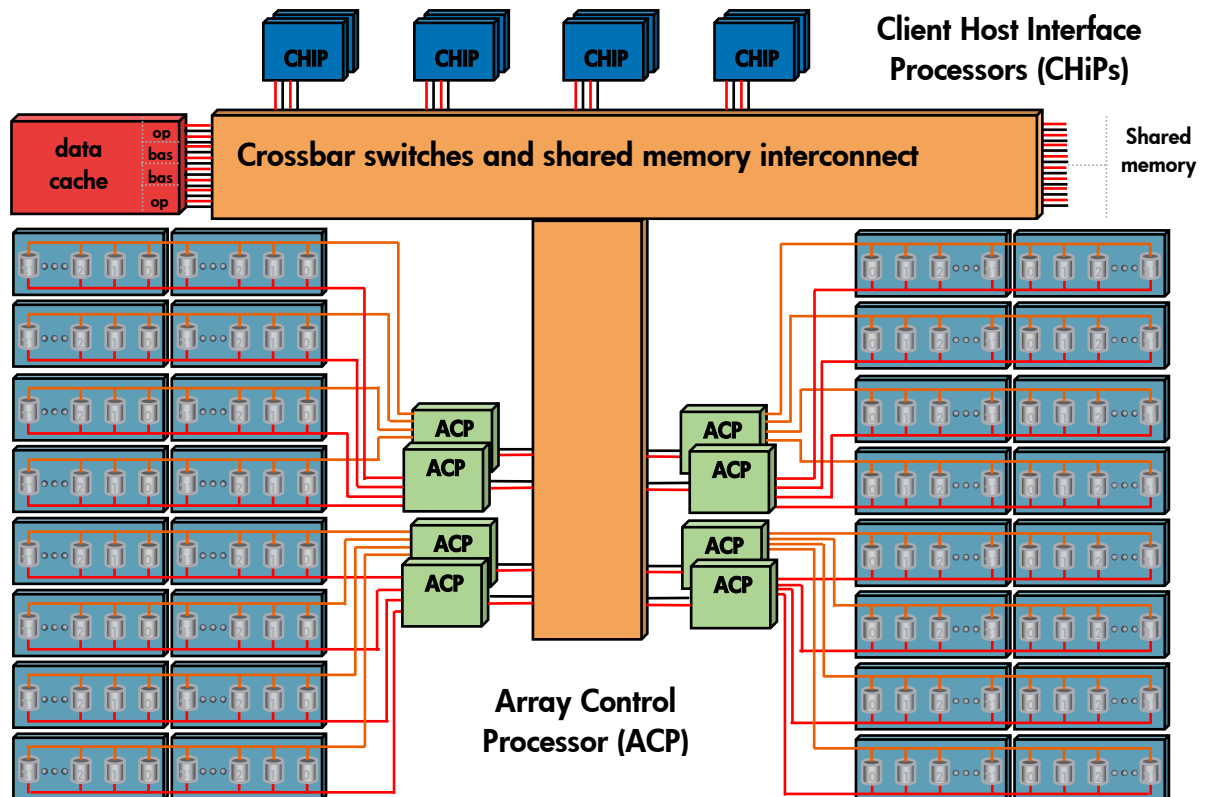


Cache and shared memory

Crossbar switch (CSW)

Communication between various subsystem components (cache, ACP, and CHIP) is implemented by means of a high-speed crossbar switch (CSW). It is the crossbar that allows for multiple redundant connections between the CHiPs, shared memory, cache, and the ACPs.

Figure 14. XP array architecture



Cache capacity and optional cache boards

Cache memory is used to transfer data between the host CHIP processors and the ACP disk controllers. In fact, cache memory is an integral part of the data path, as all data is transferred through cache before being transferred between the disk and the host server. Cache is also mirrored, allowing all write data to be written in cache twice, one time each on two separate battery-backed-up cache platform boards. The XP12000 Disk Array supports up to 256 GB of cache. Cache can also be partitioned, allowing a group of LDEVs to be allocated a specific portion of the cache memory. This will allow specific applications to have dedicated cache capacity for improved performance.

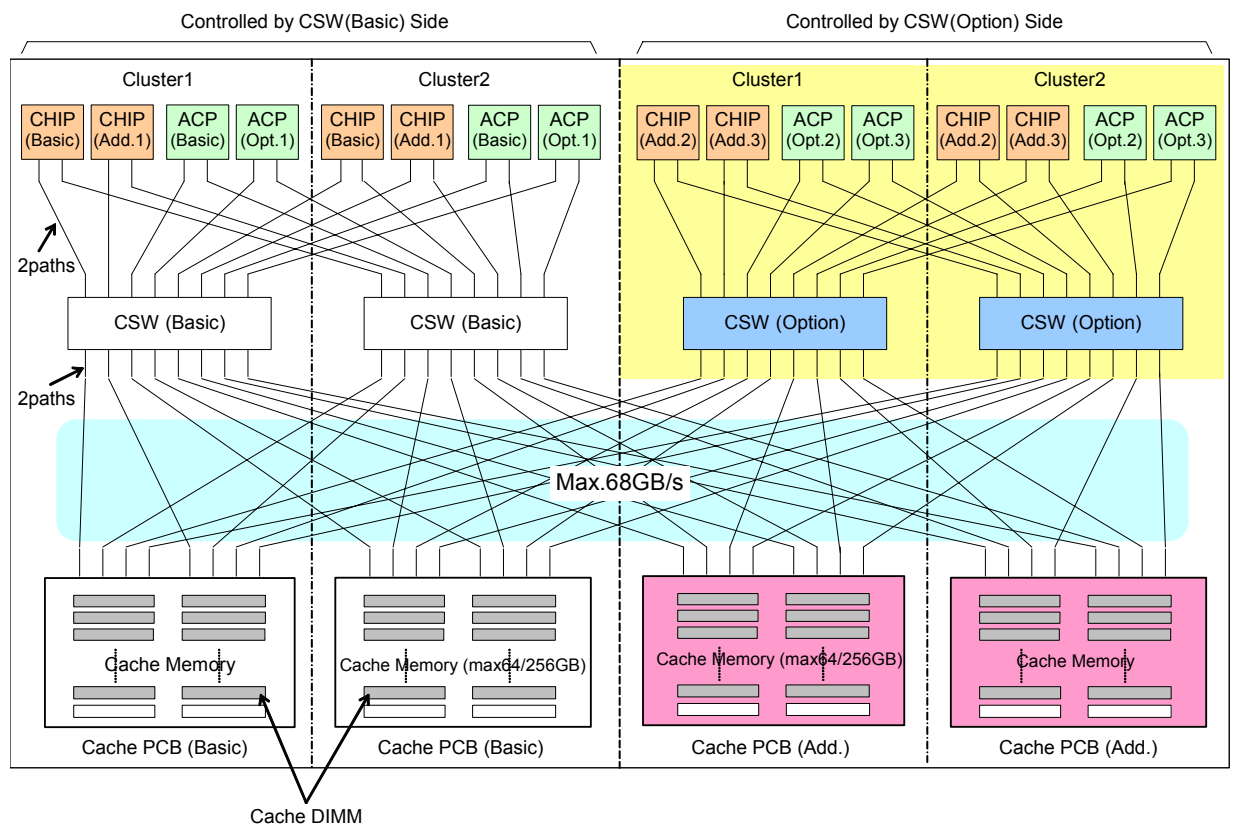
To get the highest cache performance you should add the optional cache boards and CSWs to the XP array and spread the cache memory equally across the four boards. This is referred to as High-Performance Cache Access Model. The addition of the extra boards provides more paths to and from the CSW. Extra paths mean quicker access for the processors (Figure 15).

A big advantage of large cache is the ability to cache bursts of write activity. When sizing cache you should consider the size of burst rates from the hosts attached to the XP array. To determine the amount of cache needed, read "Impact of Cache on the performance of the HP StorageWorks XP12000 Disk Array" at <http://h71028.www7.hp.com/ERC/downloads/4AA0-7924ENW.pdf>.

When cache Write Pending reaches 30%, cache destaging starts to get priority. The XP array will try to keep write data in cache to try and accumulate an entire stripe to avoid the read create parity penalty of RAID5. When Write Pending reaches 30%, destage priority does not allow write data to accumulate. Long periods with Write Pending over 30% should be avoided whenever possible. Adding more cache may help or may only delay reaching the 30% value.

If the data being accessed on the XP array has a wide locality (often the case with Microsoft® Exchange), adding more cache is not likely to provide better cache hit performance. In this case, consider back-end performance and plan for 2,500 IOPS per processor.

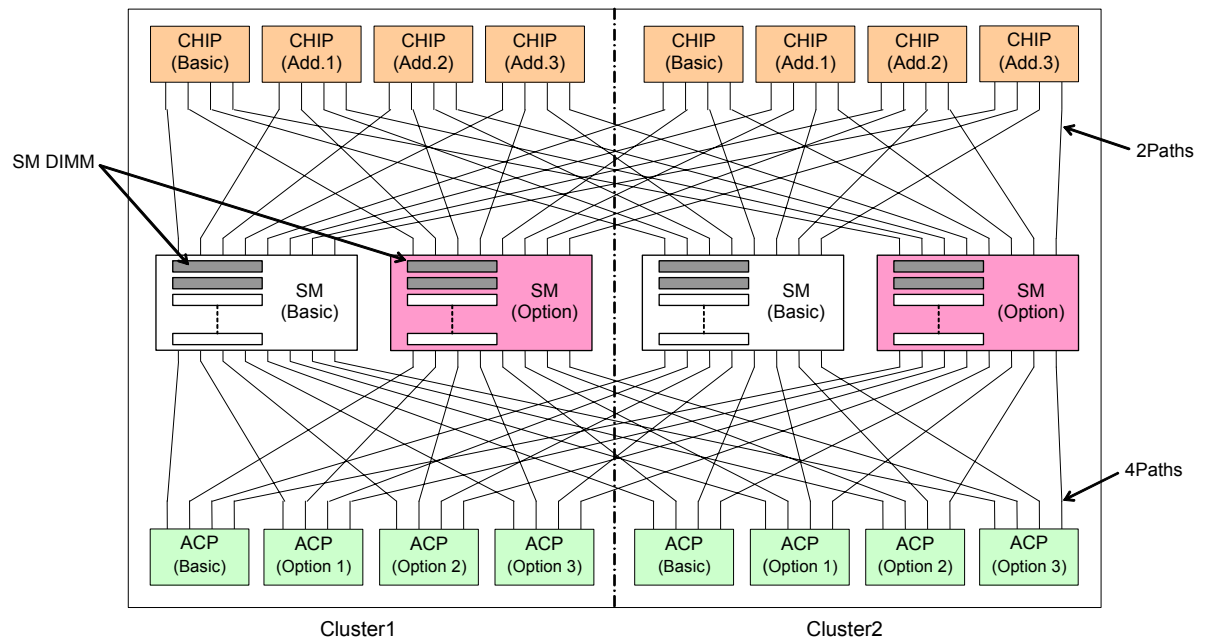
Figure 15. Cache and CSW



Shared memory

Shared memory is independent of the cache memory and is used to store tables and other information used for disk management, thus freeing up the cache memory for user data. Shared memory is also used to store system configuration information. The configuration information includes system component mapping, LUN maps, cache pointers, and RAID levels. Depending on configuration and disk capacities and functionality being deployed, 1 GB to 12 GB of shared memory may be required. As with cache the highest performance is going to be gained by adding the extra shared memory boards and equally spreading the memory across these boards. This will lower memory contention issues and increase path bandwidth.

Figure 16. Shared memory and processor interconnect



Front-end rules of thumb

- Use only one port actively of a CHiP processor port pair (for 16- and 32-port pair cards)
- Keep CHiP processor utilization below 70%
- Do not use host-based load balancing with sequential I/O
- Add the optional CSWs and cache boards
- Spread cache across all of the cache boards
- Do not have long periods with cache write pending over 30%
- Have cache size able to handle burst writes
- Add optional shared memory boards
- Spread shared memory across all shared memory boards
- For wide locality random I/O (no cache assist), plan on 2,500 IOPS per CHiP processor without replication software and 2,000 with Continuous Access XP or Business Copy XP
- Do not use SnapShot XP in a performance design

Glossary

These terms used throughout the white paper aid in understanding the innovative solutions provided by HP servers and storage.

Term	Definition
ACP	Array control processors
Array Group	See RAID Group
CHA	Client Host Adapter
CHIP	Client Host Interface Processor
CLI	Command Line Interface
CU	Control Unit
CLPR	Cache Logical Partition
DCR	Data Cache LUN Residence
Disk Group	See RAID Group
DKC	Disk control frame
DKU	Disk array frame
FC	Fibre Channel
SSID	Subsystem ID
GUI	Graphical user interface
HA	High availability
HBA	Host bus adapter
IOPS	I/Os per second
LDEV	Logical device
LUN	SCSI Logical Unit Number
nPar	HP nPartition (hard partition)
OLTP	Online Transaction Processing
Parity group	See RAID Group
PCR	Partial Cache Residence
PSA	Partition Storage Administrator
QoS	Quality of Service
RAID Group	The set of disks that make up a RAID set
RWC	Remote Web Console
SA	Storage administrator
SAN	Storage area network
SLA	Service Level Agreement
SLPR	Storage Management Logical Partition
SPOF	Single point of failure
WWN	World Wide Name, a unique 64-bit device identifier in a Fibre Channel storage area network

For more information

For additional information, refer to the following resources.

Resource description	Web address
HP StorageWorks Snapshot XP user guide	http://www.hp.com/support/rwc/manuals
HP StorageWorks RAID Manager XP user guide	http://h20000.www2.hp.com/bizsupport/TechSupport/Home.jsp

© 2007 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft is a U.S. registered trademark of Microsoft Corporation.

4AA0-7923ENW, Rev. 1, January 2007

