

Optimizing RAID Storage Performance with the SAS-based Dell PERC 5

BY DUSTIN SORENSON
LETICIA OLIVAREZ

The Serial Attached SCSI (SAS)–based Dell™ PowerEdge™ Expandable RAID Controller 5 (PERC 5) can significantly increase performance over previous-generation SCSI-based PERCs. This article outlines the storage system architecture of the PERC 5 and discusses how administrators can optimize RAID storage performance for different types of workloads.



The Dell PowerEdge Expandable RAID Controller 5 (PERC 5) is the first Dell RAID controller built on Serial Attached SCSI (SAS) technology, and can significantly increase performance over previous-generation SCSI-based PERCs. To help optimize storage performance when using the PERC 5, enterprises should consider their application workload characteristics as well as how different storage system configurations—such as RAID level, stripe element size, cache policy, logical disk array size, and hard disk specifications—can affect capacity, reliability, and performance.

Related Categories:

RAID

RAID controllers

Serial Attached SCSI (SAS)

Storage

Visit www.dell.com/powersolutions for the complete category index.

Understanding the PERC 5 storage system architecture

Figure 1 illustrates a SAS-based storage system incorporating a PERC 5. The PERC 5 is connected to the server through an x8 PCI Express slot rated at 2 GB/sec (although some systems may utilize an x4 PCI Express slot, rated at 1 GB/sec). The PERC 5 has two SAS wide ports to connect the controller to expanders. Each wide port is a combination of four SAS ports rated at 300 MB/sec each, providing a throughput of 1,200 MB/sec per expander, or 2,400 MB/sec total. The expander routes the four SAS ports to each of the 15 disks it can support, or connects to additional expanders. SAS direct links, rated at 90–100 MB/sec, connect the expander to the disks.

Assuming sustained data rates of 100 MB/sec, 24 hard disks should saturate the 2,400 MB/sec available to the two

SAS wide ports, causing the x8 PCI Express slot to become a bottleneck. But because PCI and SAS bus efficiencies can prevent the buses from reaching their rated speeds, the bottleneck may appear with a greater number of disks: x8 PCI Express buses typically have an efficiency of approximately 80 percent, while SAS typically varies between 50 and 90 percent depending on the workload. The transfer size generally dictates how much impact protocol overhead has on the potential throughput—for example, one 16 KB transfer typically has less overhead than eight 2 KB transfers.

The PERC 5 can support larger arrays than previous-generation PERCs, helping increase capacity and workload distribution, and therefore performance as well. For example, servers using many small files, such as an e-mail server, can distribute the workload across many hard disks to help increase performance, measured in I/O operations per second (IOPS) rather than bandwidth (MB/sec).

Evaluating application workloads

System configuration decisions typically require trade-offs between capacity, reliability, and performance as well as their related costs. To help optimize system performance, administrators should first understand and define their workload, which can have as much impact on performance as system configuration. In particular, they should determine the following:

“ To help optimize system performance, administrators should first understand and define their workload, which can have as much impact on performance as system configuration.”

- **I/O type:** Consecutive I/O requests that are physically near each other are *sequential*; consecutive I/O requests that are physically far from each other are *random*. Typically, sequential I/O workloads use large transfer sizes and queue depths to help optimize throughput, while random I/O workloads use small transfer sizes and variable queue depths. I/O type can affect decisions on RAID level, stripe element size, cache policy, logical disk array size, and hard drive specifications.
- **Transfer size:** Transfer size is the size of the payload or data block that the system stores. It can affect decisions on stripe element size and cache policy.
- **Read/write balance:** Understanding the balance between read and write operations can help administrators optimize a system’s efficiency for both operations. This balance can affect decisions on RAID level and cache policy.
- **Queue depth:** Queue depth is the number of outstanding I/O requests issued at any given time. It can affect decisions on stripe element size and cache policy.

On existing systems, administrators can use tools such as Microsoft® Windows® Performance Monitor (perfmon) and iostat to help understand the actual workload. Otherwise, they must make assumptions regarding the workload based on the intended application.

Configuring RAID storage systems

Once they understand their workload, administrators can use this information to make decisions regarding storage capacity, reliability, and

performance, with the goal of optimizing the configuration for their specific requirements. Configuration options they should consider include RAID level, stripe element size, cache policy, logical disk array size, and hard drive specifications.

RAID level

The PERC 5 supports RAID-0, RAID-1, RAID-5, RAID-10, and RAID-50. Different RAID levels use different combinations of striping, mirroring, and parity—elements that can affect capacity, reliability, and performance, as well as performance in a degraded state and rebuild times.

Striping, mirroring, and parity. *Striping*, used by RAID-0, RAID-5, RAID-10, and RAID-50, is designed to increase performance by spreading workloads across several disks rather than a single disk, reducing the impact of seek times

for random I/O. It spreads the workloads by subdividing each disk into *stripe elements*, the amount of data each disk handles before sending I/O operations to the next disk, which can vary in size from 8 KB to 128 KB. Striping does not increase reliability.

Mirroring, used by RAID-1 and RAID-10, creates an exact copy of one disk on another disk, helping increase reliability by mitigating the impact of disk failures. Mirroring can increase the performance of read operations, and may slightly degrade performance for write operations. However, because mirroring uses half an array’s capacity to create the mirror, it can increase deployment costs.

Parity, used by RAID-5 and RAID-50, utilizes exclusive OR (XOR) operations on data and stores the results as parity data, helping increase reliability when used with striping. Following a disk failure, the parity information can be used to re-create the failed disk’s data, and because parity storage only requires one disk, it takes up less capacity than mirroring.

Parity calculations only slightly degrade the performance of read operations, but can significantly degrade the performance of write operations. The three types of parity calculation—read-peers, read-modify-write, and full stripe write—carry different performance penalties.

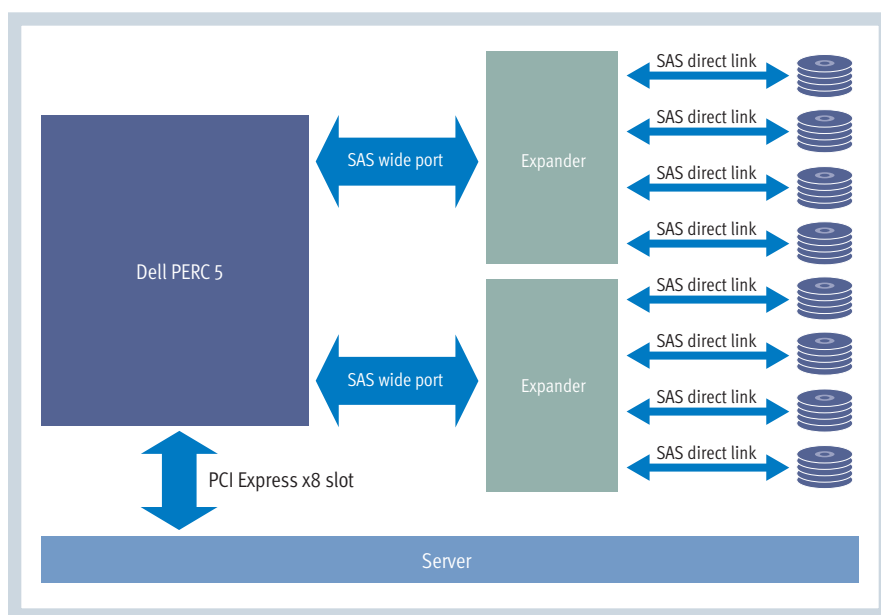


Figure 1. SAS-based storage system incorporating a Dell PERC 5

Read-peers reads each disk to calculate the parity, then performs XOR operations with the new data to calculate the new parity. *Read-modify-write* reads only the current parity from the disk and the old data, then uses that information and the new data to calculate the new parity, reducing the number of read operations to two. *Full stripe write*, the fastest of the three methods, calculates parity directly from the data being written when that data is equal to the stripe size, eliminating the need to read any information from the disk.

When configuring a PERC 5 to use a parity RAID level, administrators should consider the performance penalty based on the number of disks and the transfer size compared with the stripe size. *Read-peers*, which has the highest penalty of the three types, occurs when the logical disk array comprises fewer than four disks or when the array is in a degraded state. Full stripe write, which has the lowest penalty, occurs when the transfer size is larger than the virtual disk stripe size. In other situations, *read-modify-write* is typically used.

Capacity, reliability, and performance.

When choosing a RAID level for a particular workload, administrators must weigh capacity and reliability against performance, taking into account the different ways mirroring and parity can affect these factors (see Figure 2).

The read and write performance of a given RAID level varies depending on workload and other factors. Read performance, for example, can depend significantly on the number of available drives. Taking advantage of all disks in an array allows RAID-10 to generally offer the best read performance of the different RAID levels for workloads using random I/O. The additional parity information of RAID-5 enables it to provide slightly faster read performance than RAID-50.

Write performance is affected significantly by mirroring and parity. The mirroring in RAID-10, for example, means that RAID-10 typically has half the write performance of RAID-0, and the parity in RAID-5 and RAID-50 also reduce write performance. Random write operations typically provide worse performance than sequential

RAID level	Additional capacity requirement	Data reliability
0	None	Cannot accommodate any disk failures
1	Half of total capacity	Can accommodate failure of one disk
5	One disk	Can accommodate failure of one disk
10	Half of total capacity	Can accommodate failure of half the array
50	One disk per spanned array	Can accommodate failure of one disk from each spanned array

Figure 2. Additional capacity requirements and data reliability for RAID levels supported by the Dell PERC 5

write operations with RAID-5 and RAID-50, primarily because the small random transfer sizes prevent full stripe writes.

Degraded performance and rebuild times.

Following a disk failure, RAID-1 and RAID-10 mirroring rely on the healthy disk to provide storage availability to hosts. Degraded mirrors lose their performance advantage for random read operations. A major advantage of RAID-1 and RAID-10 is that random write and sequential operations have a minimal performance impact in a degraded state.

Disk failure reduces the read performance of RAID-5 and RAID-50 because of the XOR parity operations on the remaining disks necessary to re-create the missing data. It also reduces the performance of random write operations because the parity calculation must use *read-peers* for the data on the failed disk. The performance of sequential write operations is typically not affected (and can actually be slightly increased because of the reduced number of write operations), because these operations utilize full stripe writes for parity calculation.

The PERC 5 allows administrators to configure rebuild priorities, enabling them to balance data access performance with rebuild time.

Stripe element size

The total stripe size for RAID storage systems using the PERC 5 is defined in the PERC 5 as the stripe element size multiplied by the number of disks in the array, minus one disk for RAID levels that use parity. Administrators should

consider I/O type, transfer size, and queue depth when setting the stripe element size. In random I/O applications, data seek time is typically the primary factor affecting performance, and using the largest possible stripe element size helps maximize performance by increasing the odds that a single disk performs the data seek. If a sequential I/O application can guarantee a fixed transfer size, using a stripe size equal to the transfer size helps maximize performance by balancing the workload across each disk and allowing full stripe writes for parity RAID levels.

Setting the appropriate stripe size for sequential I/O workloads is difficult when the workloads do not guarantee data alignment and use variable transfer sizes. The goal is to set the stripe element size to evenly distribute the workload across each disk in the array. For applications with a large queue depth, administrators typically should use a large stripe element size to help ensure that the multiple sequential I/O operations are load balanced. For applications with a small queue depth, matching the stripe size to the transfer size helps provide an appropriate workload balance and can enable full stripe writes.

Administrators should also consider how stripe size can affect rebuild times: large stripes help decrease rebuild times by allowing large sections of the failed disk to be rebuilt at a time. In addition, they should test possible configurations on an actual application, using the guidelines in this section as a starting point, to help determine the optimal settings.

Cache policy

The PERC 5 supports write-back and write-through policies. *Write-back* waits until the data is stored in the cache before signaling the host that an operation is complete, and requires that the data eventually be written to the disk. The danger of this policy is that if system power is lost before data has been written to a disk, that data is lost. The PERC 5 is designed to avoid this problem by providing a battery backup to its cache memory and a recovery mechanism to complete write operations following a power failure.

Write-through waits until the data is written to the disk before signaling the host that an operation is complete. This policy is typically appropriate for non-parity RAID levels with a large queue depth. If the queue depth fills the cache faster than it can write to the disks, write-back can actually become slower than write-through.

If an application is running on a non-parity RAID level and has large transfer sizes, administrators should test the system with both write-back and write-through policies to help determine the optimal setting.

Logical disk array size

The logical disk array size—the number of physical disks in a logical disk—can affect storage capacity, random I/O performance, performance in a degraded state, and rebuild times. Large arrays typically provide better random and sequential I/O performance than small arrays, but only to the point of upstream bus saturation. However, large arrays can also reduce performance in degraded arrays and increase rebuild times by using read-peers to calculate the missing data. RAID-50 helps limit the performance impact of a large array in a degraded state by striping two RAID-5 spans, resulting in only half of the array actually being degraded.

Hard disk specifications

Hard disk specifications can affect both random and sequential I/O performance. For random I/O applications, where file sizes are typically small, moving the head to the sector is the

biggest performance hurdle; spindle speed is the primary factor affecting performance, followed by platter size, with smaller platter sizes providing increased performance. For sequential I/O applications, hard disk platter size is the primary factor affecting performance, followed by bit density and spin speed.

Optimizing RAID storage performance

The SAS-based Dell PERC 5 can help increase performance and capacity over previous-generation SCSI-based PERCs. When configuring RAID storage systems with the PERC 5, administrators should analyze the different trade-offs, particularly how the workload I/O type as well as the RAID level, stripe element size, cache policy, virtual disk array size, and hard disk specifications can affect capacity, reliability, and performance. Taking all of these elements into account can help administrators optimize their RAID storage for specific enterprise workloads. [u](#)

Dustin Sorenson is a development engineer on the Dell Host-based Storage Controllers team concentrating on controller performance and functional tests. Dustin has a B.S. in Electrical Engineering from the University of Texas at Austin.

Leticia Olivarez is a performance engineer at the Dell Server Performance and Analysis Lab. Her responsibilities include characterization of I/O performance across Dell RAID and PowerVault™ products. Leticia has a B.S. in Management/Computer Information Systems from Park University.