Copyright 2000 CSREA Press. Published in the Proceedings of the 2000 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2000), June 26-29, 2000, Las Vegas, Nevada.

HERA: Heterogeneous Extension of RAID*

Roger Zimmermann
Integrated Media Systems Center
University of Southern California
Los Angeles, California 90089
rzimmerm@imsc.usc.edu

Shahram Ghandeharizadeh
Department of Computer Science
University of Southern California
Los Angeles, California 90089
shahram@pollux.usc.edu

Abstract

A number of recent technological trends have made data intensive applications such as continuous media (audio and video) servers a reality. These servers store and retrieve a large volume of data using magnetic disks. Servers consisting of multiple nodes and large arrays of heterogeneous disk drives have become a fact of life for several reasons. First, magnetic disks might fail. Failed disks are almost always replaced with newer disk models because the current technological trend for these devices is one of annual increase in both performance and storage capacity. Second, storage requirements are ever increasing, forcing servers to be scaled up progressively. In this study we present HERA, a framework that enables parity-based data protection for heterogeneous storage systems. Based on the novel Disk Merging approach HERA provides a low cost solution for systems that require highly available secondary storage.

Keywords: Fault-tolerance, Heterogeneity, Storage Systems, RAID

1 Introduction

Applications that utilize digital continuous media, such as video and audio clips, require a vast amounts of storage space [1]. Large archives may consists of hundreds, if not thousands of disks, to satisfy both the bandwidth and storage requirements of the working sets imposed by different applications. Although a single disk is fairly reliable, with a large number of disks, the aggregate rate of disk failures can be too high. At the time of this writing, the mean time to failure (MTTF) of a single disk is on the order of 1,000,000 hours; this means that the MTTF of *some* disk in a 1,000 disk system is on the order of 1,000 hours (approximately 42 days).

With those servers that assume a hierarchical storage structure, a disk failure may not result in loss of data. This is because these systems render the entire database tertiary resident [8, 1, 7]. The disks cache the most frequently accessed objects in order to minimize the number of references to the tertiary. Data redundancy at disk level continues to be important because it is undesirable for a single disk failure to impact all the active displays[1]. Even with redundant data, multiple failures might force the system to terminate some active displays. We use the mean time to service loss (MTTSL) to quantify the fault-tolerant characteristics of the algorithms discussed in this paper.

A common technique to protect against both data and service loss is to add redundancy to the system, either by mirroring data or adding parity information [3, 14]. There is a vast body of literature analyzing techniques in support of homogeneous disk subsystems [5] but very few are concerned with heterogeneous disk subsystems [6, 4]. From a practical perspective, these techniques must be extended to support heterogeneous subsystems. This is due to the current technological trends in the area of magnetic disks, namely, the annual 40% to 60% increase in performance and 40% decrease in storage cost [10]. Consequently, it is very likely that failed disks will be replaced by newer models. In addition, with scalable storage subsystems, a system might evolve to consist of several disk models.

The foundation of HERA that allows it to provide fault-tolerance with a heterogeneous storage system is a technique called Disk Merging [18, 17]. This technique constructs a logical collection of disk drives from an array of heterogeneous disk devices. The logical disks appear homogeneous in both bandwidth and storage capacity to the upper software layers of the system. Figure 1 shows an example storage system with sixteen logical disks that are constructed from six physical disk devices. Note that a logical disk drive might be realized using a fraction of the bandwidth and storage space provided by several physical disk drives. For example logical disk number 2 (d^l_2) in Figure 1 is realized using a fraction of the bandwidth provided by physical disk drives 0, 1, and 2 (d^p_0, d^p_1, and d^p_2). Disk Merging provides effective support for diverse configurations. For example, if a system administrator extends the storage subsystem with a single new disk drive, this paradigm can utilize both its storage space and bandwidth. With the homogeneous view created by Disk Merging, a parity-based data redundancy scheme such as RAID [14] can be applied, as long as it is modified to handle the fol-

*This research was supported in part by the National Science Foundation under grants IRI-9203389, IRI-9258362 (NYI award), and CDA-9216321, a Hewlett-Packard unrestricted cash/equipment gift, and by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, Cooperative Agreement No. EEC-9529152 with additional support from the Annenberg Center for Communication at the University of Southern California and the California Trade and Commerce Agency.

[1] With no data redundancy and a disk failure, depending on the organization of data, some or all requests might be forced to retrieve the missing data from a tertiary device which we assume to have a much lower bandwidth than the aggregate bandwidth of the disk subsystem.

Term	Definition
MTTF = $\frac{1}{\lambda}$	Mean time to failure; mean lifetime of an individual, physical disk with failure rate λ
MTTSL	Mean time to service loss
MTTR = $\frac{1}{\mu}$	Mean time to repair of a physical disk with repair rate μ
d_i^p	Physical disk drive i
d_i^l	Logical disk drive i
p_i	Number of logical disks that map to physical disk i
D	Number of physical disk drives
D^l	Number of logical disk drives
G	Parity group size
$\mathcal{G}_i$	Parity group i; a data stripe allocated across a set of G disks and protected by a parity code
$R(t)$	Reliability function

Table 1: List of terms used repeatedly in this study and their respective definitions.

Disk Series	MTBF (power-on hours)
Hawk 1LP	500,000
Barracuda 4LP	1,000,000
Cheetah 4LP	1,000,000

Table 2: Single disk reliability for three commercial disk drives. In these examples, the manufacturer, Seagate Technology™, Inc. reported the reliability as mean-time-between-failures (MTBF). A simple approximation for MTBF is $MTBF = MTTF + MTTR$ where MTTR denotes the mean-time-to-repair [15, pp 206]. For most practical purposes $MTTR \ll MTTF$ and hence $MTTF \approx MTBF$.

lowing constraint: a physical disk drive should not form multiple logical disks of a single parity group. Otherwise, the failure of this physical disk would result in loss of data, i.e., an entire parity group would become in-operational at the logical level.

In Section 2, we describe analytical models to compute the MTTSL of our technique. Next, Section 3 quantifies the performance tradeoffs associated with HERA compared with RAID on a homogeneous system. These results quantify the qualitative tradeoffs discussed in this section. Conclusions are contained in Section 4.

2 Parity-based Redundancy

To support data intensive applications we assume a multi-node server architecture. Each node has a set of local disks attached to it through an I/O bus, such as the popular *SCSI* (Small Computer System Interface). The nodes are linked with each other through a high-speed interconnection network. While we recognize the importance of investigating both disk failures and node failures, we limit the focus of this paper to only disk failures.

With redundant data, a single failure does not lead to data loss. Furthermore, if the failed disk is replaced with a new device, then the data on the new disk can be rebuilt. In sum, a server can be said to operate in either of three modes: (a) *normal* mode, i.e., all nodes are fully operational, (b) *degraded* mode, i.e., some disk or node has failed, and (c) *rebuild* mode, i.e., the data on a repaired disk or node is being restored. All of these three modes must be considered when designing reliability techniques to mask disk and node failures, such that the service to the applications can be continued.

2.1 Parity Group Assignment

In parity-based systems the disks are partitioned into *parity groups*. The parity information is computed across the blocks in a parity group, most commonly with an XOR function. The large storage space overhead of mirroring is reduced since for a parity group size of G only one $1/G^{th}$ of the space is dedicated to data redundancy. In the basic case, the disks of a storage system are partitioned into non-overlapping parity groups. This scheme can tolerate one disk failure per parity group. However, when a parity group operates in degraded mode, each access to the failed disk triggers the retrieval of blocks from all of the disks within this parity group to reconstruct the lost data. Thus, the load on all operational disks increases by 100% under failure, making this parity group a hot spot for the entire system. To distribute the additional load more evenly, parity groups may be rotated such that they overlap [16]. Further improvements can be achieved by assigning blocks pseudo-randomly to parity groups [13]. To provide a focused presentation we will concentrate on the simple scenario of non-overlapping parity groups.

When the logical disks that were created with the Disk Merging technique are assigned to parity groups, the following constraint must be considered. Some logical disks may map to the same physical device and hence be dependent on each other, i.e., a failure at the physical level may cause multiple logical disks to become unavailable simultaneously. Consequently, two dependent logical disks cannot be assigned to the same parity group. The reason is that with a traditional XOR-based parity computation exactly *one* data block of a stripe can be reconstructed as long as *all the other* blocks of that particular stripe are available. Consequently, the number of independent parity groups D^l/G needs to be larger than any number of logical disks that map to a single physical disk. This can be formally expressed with the following parity group size constraint

$$G \leq \left\lfloor \frac{D^l}{p_i} \right\rfloor \qquad \text{for } 0 \leq i < D \qquad (1)$$

where G denotes the parity group size, D^l represents the total number of logical disks, and p_i denotes the number of logical disks that map to physical disk d_i^p (for example $p_0 = 2.2$ in Figure 1).

Figure 1 shows an example storage system with six

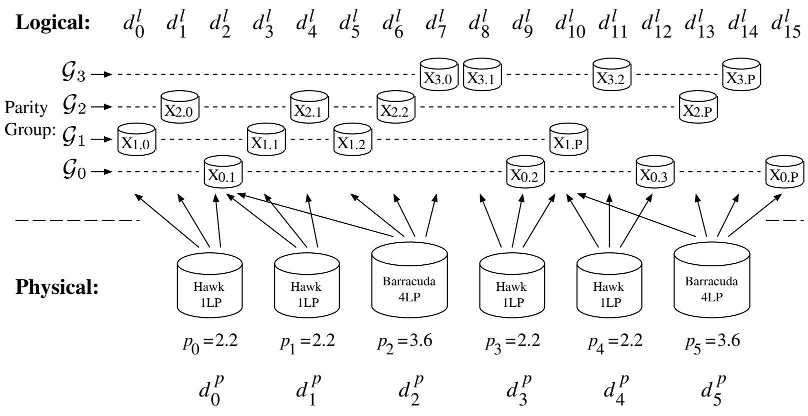


Figure 1: Physical (heterogeneous) and logical (homogeneous) view of a multi-disk storage server employing Disk Merging. Six physical disks are mapped to sixteen logical disks. In addition, a possible parity group assignment to create non-overlapping parity groups is shown. All of a physical disk's logical disks must map to different parity groups ($G = 4$).

physical disk drives that map to sixteen logical disks. The parity group size G is required to be either less than or equal to both $\left\lfloor \frac{D^l}{p_0} \right\rfloor = \left\lfloor \frac{16}{2.2} \right\rfloor = 7$ and $\left\lfloor \frac{D^l}{p_2} \right\rfloor = \left\lfloor \frac{16}{3.6} \right\rfloor = 4$. Hence, the maximum parity group size equals 4 which can be accommodated by creating $16/4 = 4$ or more parity groups $\mathcal{G}_i$. For illustration purposes, we will use a simple, non-overlapping parity group scheme. One possible assignment of the sixteen logical disks $d_0^l, \ldots, d_{15}^l$ to four parity groups $\mathcal{G}_0, \ldots, \mathcal{G}_3$ is as follows (also illustrated in Figure 1): $\mathcal{G}_0 = \{d_2^l, d_9^l, d_{12}^l, d_{15}^l\}$, $\mathcal{G}_1 = \{d_0^l, d_3^l, d_5^l, d_{10}^l\}$, $\mathcal{G}_2 = \{d_1^l, d_4^l, d_6^l, d_{13}^l\}$, and $\mathcal{G}_3 = \{d_7^l, d_8^l, d_{11}^l, d_{14}^l\}$.

From the above parity group assignment we can further determine which physical disks participate in each of the parity groups ("$\mapsto$" denotes "maps to"). Note that the parity groups $\mathcal{G}_0$ and $\mathcal{G}_1$ map to all six physical disks in the sample system.

$$\mathcal{G}_0 \mapsto d_0^p, d_1^p, d_2^p, d_3^p, d_4^p, d_5^p \quad (2)$$
$$\mathcal{G}_1 \mapsto d_0^p, d_1^p, d_2^p, d_3^p, d_4^p, d_5^p \quad (3)$$
$$\mathcal{G}_2 \mapsto d_0^p, d_1^p, d_2^p, d_5^p \quad (4)$$
$$\mathcal{G}_3 \mapsto d_2^p, d_3^p, d_4^p, d_5^p \quad (5)$$

2.2 Basic Reliability Modeling

With the help of data replication or parity coding, the reliability of a disk array can be greatly improved. Several studies have quantified the reliability of homogeneous disk arrays in the context of continuous media servers with mirroring [11], parity coding schemes (RAID) [14, 12, 9, 5, 1], or a combination of both [2]. The concepts of reliability or fault tolerance involve a large number of issues concerning software, hardware (mechanics and electronics), and environmental (e.g., power) failures. A large body of work already exists in the field of reliable computer design and it will provide the foundation for this section. Because it is beyond the scope of this workshop paper to cover all the relevant issues, we will restrict our presentation to the reliability aspects of the disk hardware.

2.2.1 Analytical Model for Reliability

The *reliability function* of a system, denoted $R(t)$, is defined as the probability that the system will perform satisfactorily from time zero to time t, given that it is initially operational [15, pp 7]. When the higher failure rates of a component at the beginning (infant mortality or burn-in period) and at the end (wear-out period) of its lifetime are excluded, then there is strong empirical evidence that the *failure rate* λ during its normal lifetime is approximately constant [15, 9]. This is equivalent to an exponential distribution of the product's lifetime and gives rise to a reliability function $R(t)$ that can be expressed as:

$$R(t) = e^{-\lambda t} \quad (6)$$

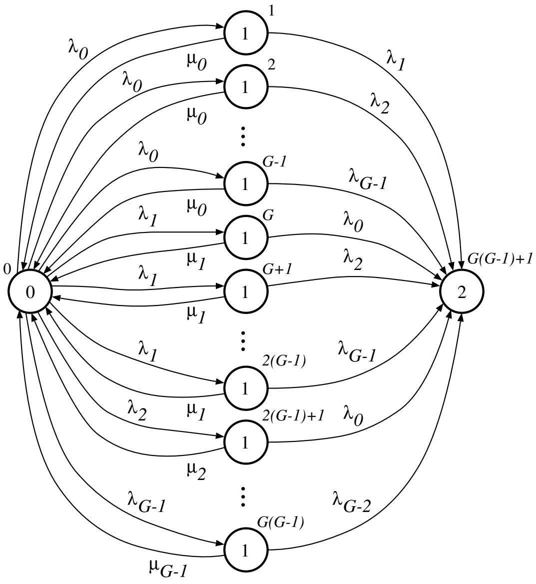
Perhaps the most commonly encountered measure of a system's reliability is its mean time to failure (MTTF) which is defined as follows:

$$MTTF = \int_0^\infty R(t)\,dt = \int_0^\infty e^{-\lambda t}\,dt = \frac{1}{\lambda} \quad (7)$$

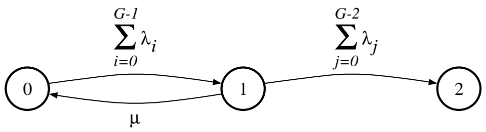
In a heterogeneous storage environment it is possible for each physical disk drive d_i^p to have its own $MTTF_i$ and hence its own failure rate λ_i (see Table 2).

The mean lifetime of a system can be greatly prolonged if it contains redundant components that can be repaired when a failure occurs. This is the case for parity based storage systems, where a parity group can run in degraded mode until a failed disk is replaced and its data is recovered. The mean time to repair (MTTR)[2] is often difficult to model analytically, and it must usually be estimated or

[2]It is customary to refer to MTTR as the mean-time-to-repair even though for most practical purposes a failed magnetic disk will be replaced and not repaired. In such a case the MTTR should include the time to rebuild the lost data on the new disk.



(a) Complete Markov model.



(b) Simplified Markov model.

Figure 2: Markov models for a heterogeneous disk array (one parity group). The labels in each state denote the number of disk failures encountered by the array. State 2 is a trapping state, i.e., the probability of exiting is zero, meaning the array has failed.

measured [15, pp 205]. If the operating environment of the system is known, then an estimate can be given. For example, if spare disks are at hand and service personnel is part of the staff at the site location, a MTTR of a few hours should be realistic. If spare parts must be ordered or service personnel called in then the MTTR may be in the order of days or even weeks. For the analytical models the repair rate is commonly denoted μ and for exponential distributions $MTTR = \frac{1}{\mu}$. In a heterogeneous environment the repair rate may not be the same for all disk types, hence in the most general case each disk has its own repair rate μ_i.

2.2.2 Markov Model for a Single Parity Group

Markov models provide a powerful tool for basic reliability modeling if the system is composed of several processes (such as a failure process and a repair process) [15, 9]. Each component of such a model is assumed to be independent. However, logical disks produced by the Disk Merging technique are not a priori independent, because several of them may map to the same physical disk drive. Conversely, fractions of a single logical disk may be spread across several physical disks. In Section 2.2.3 we will derive the mean time to failure for each individual logical disk in a storage system, such that the necessary independence for the Markov model is preserved.

Figure 2(a) shows the Markov model for a single parity group comprised of G independent disk drives. The states are labeled with the number of disk failures that the array is experiencing. For example, in state 0 all disks are operational. Then with probability λ_i disk d_i becomes unavailable and a transition is made to one of the states labeled 1. With probability μ_i repairs are completed and the array is again in state 0. Or, with probability λ_j a second disk fails and the transition to state 2 indicates that an unrecoverable failure has occurred. As illustrated, the number of states of the model is $(G \times (G-1)) + 2$, i.e., it is quadratic with respect to the parity group size G. The evaluation of such a model is computationally quite complex, especially for larger values of G. Hence we propose the following two simplifying assumptions:

1. The repair rate μ_i is the same for all the disks in the system, i.e., $\mu = \mu_0 = \mu_1 = \ldots = \mu_{G-1}$. It is likely that the time to notify the service personnel is independent of the disk type and will dominate the actual repair time. Furthermore, disks with a higher storage capacity are likely to exhibit a higher disk bandwidth, leading to an approximately constant rebuild time.

2. The probability of a transition from any of the states 1 to state 2 is $\sum_{j=0}^{G-1} \lambda_j$ minus the one failure rate that led to the transition from state 0 to 1. We propose to always subtract the smallest failure rate λ_{min} to establish a closed form solution. Hence, by ordering (without loss of generality) λ_j according to decreasing values, $\lambda_0 \geq \lambda_1 \geq \ldots \geq \lambda_{G-1}$, we can express the probability of a transition from state 1 to 2 to be $\sum_{i=0}^{G-2} \lambda_j$. This approximation will lead to a conservative estimate of the overall MTTF of a system.

With these assumptions the Markov model is simplified to three states as shown in Figure 2(b) and can be solved using a set of linear equations and Laplace transforms [15, 9, 17]. The resulting mean lifetime of a single parity group of independent heterogeneous disks is shown in Equation 8.

$$MTTSL = \frac{\mu + \sum_{i=0}^{G-1} \lambda_i + \sum_{j=0}^{G-2} \lambda_j}{\sum_{i=0}^{G-1} \lambda_i \times \sum_{j=0}^{G-2} \lambda_j} \tag{8}$$

Under most circumstances the repair rate $\mu = 1/MTTR_{disk}$ will be much larger than than any of the failure rates $\lambda_i = 1/MTTF_{disk_i}$. Thus, the numerator of Equation 8 can be simplified as presented in Equation 9 (see [9, pp 141]).

$$MTTSL = \frac{\mu}{\sum_{i=0}^{G-1} \lambda_i \times \sum_{j=0}^{G-2} \lambda_j} = \tag{9}$$

$$\frac{1}{\sum_{i=0}^{G-1}\frac{1}{MTTF_{disk_i}}\times\sum_{j=0}^{G-2}\frac{1}{MTTF_{disk_j}}\times MTTR_{disk}}$$

If all the failure rates λ_i are equal, i.e., $\lambda = \lambda_0 = \lambda_1 = \ldots = \lambda_{G-1}$ then Equation 9 will, happily, correspond to the derivation for a homogeneous disk RAID level 5 array with group size G [14]

$$MTTSL_{RAID5} = \frac{MTTF_{disk}^2}{G \times (G-1) \times MTTR_{disk}} \quad (10)$$

where $MTTF_{disk}$ is the mean-time-to-failure of an individual disk, $MTTR_{disk}$ is the mean-time-to-repair of a single disk, and G is the parity group size. Both, Equations 9 and 10, capture the probability of an independent double disk failure within a single parity group.

2.2.3 Logical Disk Independence

The Markov model of the previous section assumes that all the disks are independent. This assumption is not guaranteed for Disk Merging, because one logical disk may map to several physical disks and vice versa. Hence, to be able to apply Equation 9 at the logical level of a Disk Merging storage system, we will need to derive the mean time to failure of each individual, logical disk. Two cases are possible: (1) a logical disk maps to exactly one physical disk and (2) a logical disk maps to multiple physical disks. Consider each case in turn.

If a logical disk maps completely to one physical disk then its life expectancy is equivalent to the lifetime of that physical disk. Consequently, it inherits the mean time to failure of that disk. For example, the mean lifetimes of the logical disks d_9^l and d_{12}^l of Figure 1 are 500,000 hours each.

If, on the other hand, a logical disk depends on multiple physical disks, then it will fail whenever any one of the physical disks fails. Hence, we can apply the harmonic sum for failure rates of independent components [5]

$$MTTF_{d^l} = \frac{1}{\frac{1}{MTTF_{d_{0'}^p}} + \ldots + \frac{1}{MTTF_{d_k^p}}}. \quad (11)$$

As an example, consider applying the above two observations to the the four logical disks in parity group $\mathcal{G}_0$ of Figure 1. This will result in the following mean lifetime for each logical disk

$$\begin{aligned} MTTF_{d_2^l} &= \frac{1}{\frac{1}{500{,}000}+\frac{1}{500{,}000}+\frac{1}{1{,}000{,}000}} \text{ hours,} \\ &= 200,000 \text{ hours} \\ MTTF_{d_9^l} &= 500,000 \text{ hours,} \quad (12) \\ MTTF_{d_{12}^l} &= 500,000 \text{ hours,} \\ MTTF_{d_{15}^l} &= 1,000,000 \text{ hours.} \end{aligned}$$

Recall that, if multiple logical disks map to the same physical disk, then a failure of that physical drive will concurrently render all its logical drives unavailable. For the aforementioned reason all logical disks that map to the same physical disk must be assigned to different parity groups (see Figure 1).

2.2.4 Multiple Parity Groups

Large storage systems are typically composed of several parity groups. All of the groups need to be operational for the system to function properly. If the groups are assumed to be independent, then the system can be thought of as a series of components. The overall reliability of such a configuration is

$$R_{Series}(t) = \prod_{i=1}^{\frac{D}{G}} R_i(t) \quad (13)$$

where D/G is the number of parity groups and $R_i(t)$ is the individual reliability function of each group.

The overall mean lifetime $MTTSL$ of a series of groups can then be derived from the harmonic sum of the individual, independent failure rates of all the components as shown in Equation 14 [5].

$$MTTSL_{System} = \frac{1}{\frac{1}{MTTSL_1} + \ldots + \frac{1}{MTTSL_{\frac{D}{G}}}} \quad (14)$$

Consider the simple example shown in Figure 1. Four parity groups are formed from 16 logical disks, which in turn have been constructed from 6 physical devices. The physical $MTTF_{disk}$ are assumed to be 500,000 hours (d_0^p, d_1^p, d_3^p, and d_4^p), respectively, 1,000,000 hours (d_2^p and d_5^p), corresponding to Hawk 1LP and Barracuda 4LP devices (see Table 2). Each logical disk inherits its $MTTF_{disk}$ from the physical drive it is mapped to. For example, $MTTF_{d_0^l} = 500,000$ hours.

The resulting mean lifetimes for both, group $\mathcal{G}_0$ and $\mathcal{G}_1$, are $MTTSL_{\mathcal{G}_0} = MTTSL_{\mathcal{G}_1} = 211,422$ years, while the $MTTSL_{\mathcal{G}_2}$ and $MTTSL_{\mathcal{G}_3}$ are 634,237 years each (Equation 8). The mean time to failure for the whole storage system is therefore

$$MTTSL_{System} = \frac{1}{\frac{2}{211{,}422} + \frac{2}{634{,}237}} = 72,282 \text{ years} \quad (15)$$

Such an extraordinarily high mean lifetime will minimize the chance of failure during any time interval.

3 Analytical Results

To evaluate our analytical models we compared their output with the results obtained from a simple reliability simulator that we coded for this purpose. Initially the simulator was provided with a physical and logical storage configuration as well as a a mapping between the two. The simulator then generated random failures based on the $MTTF_{disk}$ of each of the physical disks. A failed disk was assumed to be repaired within a fixed time of $MTTR_{disk} = 6$ hours. The simulator monitored the parity groups consisting of logical disks for two or more concurrent failures. The time between two double-failures (termed *events*) was recorded as the lifetime of the storage system. To obtain an average value a total of 10,000 system failure and recovery cycles were simulated.

Table 3 lists the six different setups for the reliability tests. The physical storage system consisted of a set of eight disk drives. Six different configurations (a) though (f) were realized and tested, all based on the same eight physical disks. All physical disks were assumed to be equal in storage space and bandwidth and accommodate two logical disk drives. Such a simple storage system was chosen to be able to configure it with four different Disk Merging setups and furthermore to allow a direct comparison with a homogeneous RAID 5 setup. The six storage configurations are listed in the order of increasing complexity in Table 3.

Configurations (a) and (b) represented two standard, homogeneous RAID level 5 arrays with a parity group size of 4. They were introduced as a reference and baseline for comparisons with the Disk Merging variants. The difference between (a) and (b) was the assumed reliability of the disks, with $MTTF_{disk} = 1,000,000$ hours and $MTTF_{disk} = 500,000$ hours respectively. For configuration (c) each physical disk with a $MTTF_{disk} = 1,000,000$ hours was split into two independent logical drives for a total of 16 logical disks. Four parity groups of size 4 were formed from these 16 disks. To maximize independence, no two disks of a single parity group mapped to the same physical disk. Configuration (d) was an extension of (c) in that it assumed a reduced mean lifetime of $MTTF_{disk} = 500,000$ hours for half of the physical disks. Configurations (e) and (f) were similar to (c) and (d) but with the additional complexity of mapping four of the logical disks to two physical disks each, i.e., four physical disks were divided into 1.0+0.5+0.5 logical disks. Again, care was taken to maximize independence among the parity groups.

Figure 3 shows the resulting mean lifetimes for each of the configurations (a) through (f). The RAID 5 configuration of Figure 3(a) provides a $MTTSL$ of 6.94×10^9 hours. Because of the quadratic nature of Equation 10, the same RAID 5 system will have a reduced mean lifetime of approximately one quarter if the reliability of the disks is halved, see Figure 3(b). The simulation results for Figures 3(c) through 3(f) show two values each. The lower value assumes that all logical disks are independent. Therefore, a physical disk failure that causes two (or three) parity groups to fail simultaneously is recorded as two (respectively three) separate events. This value should be close to the analytical prediction. The higher value counts multiple related occurrences as just one event, and hence is more indicative of the actual expected lifetime of a system.

The simplest and most reliable Disk Merging configuration, shown in Figure 3(c), achieves approximately an upper/lower reliability of 100%/50% of the $MTTDL$ of the comparable RAID 5 setup of (a). The reason for the analytical reliability of 50% is the increased number of parity groups (4) as compared with the RAID setup (2). Configuration (d) provides a lower reliability because half of the physical disks have a reduced $MTTF_{disk}$ of 500,000 hours. The analytical models for mixed MTTFs of Section 2.2.2 introduced some simplifying assumptions which lead to a conservative estimate of the mean lifetime. Consequently, in Figure 3(d) both simulation results are higher than the analytical predictions. In configurations (e) and (f) a physical disk may be part of up to three different parity groups, which further reduces the overall reliability of the system.

4 Conclusion and Future Directions

In this study we investigated parity-based fault tolerance techniques for heterogeneous storage systems. We introduced HERA, a framework that extends RAID to allow a mix of physical disk drives to be protected from data and service loss. We provided analytical models for the calculation of the mean time to service loss that were verified through simulations. We also provided some design rules for parity group assignments. Collectively, our results show that even though Disk Merging can result in a lower expected lifetime of a system as compared with RAID 5, it is still of the same order of magnitude (i.e., very high). For many applications, however, the slightly reduced reliability of Disk Merging is more than outweighed by its flexibility.

There remain several open research issues within the HERA framework that we plan to address in the future. For example, for a storage system with a large collection of logical disks, an algorithm to map logical disks to parity groups is needed. Also, a configuration planner could be directed to find the best logical to physical disk assignment that produces the most reliable Disk Merging configuration for a given set of physical disk drives.

References

[1] S. Berson, L. Golubchik, and R. R. Muntz. Fault Tolerant Design of Multimedia Servers. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 364–375, 1995.

[2] E. W. Biersack and Ch. Bernhardt. A Fault Tolerant Video Server Using Combined Raid 5 and Mirroring. In *Proceedings of Multimedia Computing and Networking 1997 Conference (MMCN'97)*, pages 106–117, San Jose, California, February 1997.

	Logical Disk Fractions	MTTF	Description
(a)	N/A	All 1,000,000 h	RAID level 5 array for comparison.
(b)	N/A	All 500,000 h	RAID level 5 array for comparison.
(c)	No	All 1,000,000 h	Each physical disk maps to two complete logical disks (1.0+1.0).
(d)	No	500,000 h and 1,000,000 h	Same as (c) but half of the physical disks have a MTTF of only 500,000 hours.
(e)	Yes	All 1,000,000 h	Four of the physical disks map to three logical disks each (1.0+0.5+0.5).
(f)	Yes	500,000 h and 1,000,000 h	Same as (e) but half of the physical disks have a MTTF of only 500,000 hours.

Table 3: Six storage configurations based on 8 physical and 16 logical disks each (no logical disks in case of (a) and (b)) and a parity group size of $G = 4$.

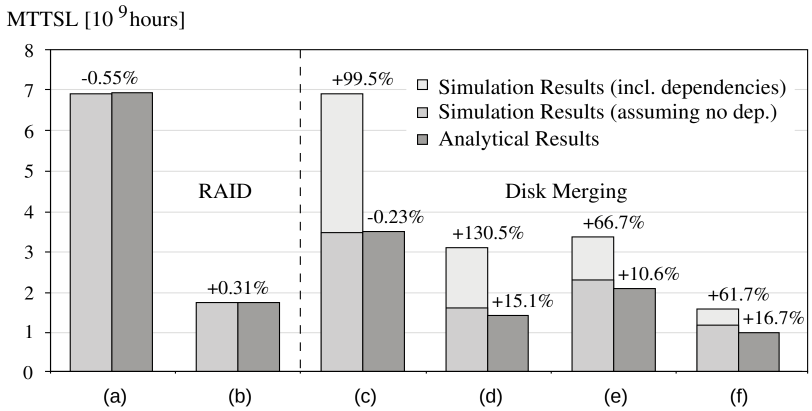


Figure 3: Mean time to data loss for six storage configurations based on 8 physical and 16 logical disks ((c) through (f)) and a parity group size of $G = 4$. See Table 3 for configuration parameters.

[3] D. Bitton and J. Gray. Disk shadowing. In *Proceedings of the International Conference on Very Large Databases*, pages 331–338, September 1988.

[4] L. T. Chen, D. Rotem, and S. Seshadri. Declustering Databases on Heterogeneous Disk Systems. In *Proceedings of the 21st International Conference on Very Large Data Bases*, pages 110–121, Zürich, Switzerland, September 1995.

[5] P. M. Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D. A. Patterson. RAID: High-Performance, Reliable Secondary Storage. *ACM Computing Surveys*, 26(2):145–185, June 1994.

[6] A. Dan and D. Sitaram. An Online Video Placement Policy based on Bandwidth to Space Ratio (BSR). In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 376–385, San Jose, May 1995.

[7] D. J. Gemmell, H. M. Vin, D. D. Kandlur, P. V. Rangan, and L. A. Rowe. Multimedia Storage Servers: A Tutorial. *IEEE Computer*, May 1995.

[8] S. Ghandeharizadeh and C. Shahabi. Management of Physical Replicas in Parallel Multimedia Information Systems. In *Proceedings of the Foundations of Data Organization and Algorithms (FODO) Conference*, October 1993.

[9] G. A. Gibson. *Redundant Disk Arrays: Reliable, Parallel Secondary Storage*. Ph.D. Dissertation, University of California at Berkeley, Berkeley, California, December 1991. Also available from MIT Press, 1992.

[10] E. Grochowski. Internal (media) data rate trend, 1997. IBM Almaden Research Center, San Jose, California. URL: http://www.storage.ibm.com/technolo/grochows/g16.htm.

[11] A. Mourad. Reliable Disk Striping in Video-On-Demand Servers. In *Proceedings of the 2nd IASTED/ISMM International Conference on Distributed Multimedia Systems and Applications*, pages 113–118, Stanford, CA, August 1995.

[12] R. R. Muntz and J. C.S. Lui. Performance Analysis of Disk Arrays Under Failure. In *Proceedings of the 16^{th} Very Large Databases Conference*, pages 162–173, Brisbane, Australia, 1990.

[13] B. Özden, R. Rastogi, P. Shenoy, and A. Silberschatz. Fault-tolerant Architectures for Continuous Media Servers. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 79–90, June 1996.

[14] D. A. Patterson, G. A. Gibson, and R. H. Katz. A Case for Redundant Arrays of Inexpensive Disks (RAID). In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 109–116, June 1988.

[15] D. P. Siewiorek and R. S. Swarz. *The Theory and Practice of Reliable Systems Design*. Digital Press, Bedford, Massachusetts, 1982. ISBN 0-932376-13-4.

[16] R. Tewari, R. P. King, D. Kandlur, and D. M. Dias. Placement of Multimedia Blocks on Zoned Disks. In *Proceedings of IS&T/SPIE Multimedia Computing and Networking*, San Jose, January 1996.

[17] R. Zimmermann. *Continuous Media Placement and Scheduling in Heterogeneous Disk Storage Systems*. Ph.D. Dissertation, University of Southern California, Los Angeles, California, December 1998.

[18] R. Zimmermann and S. Ghandeharizadeh. Continuous Display Using Heterogeneous Disk-Subsystems. In *Proceedings of the Fifth ACM Multimedia Conference*, pages 227–236, Seattle, Washington, November 9-13, 1997.