RAID Technology

In the 1980s, hard-disk drive capacities were limited and large drives commanded a premium price. As an alternative to costly, high-capacity individual drives, storage system developers began experimenting with arrays of smaller, less expensive hard-disk drives. In a 1988 publication, *A Case for Redundant Arrays of Inexpensive Disks* , three University of California-Berkeley researchers (David Patterson, Randy Katz, and Garth Gibson) proposed guidelines for these arrays. They originated the term RAID—redundant array of inexpensive disks—to reflect the data accessibility and cost advantages that properly implemented arrays could provide. (As storage technology has advanced and the cost per megabyte of storage has decreased, the term RAID has been redefined to refer to independent disks, emphasizing the potential data availability advantages relative to non-RAID arrays.)

RAID storage has now grown from an academic concept to an essential part of many workstation and server installations. Adoption of RAID technology has been driven by four key trends:

Increased capacity requirements — As file sizes of networked applications increase, so do storage needs. RAID-based storage systems offer high storage capacity as well as good scalability for future expansion.

Faster microprocessors — Advances in microprocessors have surpassed gains in drive technology. Since 1985, there has been a 100-fold increase in processing performance compared to a quadrupling of disk data-transfer rates. A properly implemented RAID storage system matched with the computer's application program can alleviate this *input/output (I/O) gap* and improve system throughput or I/O transfer rates relative to single-disk storage.

Reliability requirements — As mission-critical applications migrate to networked servers, RAID storage is becoming an integral tool in maintaining access to data and preventing loss of vital information in the event of a drive failure. RAID systems employ a variety of data redundancy and data restoration methods.

Decreasing costs — As costs per megabyte of storage have decreased, RAID storage use has spread from large data centers to networked servers, workstations, and desktop computers. In each case, the benefits of RAID can outweigh the cost of the additional drives required.

This white paper discusses the various RAID levels in current use, how RAID is implemented at the system level, and factors affecting RAID storage system performance. The paper concludes with a brief look at future RAID trends, a description of Dell's current RAID product offerings, and suggestions for appropriate RAID choices in different situations.

RAID Overview

In their landmark 1988 paper and an additional paper the following year, the Berkeley researchers established guidelines for six models or *RAID levels* — RAID 1 through RAID 6. (The *term level* is somewhat misleading because these models do not represent a hierarchy. A RAID 5 array is not inherently better or worse than a RAID 1 array.)

Current RAID products vary in their implementation of the Berkeley RAID levels. However, examining the original

RAID specifications, as well as the commonly used RAID 0 model, helps to illustrate basic RAID principles and why particular RAID levels are more appropriate than others for certain environments.

RAID 0

A majority of RAID levels involve a storage technique known as *data striping* . The most basic implementation of this technique has become known as RAID 0 and is supported by many storage product manufacturers. However, because this type of array has no inherent fault tolerance, RAID 0 is not *true* RAID unless it is used in conjunction with other RAID levels. (These hybrid RAID types are discussed later in this paper.)

Striping is a method of mapping data across the physical drives in an array to create a large virtual drive. The data is subdivided into consecutive segments or *stripes* that are written sequentially across the drives in the array (see Figure 1). Each stripe has a defined size or *depth* in blocks.

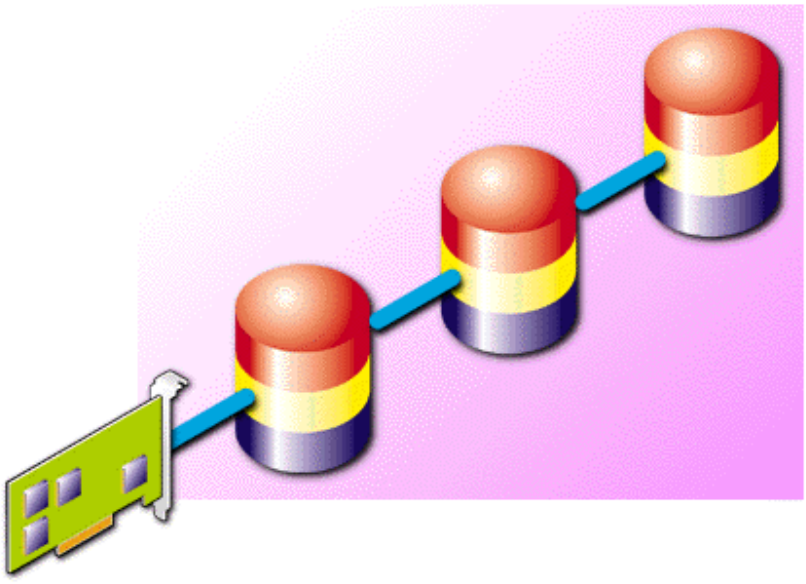

Figure 1. RAID 0 Array With Data Striping

A striped array of drives can offer improved performance compared to an individual drive if the stripe size is matched to the type of application program supported by the array:

- In an I/O-intensive or transactional environment where multiple concurrent requests for small data records occur, larger (block-level) stripes are preferable. If a stripe on an individual drive is large enough to contain an entire record, the drives in the array can respond independently to these simultaneous data requests.
- In a data-intensive environment where large data records are stored, smaller (byte-level) stripes are more appropriate. If a given data record extends across several drives in the array, the contents of the record can be read in parallel, improving the overall data transfer rate.

RAID 0 arrays can provide high write performance relative to *true* RAID levels because there is none of the overhead associated with parity calculations or other data recovery techniques. This same lack of provision for rebuilding lost data means that RAID 0 arrays should be restricted to storage of noncritical data and combined with a strict backup program. (A RAID 0 array is actually less reliable than a single drive of the same capacity. The mean time between failures [MTBF] of a RAID 0 array of n drives each having an MTBF of *x* hours is *x/n* hours. For example, a RAID 0 array of four drives with a MTBF of 100,000 hours has an MTBF of 25,000 hours.)

RAID 1

RAID 1 is the simplest form of fault-tolerant array. Based on the concept of *mirroring* introduced in the 1988

Berkeley paper, this array consists of multiple sets of data stored on two or more drives (see Figure 2). Although many RAID 1 implementations involve two sets of data (hence the term *mirror*), three or more sets can be created if increased reliability is desired.

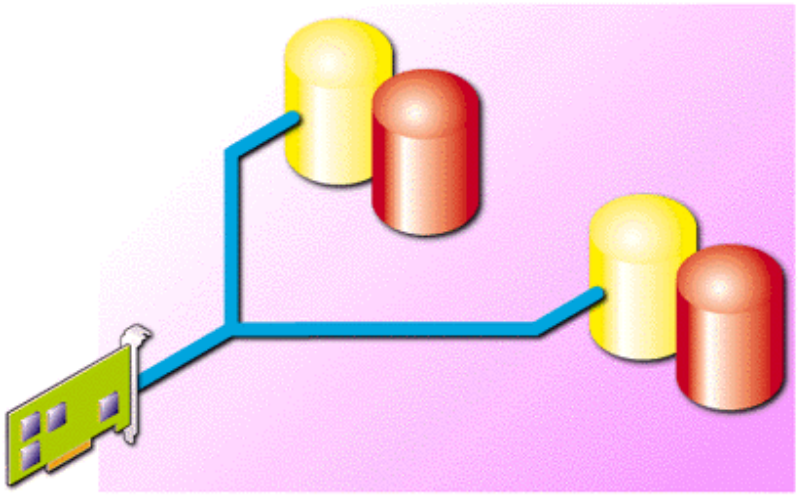

Figure 2. RAID 1 Array With Mirroring

If a drive failure occurs in a RAID 1 array, subsequent read and write operations are directed to the surviving drive(s). A replacement drive is then rebuilt using data from the surviving drive. This rebuilding process has some impact on the array's I/O performance because all data must be read and copied from the surviving drive(s) to the replacement drive.

RAID 1 offers high data availability because at least two complete sets of data are stored. Connecting the primary drives and mirrored drives to separate drive controllers can further enhance fault tolerance by eliminating the controller as a single point of failure.

RAID 1 has the highest storage cost of any nonhybrid RAID level because it requires sufficient drive capacity to store at least two complete sets of data. Thus, RAID 1 arrays are better suited for small databases or other small-scale systems that emphasize reliability.

RAID 2

The 1988 Berkeley paper proposed the RAID 2 level as a means of adding data protection to a basic striped array. The RAID 2 specification calls for the use of an error checking and correction (ECC) method (Hamming code) that groups data bits and check bits. Because commercially available hard-disk drives do not support the specified ECC code, RAID 2 has not been implemented commercially.

RAID 3

Like RAID 2, RAID 3 is a type of striped array, but it incorporates a more practical method of data protection. RAID 3 uses parity information for data recovery and stores it on a dedicated *parity drive* (see Figure 3).

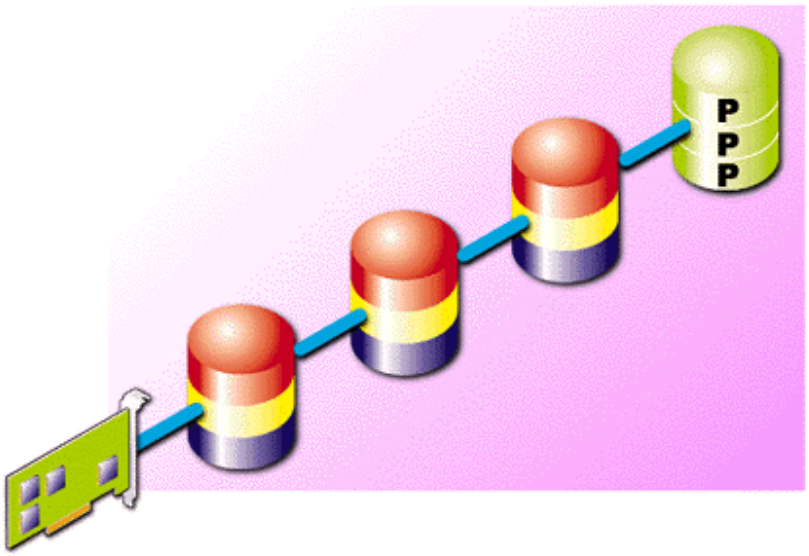

Figure 3. RAID 3 Array With Dedicated Parity Drive

The remaining *data drives* in the array are configured to use small (byte-level) data stripes, which distribute the contents of a large data record across all the drives. This arrangement can improve performance relative to a single drive because the data record's contents can be transferred in parallel to and from all the data drives in the array. (By comparison, if an application program such as a database requires frequent small data transfers, the overall efficiency of a parallel access array is similar to that of a single, relatively slow hard-disk drive.)

RAID 3 arrays and other parity arrays employ a more involved data recovery process than mirrored arrays such as RAID 1. The exclusive OR (XOR) function of data and parity information on the remaining drives is computed to *regenerate* the data on the failed drive. (See the following subsection, "Data Recovery Using Parity.")

Because all parity data is written to a single drive, RAID 3 suffers from a *write bottleneck* . Whenever data is written to the array, existing parity information is usually read from the parity drive and new parity information must always be written to the parity drive before the next write request can be fulfilled. Because of these two I/O operations, the parity drive becomes the limiting factor in overall write performance. (Full implementation of RAID 3 would include synchronization of drive access usually achieved by *drive spindle sync* , which is rarely implemented in the storage industry.)

Because they require only one additional drive for data protection, RAID 3 arrays are less expensive than RAID 1 arrays. Their ability to perform high-bandwidth data transfers and the disk drives' read-ahead algorithms make RAID 3 arrays particularly appropriate for single-user environments or applications that require access to large sequential records, such as video or image processing. This RAID level is not suitable for transactional environments that generate multiple simultaneous updates of data.

Data Recovery Using Parity

Parity RAID levels combine striping and parity calculations to permit data recovery if a disk fails. Parity values are calculated for the data in each stripe on a bit-by-bit basis. In an even-parity scheme, if the sum of a given bit position is odd, the parity value for that bit position is set to 1; if the sum is even, the parity bit is set to 0. (The converse is true for an odd-parity scheme.) In the following example, a block of data containing the values 135, 11, 96, and 157 is striped across a RAID 3 array with four data drives and a parity drive, using the even-parity method (see Figure 4).

Figure 4. Example of RAID 3 Array With Parity Drive

Table 1 shows the binary values of the data on each drive and the resulting parity value for each bit position.

		Binary Value of Data							
Drive	Data Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	135	1	0	0	0	0	1	1	1
2	11	0	0	0	0	1	0	1	1
3	96	0	1	1	0	0	0	0	0
4	157	1	0	0	1	1	1	0	1
Sum of bits		Even	Odd	Odd	Odd	Even	Even	Even	Odd
Parity Values		0	1	1	1	0	0	0	1

Table 1. Example of Parity Calculation

If a drive fails, the array regenerates the missing data by determining the appropriate value (0 or 1) of each missing bit. For example, if Drive 2 fails, the Bit 0 value of the missing data is determined by solving for 1+ x + 0 + 1 = y, where y represents an odd number (because the parity value for Bit 0 is 1). In this case, Bit 0 on Drive 2 must be 1.

RAID 4

RAID 4 arrays are similar in some respects to RAID 3 arrays, as both levels employ two or more data disks and a dedicated parity drive. In this RAID level, however, the stripe size (or *depth*) is sufficient (block level versus byte level for RAID 3) to accommodate an entire record. This allows independent reads of stored information, making RAID 4 arrays well suited for transactional environments that require many small, simultaneous reads. The same write bottleneck encountered in RAID 3 arrays adversely affects performance of a RAID 4 array. Data recovery performance and costs are similar to that of a RAID 3 array.

RAID 5

This widely used RAID type overcomes some of the drawbacks of other parity-based arrays such as RAID 3 and RAID 4. Parity information for the array's data is distributed among all drives in the array (see Figure 5), instead of being stored on a dedicated parity drive.

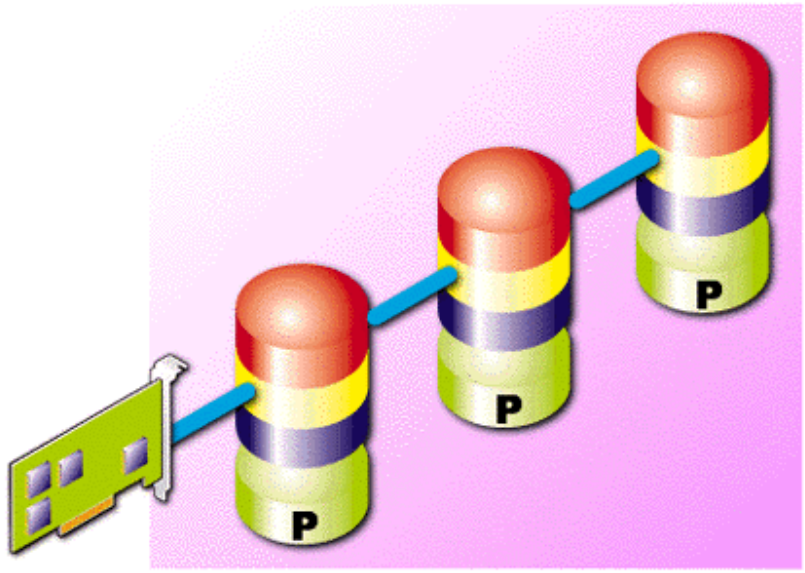

Figure 5. RAID 5 Array With Distributed Parity

This distributed parity approach reduces the write bottleneck common to RAID 3 and RAID 4 because concurrent writes do not always require access to parity information on a dedicated drive. However, overall write performance still suffers because of the overhead caused by reading, recalculating, and updating parity information.

To improve the read performance of a RAID 5 array, the data stripe size (depth) can be optimized for the particular application program using the array. Overall RAID 5 array performance is equivalent to that of a RAID 3 array except in the case of sequential reads, which reduce the efficiency of the drives' read-ahead algorithms because of the distributed parity information.
As in other parity-based arrays, data recovery in a RAID 5 array is accomplished by computing the XOR of information on the array's remaining drives. Because parity information is distributed among all the drives, loss of any drive reduces the availability of both data and parity information until the failed drive is regenerated. This can cause degradation in application program performance for both reads and writes. The cost of a RAID 5 array is comparable to that of RAID 3 and RAID 4 arrays.

RAID 6

The Berkeley researchers in a subsequent 1989 paper proposed a sixth RAID level. RAID 6 carries RAID 5's distributed parity approach one step further by performing two independent parity computations and storing the results on different drives. As a result, a RAID 6 array has the highest reliability of any nonhybrid RAID level. Even if two drives fail, affecting a portion of the data and one set of parity information, the remaining parity information can be used to regenerate the lost data. As in RAID 5, I/O performance is adversely affected while a failed drive is rebuilt.
Read performance is comparable to that of a RAID 5 array. However, the complexity of this array type is higher when data is written to the array because two parity calculations and writes must be performed. In addition, costs for RAID 6 arrays are relatively high because space must be allocated for two sets of parity information. These factors have limited use of RAID 6 arrays to storage systems that place a premium on data accessibility.

Hybrid RAID Levels

Since the 1980s' publication of the Berkeley RAID papers, many storage developers have created hybrid RAID levels combining features of the original RAID levels. Three of the most common hybrid levels are RAID 10, RAID 30, and RAID 50.

RAID 10

RAID 10 is a multilevel array that, as its name implies, combines mirrored drives (RAID 1) with data striping (RAID 0).
The particular method of creating a RAID 10 array varies. In a RAID 0+1 implementation, data is striped across mirrored sets of drives, as shown in Figure 6. (This arrangement is known as a stripe of *mirrors* .) In a RAID 1+0 implementation, data is striped across several drives, and this complete array of drives is mirrored by one or more arrays of drives. (This latter configuration can be termed a *mirror of stripes* .)

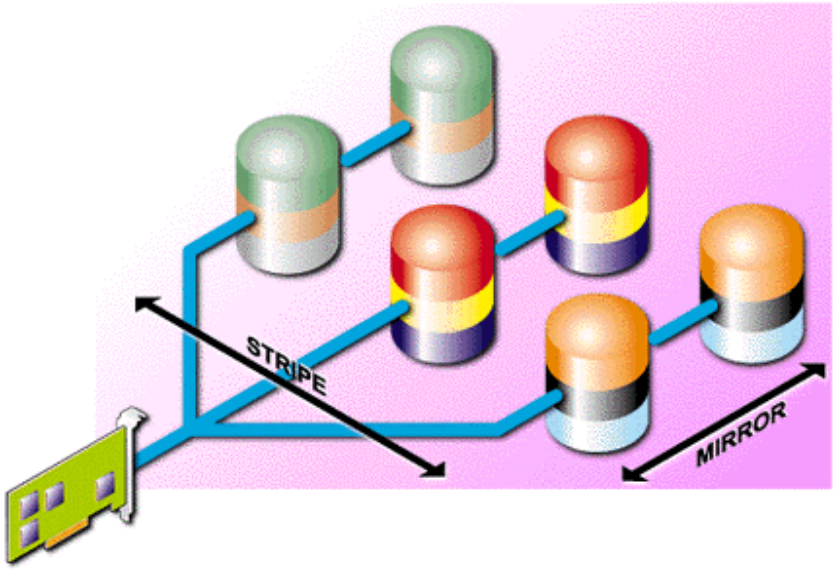

Figure 6. RAID 10 Array
RAID 10 offers the data transfer advantages of striped arrays and the data accessibility features of mirrored arrays. System performance during a drive rebuild is also better than that of parity-based arrays, since data does not need to be regenerated from parity information, but simply copied from a surviving drive. The cost of a RAID 10 array is the same as that of a RAID 1 array, as both arrays require that at least two complete sets of data be

stored.

RAID 30 and RAID 50

RAID 30 and RAID 50 are hybrid RAID levels that combine parity RAID techniques with data striping. A RAID 30 or RAID 50 array is essentially an array with information striped (RAID 0) across two RAID 3 or RAID 5 arrays (see Figure 7).

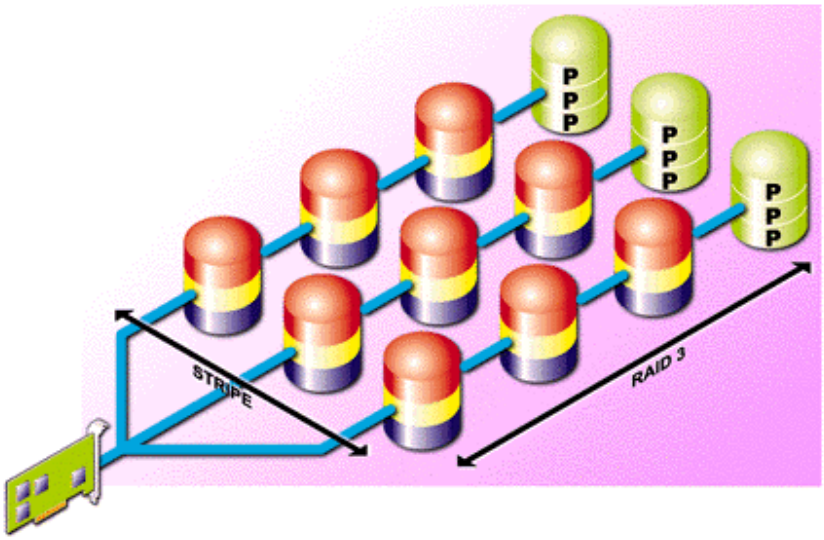

Figure 7. RAID 30 Array

Depending on the stripe size established during array configuration, these hybrid arrays can provide the benefits of parallel-access parity arrays (high data transfer rates) or independent-access parity arrays (high throughput). As with other parity RAID arrays, rebuilding a failed drive impacts program performance. Costs are higher than with RAID 3 or RAID 5 arrays because multiple parity drives (one drive per RAID 3 or RAID 5 array component) are allocated to parity information.

Table 2 compares attributes of the various RAID levels.

RAID Level	Data Availability	Read Performance	Write Performance	Rebuild Performance	Minimum Number of	Appropriate Uses
RAID 0	None	Very good	Very good	N/A	N	Noncritical data
RAID 1	Excellent	Very good	Good	Good	2N x X (X =number of	Small databases, database logs.
RAID 2	Good	Very good	Good	Good	N + 1	Requires proprietary drives.
RAID 3	Good	Sequential I/O: Very Good	Sequential I/O: Good	Fair	N + 1 (N = at least 2)	Single-user data-intensive
RAID 4	Good	Sequential I/O: Good	Sequential I/O: Very Good	Fair	N + 1 (N = at least 2)	Databases, other read-intensive
RAID 5	Good	Sequential I/O: Good	Fair unless write-back cache is	Poor	N + 1 (N = at least 2)	Databases, other read-intensive

		Transactional I/O: Very Good	used			transactional uses
RAID 6	Excellent	Very good	Poor	Poor	N + 2	Small- and medium-sized highly available databases
RAID 10	Excellent	Very good	Fair	Good	2N x X (X = number of RAID sets)	Data-intensive environment (large records)
RAID 30, RAID 50	Excellent	Very good	Fair	Fair	N + 2 (N = at least 4, X = number of RAID sets)	Medium-sized transactional or data-intensive uses
* N = Storage space requirements of data						

Table 2. Comparison of RAID Levels

Implementation of RAID at the System Level

Storage system developers have followed different paths to implement the various RAID levels. Like the RAID levels themselves, these implementations have advantages and disadvantages. Cost, complexity, system performance, and effect on data accessibility should all be considered.

Host-Based RAID Storage Systems

Host-based RAID storage systems are servers that incorporate software- or hardware-based control of the RAID array within the host system itself. Although their drive capacity is somewhat limited (typically two to eight drives), these systems can be connected to external drive enclosures to add storage capacity. Some variations on host-based RAID storage include the following implementations:

- **Software-based RAID arrays**, which are typically employed in entry-level servers or workstations. Network operating systems, including Microsoft® Windows NT® and Novell® NetWare® , feature software implementations of selected RAID levels such as 0 and 5.

Although software-based RAID can provide cost advantages, it also places demands on the system's microprocessor resources, particularly if data must be regenerated. This resource conflict can adversely affect the performance of application programs running on the host system. However, if an entry-level system is I/O-bound, spare system-processor bandwidth may be available, making software-based RAID a price/performance-effective solution.

- **Embedded RAID controllers** on the server or workstation's system board (also known as RAID On Mother Board), which offer some cost benefits compared to other hardware-based RAID implementations. Because of system board space constraints and cost considerations, most embedded RAID controllers offer a subset of the features offered by more expensive card-based controllers. Some implementations do incorporate intelligent I/O processors to improve data transfer rates and reduce interrupt demands on the system microprocessor. Because these I/O processors are incorporated in the system board, they can be used for non-RAID-related tasks as well.
- **PCI card-based RAID controllers**, which offer flexibility and scalability as storage requirements change and can be duplexed for improved fault tolerance. Some RAID controller cards are designed as low-cost alternatives to software-based RAID, while others offer advanced features such as multiple channels for system scalability, hardware-based XOR to improve parity-based array performance, and battery-backed cache memory to support write-back data writes for improved data integrity and I/O performance. (For more on write-back caching, see "RAID System Performance Factors" found later in this document.) As advances in these controller cards are introduced, workstations and servers can be easily upgraded.

External RAID Storage

External RAID storage systems are enclosures of hard-disk drives that are managed by one or more integrated controllers. Available in freestanding and rack-mounted configurations, these storage systems offer a high degree of flexibility and scalability, but at a higher cost due to the additional enclosures, host adapters, and other components required. One or more external storage systems may be connected to a server or workstation, or a single storage system may serve multiple host systems in a clustering environment.

RAID and Fault-Tolerant Systems

Although RAID technology protects against data loss due to hard-disk drive failure, system downtime due to

other component problems also impacts data availability. *Fault-tolerant systems* are designed to reduce the impact of single-point failures by incorporating some or all of the following measures:

- Connecting systems to an uninterruptable power supply (UPS) or multiple power circuits to alleviate site power-source problems.
- Providing redundant, hot-swappable power supplies and cooling fans.
- Providing redundant controllers to reduce the impact of a controller or data path failure.
- Clustering servers to achieve the highest degree of fault tolerance by providing redundancy for the server itself, the network operating system, and the application programs that access the RAID array. Some cluster architectures allow sharing of RAID array resources by different host servers, while other architectures only transfer storage resources if a server fails.

RAID System Performance Factors

In addition to the method chosen for system-level RAID implementation, specific hardware and software configuration decisions can affect overall RAID system performance.

Drives

Hot-spare hard-disk drives can improve data availability if the system hardware and storage management software support them. These drives can be safely removed or installed in the array without shutting down system power, thus maintaining data availability. In the event of drive failure, the controller can automatically rebuild data (provide *automatic failover*) from the failed drive onto a hot-spare drive. Some array management software also allows dynamic expansion of the array onto hot-spare drives.

Self-Monitoring Analysis and Reporting Technology (SMART) drives can also reduce system downtime. These drives analyze various internal drive components and overall performance. The host system software can then alert the storage system management software if a drive failure is imminent, allowing timely replacement of the drive. Note that SMART is effective in predicting certain types of drive failures but does not anticipate all drive failure modes. Hence, SMART does not obviate the need for RAID data protection.

Controllers

RAID controller support for read caching or write caching of data can improve array performance. However, the type of caching must be matched to the application programs used with the array.

With *read-ahead caching* , data specified by a read request is read, along with a portion of the succeeding data on the drive. This additional data is stored in cache memory on the controller. If a subsequent read request can be fulfilled by the cached data, access to the drive is avoided and the data is retrieved at the speed of the system I/O bus. This technique is well suited for applications such as video image processing that store data in large sequential records. However, read-ahead caching actually impairs performance of random-access applications, such as transactional or database applications, because little of the data read for caching purposes is used. Some controllers can be manually configured to enable or disable read-ahead caching, while other intelligent controllers enable read-ahead caching only if I/O conditions are appropriate for its use.

Write caching is particularly beneficial for RAID 4 and RAID 5 arrays because it alleviates the write penalty inherent in these RAID levels. RAID array controllers use two distinct types of write caching. With *write-through caching* , the controller does not acknowledge the completion of the write operation until the data is written to the drive.

In *write-back caching* , the controller signals that the write request is complete after data is stored in the cache but before it is written to the drive. This improves performance relative to write-through caching because the application program can resume while the data is being written to the drive. However, if system power is interrupted, any information in the cache is lost. This risk is magnified in parity RAID arrays because both data and parity information are updated. For this reason, system power backup through use of a UPS or battery backup for the controller itself is necessary to ensure data integrity of the array. Many high-performance PCI and external RAID controllers provide battery backup cache, so when system power is restored, the data write is completed to the drives in the RAID array.

Effect of Drive Rebuilds

As noted in the preceding descriptions of the various RAID levels, RAID system performance degrades following drive failures because of the overhead associated with rebuilding or regenerating data on the affected drive(s). In some cases, RAID controller performance can decrease by as much as 50 percent. In a heavily loaded system, this decrease impacts overall system performance during drive rebuilds. Consequently, system I/O loading should be planned so it does not exceed the RAID controller performance in rebuild mode, unless reduced system performance is acceptable in the specific application program.

To reduce the effect of drive rebuilds on system performance, many high-performance PCI and external RAID

controllers allow the relative priority of rebuilds to be specified through configuration software. Setting this priority involves a trade-off between overall performance and data redundancy. Fast drive rebuilds restore data redundancy quickly, but at the expense of performance. Slow rebuilds have the least impact on system performance, but the array will remain in a non-redundant state longer.

Array Management and System Management Software

A wide range of software tools are available to enhance RAID array operation and overall system reliability. RAID controllers typically include device-specific *array management software* . This software selects a particular RAID level to match the application programs that will use the array, sets up the new array, manages the array's day-to-day operation, and deals with drive failures. Some array management software allows users to establish priority between drive rebuilds and application program performance. Array management software often includes monitoring features to help tune the array for best performance.

At the system level, *enclosure management software* monitors the status of and reports problems with storage system power supplies, fans, enclosure temperatures, voltages, and drives. Most storage system and controller designers follow the SCSI Accessed Fault-Tolerant Enclosures (SAF-TE) or SCSI Enclosure Services (SES) open standards.

Storage System Attachment Technology

Historically, RAID arrays have employed SCSI protocol to connect hard-disk drives, controllers, and host systems. As the size and complexity of a storage system increases, Fibre Channel technology becomes a viable alternative to the SCSI interface.

Table 3 compares some key attributes of Fibre Channel and SCSI technologies.

	Fibre Channel	SCSI
Transfer rate	100 megabytes (MB)/sec	80 MB/sec (Ultra2/LVD)
Maximum bus length	Copper cable — 30 m Optical cable — 500 m	12 m (Ultra2/LVD)
Capacity	126 devices per loop	16 devices (including controller)

Table 3. Comparison of Fibre Channel and SCSI Technology

Based on the criteria in Table 3, enterprise and data center storage facilities can benefit from Fibre Channel's higher throughput and scalability. SCSI host-based RAID arrays are more appropriate for cost-conscious small- to medium-sized storage installations with limited scalability and bandwidth requirements. For more information on Fibre Channel and other attachment technologies, see the Dell white paper "Storage Area Network (SAN) Technology " (August 1998).

Future RAID Trends

As the costs of system downtime grow, RAID storage will become more pervasive as it continues to spread from corporate data centers to workgroups and small businesses. More importance may be placed on improving the overall fault tolerance of RAID storage systems. The RAID Advisory Board, an organization of RAID product suppliers and consumers, has devised a set of criteria based on levels of fault tolerance. These Extended Data Availability and Protection (EDAP) guidelines can be used to rate the response of a storage system to various internal, external, and environmental failures. Ease-of-use will also influence purchasing decisions.

Because of their strong scalability, fault tolerance, and performance features, Fibre Channel RAID arrays will grow in popularity in enterprise and departmental storage facilities.