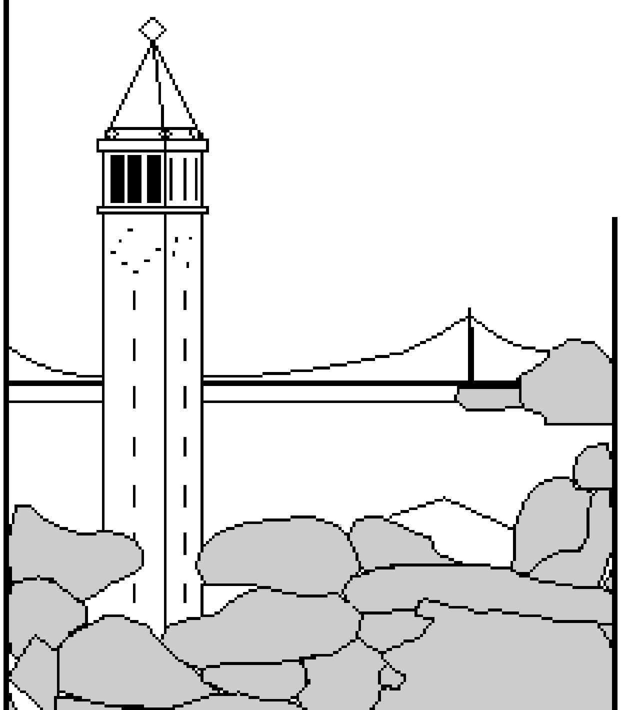
An Analysis of Error Behavior in a Large Storage System

Nisha Talagala, David Patterson



Report No. UCB/CSD-99-1042

February 1999

Computer Science Division (EECS)
University of California
Berkeley, California 94720

An Analysis of Error Behavior in a Large Storage System

Nisha Talagala, David Patterson
{nisha, pattrsn}@cs.berkeley.edu
477 Soda Hall, Computer Science Division
University of California at Berkeley
Berkeley, CA 94720
Phone: 1-(510)-642-1845, Fax: 1-(510)-642-5775
February 1999

Abstract

This paper analyzes the error behavior of a 3.2TB disk storage system. We report reliability data for 18 months of the prototype's operation, and analyze 6 months of error logs from nodes in the prototype. We found that the disks drives were among the most reliable components in the system. We were also able to divide errors into eleven categories, comprising disk errors, network errors and SCSI errors that appeared repeatedly across all nodes. We also gained insight into the types of error messages reported by devices in various conditions, and the effects of these events on the operating system. We also present data from four cases of disk drive failures. These results and insights should be useful to any designer of a fault tolerant storage system.

1.0 Introduction

Packaged storage systems are a billion dollar industry, with more than fifty companies making storage subsystems. The invention of RAID (Redundant Arrays of Inexpensive Disks) in 1980 [1,2] fueled work on reliable storage systems containing many small disks with built in data redundancy. Since then, literally hundreds of papers have appeared in the literature with designs for reliable storage systems. Many of these papers have introduced schemes for achieving even greater reliability than RAID 5 using novel technologies and data layouts [3,4].

Even though the RAID work focused on improving availability, most papers on the subject used simplistic assumptions about component failures, mechanisms and models. For instance, most models used in the literature assume an ideal world with independent component failures, exponential lifetimes, and instantaneous failures. Intuition tells us that these assumptions are simplistic. These simplistic assumptions were made partly because real data on component reliability is hard to come by [1]. Disk manufacturers specify MTTF (Mean Time To Failure) values for their products in millions of hours [5], but actual data on how and why disks fail is guarded jealously by these companies. Also, since end to end availability is what really matters, a truly fault tolerant system must take into account not only disk failure characteristics, but also the reliability of other components like disk controllers, buses, cables, and so on. MTTF values for these components are even harder to come by than failure data on disks.

The past five years have also seen the push to use off-the-shelf hardware in cluster solutions for computing and storage. Work on distributed file systems [6,7] has made it possible to view a cluster of storage nodes as a single system. The availability of this type of architecture is even harder to model than a traditional custom designed hardware storage array since it has even more components and complex interactions between components.

Finally, reliability and performance are usually dealt with separately in storage systems work. This separation comes partly from the assumption that failures are instantaneous and complete. Therefore there is no need to consider the performance implications of partially failed devices. However, recent work by the disk industry on the S.M.A.R.T (Self Monitoring, Analysis and Reporting Technology) interface [8], and prior reliability work [9] suggest that disks, at least, do not fail instantaneously but operate in some degraded mode before final failure.

This paper presents data on the component failure characteristics of a large storage system. We analyze system error logs from a 368 disk, 3.2 terabyte, storage system. To our knowledge, we present the first public in-depth analysis of error data from a storage system of this size. We describe the failure characteristics of disks and SCSI components, the effects of component failures on the operating system of the host machine, and the effects of network failures. We also discuss modes of failure and correlation between failure of different components.

This research is funded by the DARPA Roboline grant N00600-93-K-2481, donations of disk drives and machines from IBM and Intel, and the California State MICRO program

The analysis leads to some interesting insights. We found that the disks drives were among the most reliable components in the system. Even though they were the most numerous component, they experienced the least failures. Also, we found that all the errors observed in six months can divided into eleven categories, comprising disk errors, network errors and SCSI errors. We also gained insight into the types of error messages reported by devices in various conditions, and the effects of these events on the operating system. Our data supports the notion that disk and SCSI failures are predictable, and suggests that partially failed SCSI devices can severely degrade performance. Finally, we observed the disastrous effects of single points of failure in our system.

The paper is organized as follows: Section 2 describes our storage system. Section 3 describes the logs used in this study. Section 4 describes the results we have obtained from studying these logs. Section 5 discusses the results and their implications. Section 6 outlines related work and Section 7 concludes with a summary.

2.0 The Storage System

In this section we describe the storage system on which the failure data was collected. This section is intended to give the readers some perspective for the log analysis that follows.

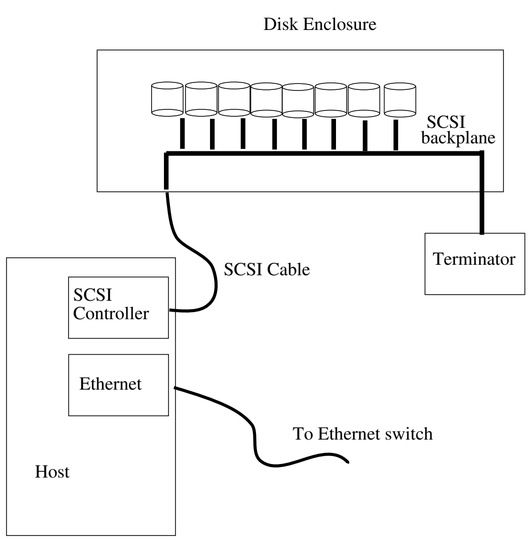


Figure 1: This figure shows the architecture of a node, highlighting the components that will be discussed in this study. For clarity, the figure only shows one SCSI string. The SCSI bus is made up of a cable between the controller and the disk enclosure, and the enclosure backplane. The disk canisters plug directly into this backplane.

The prototype is made up of 20 PCs and 368, 8.4 GB, disks. Each PC hosts a set of disks through SCSI, and the PCs are connected through a switched Ethernet network. The PCs are 200 MHz P6 machines with 96 MB of RAM, running FreeBSD version 2.2. The disks are connected to the host machines using fast-wide SCSI 2 in the single ended mode; twin channel SCSI controllers are used. Four of the 20 nodes host 28 disks each, the remaining 16 host 16 disks each.

Figure 1 shows the internal hardware architecture of a storage node. For clarity, the figure only shows one SCSI string. However, all storage nodes have more than one SCSI string. The machines with 28 disks each have two SCSI strings, with 14 disks per string. These 14 disks are housed in two disk enclosures of 7 disks each. The machines with 16 disks each have two SCSI strings of 8 disks each. Disks plug directly into the enclosure's backplane, which contains the SCSI bus. This design reduces the SCSI cable length within the disk enclosure. The SCSI bus is made up of the SCSI cable, going between the SCSI controller and the enclosure, and the backplane of the enclosure. Each enclosure is powered by two power supplies and cooled with a single fan. Each machine also contains a single Ethernet card and a cable connecting the machine to the switched network. Inside the host PC is a 2GB internal IDE disk.

All operating system software is stored on an internal IDE drive. The data disks are organized as striped disk arrays within each machine; data is mirrored between machines. The machines also mount certain directories from external file servers and rely on external name services. The effects of this external dependency will be discussed in more detail in Section 4. In addition to the switched ethernet, all machines are accessible through serial port connection.

The application for this storage system is a web accessible image collection. The collection is available to users 24 hours a day, 7 days a week. We do not describe the application in detail here, a more detailed description is available in [10]. Thus, most users of this system are external, coming over the internet. They perform only reads. In addition, less than ten other developers also interact with the storage system on a regular basis. These developers do both reads and writes; they manage the image collection, upload and download images, convert images between various formats, and so on.

```
Feb 6 08:09:21 m2 /kernel: (da1:ahc0:0:1:0): SCB 0x85 - timed out while idle,
LASTPHASE == 0x1, SCSISIGI == 0x0
```

Figure 2: A sample line from syslog showing a SCSI TimeOut

3.0 Logs and Analysis Methodology

This section describes how the failure data for this storage system was collected and analyzed. The raw data was the collection of system logs from the machines in the prototype. The operating system reports error messages, boot messages, and other status messages to the internal system log. The kernel, system daemons, and user processes can contribute to this log using the *syslog* and *logger* utilities [11]. These logs are located at `/var/log/messages` in our configuration of FreeBSD 2.2

We began by filtering out messages that reported status and login information. To this end, we removed all messages from *sshd* (secure shell logins), *sudo* messages, other login messages, and all boot messages. This preprocessing reduced the size of the logs between 30% and 50%. The messages that remained were primarily from the OS kernel and network daemons.

Figure 2 shows a sample error message from a system log, reporting a timeout on the SCSI bus. This log line has seven pieces of information. The first three fields contain the date and time. The fourth field is the machine name, in this case *m2*. The fifth field lists the source of the message; in this case the operating system kernel reporting the error. The sixth field specifies the device on which the timeout occurred. The first two subfields of the sixth field specify the disk number and SCSI bus number within the system; in this case, the error is on the disk *da1* that is attached to SCSI bus *ahc0*. The remainder of the message describes the error; the value of the SCSI Control Block is 0x85, and the device timed out while in the idle phase of the SCSI protocol.

We use the following terms in the rest of the paper to describe our results:

> <u>ErrorMessage:</u> An error message is a single line in a log file, as in Figure 2.
>
> <u>ErrorInstance:</u> An error instance is a related group, or tuple, of error messages. The notion of error tuples has been described in detail in [12]. We used a very simple grouping scheme; error messages from the same error category that were within 10 seconds of each other were considered to be a single error instance.
>
> <u>ErrorCategory:</u> By manually examining the logs, we identified eleven categories of errors. For example, the message above fell into the category "SCSI Timeout". These categories are described in detail in Section 4.2. By searching for keywords in each message, we separated the messages from each category.
>
> <u>ErrorFrequency:</u> An error frequency is the number of error instances over some predefined time period. Section 4.3 presents results on error frequencies.
>
> <u>Absolute failure:</u> An absolute failure is when a component was replaced. An absolute failure is usually preceded by many error instances reported in the log.

4.0 Results

In this section we present the results of the system log analysis. We begin in Section 4.1 by describing absolute failures for eighteen months of prototype operation. This data gives a sense of the reliability of the different components used. In the next sections, the system logs for the last six months of this time period are studied in detail.

Section 4.2 lists and defines all the error categories, the types of error messages that we encountered in the logs. These definitions are used in the remainder of Section 4. Sections 4.3-4.5 report results on six months of log data for 16 of the 20 machines. We were not able to include four nodes in the study because they did not have six months worth of log data. The storage nodes are labeled 1 through 16; nodes 1 through 4 have 28 disks each, and all other nodes have 16 disks each. Section 4.3 describes the frequencies of each error category, within and across machines. The effects of these errors, in particular their relationship to machine restarts, is discussed in Section 4.4. Section 4.5 discusses the correlation between

Component	Total in System	Total Failed (Absolute Failures)	% Failed
SCSI Controller	44	1	2.3%
SCSI Cable	39	1	2.6%
SCSI Disk	368	7	1.9%
IDE Disk	24	6	25.0%
Disk Enclosure	46	13	28.3%
Enclosure Power	92	3	3.26%
Ethernet Controller	20	1	9.8%
Ethernet Switch	1	1	50.0%
Ethernet Cable	42	1	2.3%
Total Failures		34	

Table 1: Absolute failures over 18 months of operation. For each type of component, the table shows the total number used in the system, the number that failed, and the percentage failure rate. Note that this is the failure rate over 18 months (it can be used to estimate the annual failure rate). Disk enclosures have two entries in the table because they experienced two types of problems, backplane integrity failure and power supply failure. Since each enclosure had two power supplies, a power supply failure did not affect availability. As the table shows, the SCSI data disks are among the most reliable components, while the IDE drives and SCSI disk enclosures are among the least reliable.

errors. Finally, section 4.6 presents four case studies of data disk failures.

4.1 Failure Statistics

We begin with statistics on absolute hardware failures for eighteen months of the prototype's operation. Table 1 shows the number of components that failed within this one and a half year time frame. We do not mention the manufacturer name for any component, however all components were the state of the art available in 1996. For each type of component, the table shows the number in the entire system, the number that failed, and the percentage failure rate. Since our prototype has different numbers of each component, we cannot directly compare the failure rates. However, we can make some qualitative observations about the reliability of each component.

Our first observation is that, of all the components that failed, the data disks are the most reliable. Even though there are more data disks in the system than any other component, their percentage failure rate is the least of all components. The enclosures that house these disks, however, are among the least reliable in the system. The disk enclosures have two entries in the table because they had two types of failure, power supply problems and SCSI bus backplane integrity failures. The enclosure backplane has a high failure rate while the enclosure power supplies are relatively more reliable. Also, since each enclosure has two power supplies, a power supply failure does not incapacitate the enclosure. The IDE internal disks are also one of the least reliable components in the system, with a 25% failure rate. The flakiness of the IDE disks could be related to their operating environment. While the SCSI drives are in enclosures specially designed for good cooling and reduced vibration, the IDE drives are in regular PC chassis. Overall, the system experienced 34 absolute failures in eighteen months, or nearly two absolute failures every month.

We note that Table 1 only lists components that failed over eighteen months. The prototype also contains other components that did not fail at all. These components include the PC internals other than the disk: the motherboard, power supply, memory modules, etc. The prototype also contains a serial port hub and cables to each node and this hardware also had no problems over the year.

4.2 Error Types

We now define all the error categories that we observed in the logs. Table 2 lists a sample message for each type of error that we include in this study. While some errors appear as one line in the log, others appear as multiple lines. Definitions of each error category follow.

1. Data Disk Errors
Recall that the data disks are SCSI drives. An error from a data disk usually has three lines. The first line reports the command that caused the error. The second line reports the type of error and the third contains additional information about the error. The messages in the second and third line are defined in the SCSI specification [14]. Although the spec defines many error conditions, we only mention those that actually appeared in the logs.

The Hardware Failure message indicates that the command terminated (unsuccessfully) due to a non-recoverable hardware failure. The first and third lines describe the type of failure that occurred. The Medium Error indicates that the operation was unsuccessful due to a flaw in the medium. In this case, the third line recommends that some sectors be reassigned. The line between Hardware Failures and Medium Errors is blurry; it is possible for a drive to report a flaw in the medium as a Hardware Failure [14]. A Recovered Error indicates that the last command completed with the help of some error recovery at the target. This happens, for instance, if a bad sector is discovered. Drives handle bad sectors by dynamically re-assigning the affected sector to an available spare sector [15]. The table shows such an instance. If more than one recoverable error occurs within a single request, the drive chooses which error to report. Finally, A Not Ready message means that the drive cannot be accessed at all.

2. Internal Disk Errors
The internal disks are IDE Drives. The logs contained two types of errors for IDE drives: soft errors and hard errors. Unlike the SCSI disk errors, these messages are operating system specific. By examining the operating system source code, we learned that soft errors were operations that encountered some form of error but recovered, while hard errors were operations that were not successful after the maximum number of retries. The request information is buried within the error message; for instance, the hard error message in the table occurred while trying to read block number 1970460.

3. Internal vm_fault
This error message appears when the OS kernel attempts to read a page into virtual memory for a process. The error indicates that the read needed to satisfy the page fault did not complete successfully. This error usually causes the affected process to terminate abnormally.

4. Network Errors
Our system reported two types of network errors, those related to the naming (NIS) services and those related to network file system (NFS) services. These errors were reported whenever the system was unable to contact one of these services (i.e., the problem was not in the reporting machine).

5. SCSI Errors
The two SCSI errors are TimeOuts and Parity errors; both are self explanatory. SCSI Timeouts can happen in any of the SCSI bus phases. In our analysis, we don’t separate the SCSI Timeout errors by SCSI BUS phase. By inspecting the OS source, we found that the SCSI driver usually responds to a SCSI timeout by issuing a BUS RESET command. This operation aborts all outstanding commands on the SCSI bus. The other type of SCSI error is Parity. As Table 2 shows, SCSI parity error messages appear as the cause of an aborted request.

4.3 Error Frequencies

Now we analyze the errors that appeared in six months of system logs of 16 of the 20 host machines. These logs are for the last six months of the 18 month period of

Type	Sample Message

Data Disk: Hardware Failure	`May 23 08:00:20 m5 /kernel: (da45:ahc2:0:13:0): WRITE(10). CDB: 2a 0 0 29 de f 0 0 10 0`
	`May 23 08:00:20 m5 /kernel: (da45:ahc2:0:13:0): HARDWARE FAILURE asc:2,0`
	`May 23 08:00:20 m5 /kernel: (da45:ahc2:0:13:0): No seek complete field replaceable unit: 1 sks:80,3`

Data Disk: Medium Error	`Dec 13 00:55:31 m1 /kernel: (da41:ahc2:0:9:0): READ(10). CDB: 28 0 0 71 29 1f 0 0 30 0`
	`Dec 13 00:55:31 m1 /kernel: (da41:ahc2:0:9:0): MEDIUM ERROR info:712935 asc:16,4`
	`Dec 13 00:55:31 m1 /kernel: (da41:ahc2:0:9:0): Data sync error - recommend reassignment sks:80,2f`

Data Disk: Recovered Error	`Jul 24 10:40:09 m0 /kernel: (da73:ahc4:0:9:0): READ(10). CDB: 28 0 0 50 54 cf 0 0 80 0`
	`Jul 24 10:40:09 m0 /kernel: (da73:ahc4:0:9:0): RECOVERED ERROR info:505546 asc:18,2`
	`Jul 24 10:40:09 m0 /kernel: (da73:ahc4:0:9:0): Recovered data- data auto-reallocated sks:80,12`

Data Disk: Not Ready	`May 20 11:14:09 m14 /kernel: (da1:ahc0:0:1:0): WRITE(10). CDB: 2a 0 0 26 2e 6 0 0 10 0`
	`May 20 11:14:09 m14 /kernel: (da1:ahc0:0:1:0): NOT READY asc:40,80`
	`May 20 11:14:09 m14 /kernel: (da1:ahc0:0:1:0): Diagnostic failure: ASCQ = Component ID field replaceable unit: 1`

Internal Disk: Hard Error	`Aug 19 16:43:12 m13 /kernel: wd0h: hard error reading fsbn 1970460 of 1970384-1970511 (wd0 bn 3412252; cn 54162 tn 2 sn 12)wd0: status 59<rdy,seekdone,drq,err> error 40<uncorr>`
Internal Disk: Soft Error	`Aug 19 16:43:14 m13 /kernel: wd0h: soft error reading fsbn 1970461 of 1970400-1970511 (wd0 bn 3412253; cn 54162 tn 2 sn 13)wd0: status 58<rdy,seekdone,drq> error 40<uncorr>`
Internal: VM_fault	`Jul 31 12:12:37 m14 /kernel: vm_fault: pager input (probably hardware) error, PID 15211 failure`
Network Error: NIS	`Nov 20 16:22:13 m17 ypbind[95]: NIS server [128.32.45.124] for domain "td" not responding`
Network Error: NFS	`Nov 20 16:23:10 m17 /kernel: nfs server stampede:/disks/stampede/sandbox1: not responding`

SCSI: Parity	`May 12 01:10:32 m2 /kernel: (da40:ahc2:0:8:0): WRITE(10). CDB: 2a 0 0 b9 54 cf 0 0 50 0`
	`May 12 01:10:32 m2 /kernel: (da40:ahc2:0:8:0): ABORTED COMMAND asc:47,0`
	`May 12 01:10:32 m2 /kernel: (da40:ahc2:0:8:0): SCSI parity error`

SCSI TimeOut	`May 17 02:14:58 m0 /kernel: (da33:ahc2:0:1:0): SCB 0x61 - timed out while idle, LASTPHASE == 0x1, SCSISIGI == 0x0`

Table 2: This table lists the categories of errors that are discussed in this paper and includes a sample message for each type of error. The errors are associated with data disks, internal system disks, virtual memory faults and SCSI problems. The data disks and SCSI error messages are usually part of a message reporting a failed read or write request.

Error Type	Number	% of Total (Including Network Errors)	% of Total (Not Including Network Errors)
Data Disk: Hardware Failure	2	0.29%	0.52%
Data Disk: Medium Error	3	0.43%	0.78%
Data Disk: Recovered Error	10	1.45%	2.60%
Internal Disk: Hard Error	24	3.49%	6.23%
Internal Disk: Soft Error	4	0.58%	1.04%
Internal: VM_fault	6	0.87%	1.56%
Network Error: NFS	43	6.25%	-
Network Error: NIS	260	37.79%	-
SCSI: Parity	129	18.75%	33.50%
SCSI TimeOut	207	30.09%	53.76%
Total	688	100%	-

Table 3: Error frequencies of 17 machines over six months. The table shows the percentages of each error type. Since our network errors were due to a single point of failure that can be removed, the last column shows error frequencies without including network errors.

Table 1. During this time, the system experienced three IDE disk failures and one data disk failure. As mentioned previously, we were unable to include three machines because they did not have six months of log data. Ironically, the logs of two of these machines were destroyed when the IDE internal disks failed. The machine whose data disk failed is not included in this section's data, but the failure is discussed separately in Section 4.6 with other data disk failures The data presented here are based on error instances, all groups of errors that occurred more than 10 seconds apart.

Table 3 shows how frequently each error happened over all 16 machines. 688 error instances were reported, on average, almost 4 errors appeared per day. As the table shows, the network is a large source of error. Together, the NIS and NFS error messages make up over 40% of all error instances over six months. These errors happened because the storage nodes were dependent on external sources for name service and certain NFS mounted file systems. Since the source is external, these errors are also highly correlated between machines (we discuss error correlation further in Section 4.5). This correlation is partly why the number of network errors is so high; one external fault, if it affects all the machines, will be reported as 16 error instances. These services created single points of failure in the system. However, we do not believe that highly available storage systems will have such dependencies. In our system, they were kept only for the convenience of local users. These errors can be removed simply by removing the dependencies. For this reason, we also present the percentage frequencies of other errors without including network errors. Once the network errors are removed, the total number of errors for 6 months was 385, or an average of 2.2 error instances per day.

The largest source of errors in our system are SCSI timeouts and parity problems. SCSI timeouts and parity errors together make up 49% of all errors; when the network errors are removed, this figure rises to 87% of all error instances. Data disk errors, on the other hand, make up a surprisingly small percentage of the total error count, around 4% overall. This happens even though disks make up 90% of the components of the system. Even in these disk errors, the bulk, 3%, are Recovered Errors where the requests did complete successfully. Not all disks on the system are this reliable; IDE disk drives are responsible for over 8% of all reported errors, even though there are only 20 IDE drives in the system. This high count of IDE errors is partly due to a failed IDE disk in machine 8. For the most part, the error percentages match the failure rates in Table 1; SCSI bus failures on enclosures and IDE drive failures make up the bulk of the absolute failures on the system.

Next we break the error information down by machine. Figure 3 shows the categories of errors that each machine experienced. Note that figure 3 does not show the total number of errors reported on each machine; that data is in the accompanying table, Table 4. Not surprisingly, all machines had a share of network errors.

Figure 3 shows that IDE disk errors actually appeared on only 3 machines, machines 5, 8 and 13. Data disk errors also appeared on 8 machines. The figure also shows that 11 of the 16 machines experienced SCSI timeout errors.

Table 4 shows that the error frequencies vary widely between machines. Ten machines reported between ten to 30 error instances, while three of the machines

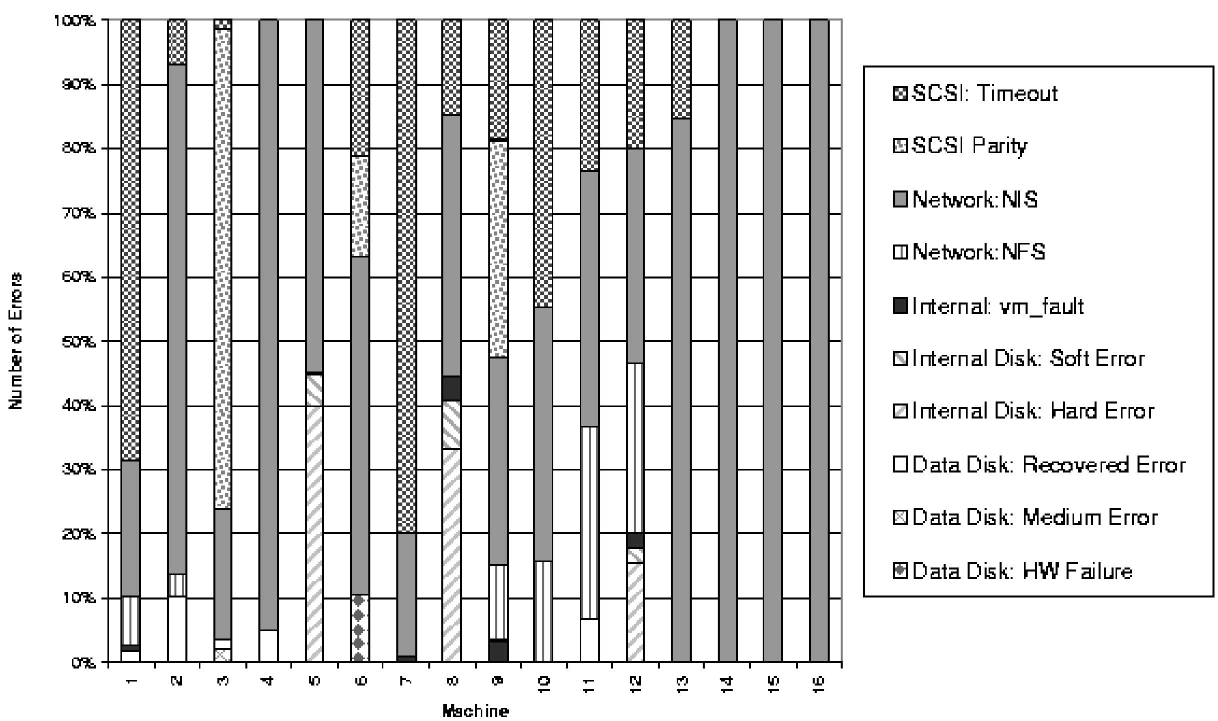


Figure 3: Distribution of errors by machine over a six month period. Each column represents a single machine; the column shows the relative percentages of each error type on that machine. The figure shows that network errors occurred on all machines, but other errors each occurred in two or three of the machines.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Network	31	24	29	19	11	10	18	11	26	21	21	27	11	12	18	14
Others	77	5	113	1	9	9	76	16	33	17	9	18	2	-	-	-
Total	108	29	142	20	20	19	94	27	59	38	30	45	13	12	18	14

Table 4: Total number of errors per machine. This table and Figure 3 together describe how the errors were distributed between machines. The table shows that errors are not evenly distributed; machines 1, 3 and 7 had many more error entries than the others.

reported over 90 errors in the same time frame. Machines 1, 3 and 7 reported the most errors. Figure 3 shows that, in all three cases, the bulk of the messages were all in a single category; for machines 1 and 7 the category was SCSI timeout, while for machine 3 it was SCSI parity. This data suggests an impending failure or other serious problem in each machine. We were able to trace the parity errors in machine 3 to an enclosure replacement that happened later while this paper was being written. There were no SCSI component replacements in machines 1 and 7; this suggests that the problem may have been a loose cable that was later fixed. The SCSI errors in machine 9 also led to a cable replacement while the paper was being written. The only other component replacement that occurred during the six months was the IDE drive on machine 8.

We can make several other observations from this graph and table. First, all machines experienced NIS errors. This behavior is not surprising, since these errors appear when the nodes lose a connection with an external service. If the external service is down, all storage nodes will report the same error. In Section 4.4, we show that NIS errors are heavily correlated between machines. The other type of network error, NFS, does not occur on all machines. This happened because not all machines were mounting the same NFS filesystems at the same time. Second, 10 of the 16 machines reported SCSI timeouts. In this case, the cause was not external; the SCSI subsystems of the machines are independent of each other. Also, the number of SCSI timeouts is not correlated with the number of disks on a node; node seven has a large number of timeouts even though it only hosts 16 disks. Finally, although Table 3 shows that SCSI parity errors have high frequency, Figure 3 shows that almost all of these errors appeared on a single machine, caused by an enclosure failure.

Even though the number of potential problems on a system this large is virtually unlimited, only ten different types of problems occurred over the six months. Another interesting observation is that no type of error was limited to only one machine. SCSI, IDE disk and other errors all occurred on at least two machines. This suggests that even though many combinations of errors can occur in theory on a storage system, there are a small set of problems that can be expected to occur in a given architecture. We can also conjecture that if an error happens once, it may happen again on a different machine.

4.4 Analysis of Reboots

The prior section looked at the errors that appeared in six months of system logs. The real question is though, what are the consequences of these errors? To address this question, we looked at restarts of nodes in the prototype. For each restart that occurred, we checked the prior 24 hours of the system log for any errors that could be related to the shutdown. We used these errors to guess the reason for the restart.

After studying the causes of restarts, we classified the restarts into the following four categories:

> <u>Cold Boot:</u> A Cold Boot is a reboot that occurred without an explicit shutdown or reboot command. All reboots or shutdown commands leave an entry in the system log. When no such entry is present, we assume that the machine was power cycled, either intentionally or because of a power outage. Normally, a machine will not be power cycled unless all attempts to login via network or serial port have failed.
>
> <u>Reboot:</u> A reboot is a restart of a machine with a prior reboot or shutdown command.
>
> <u>Within Maintenance Reboot:</u> This is a reboot that happened within 3 hours of a prior reboot. In this case, we assume that both reboots are part of the same maintenance session.
>
> <u>For Schedulable Maintenance:</u> If an explicit shutdown occurs without any error messages within the prior 24 hours, we assume that the shutdown was for a planned maintenance activity, such as a hardware replacement or upgrade. We call this category Schedulable because we assume that the shutdown could have been moved to another time.

Table 5(a) shows the number of times that each machine was restarted, and Figure 4 shows the percentages of restarts from each category for each machine. This data does not include Within-Maintenance Reboots, since we consider them to have happened while the node was down. Overall, we found that all machines were restarted at least twice in the six months. While most machines had 3-4 reboots, several had 7 to 10 each. There were 73 reboots over all 16 nodes. In addition to schedulable maintenance, we found cold boots with errors, cold boots without errors, reboots with errors, and reboots without errors.

Table 5(b) shows the frequency of each type of restart. Overall, 11% of these reboots were for schedulable

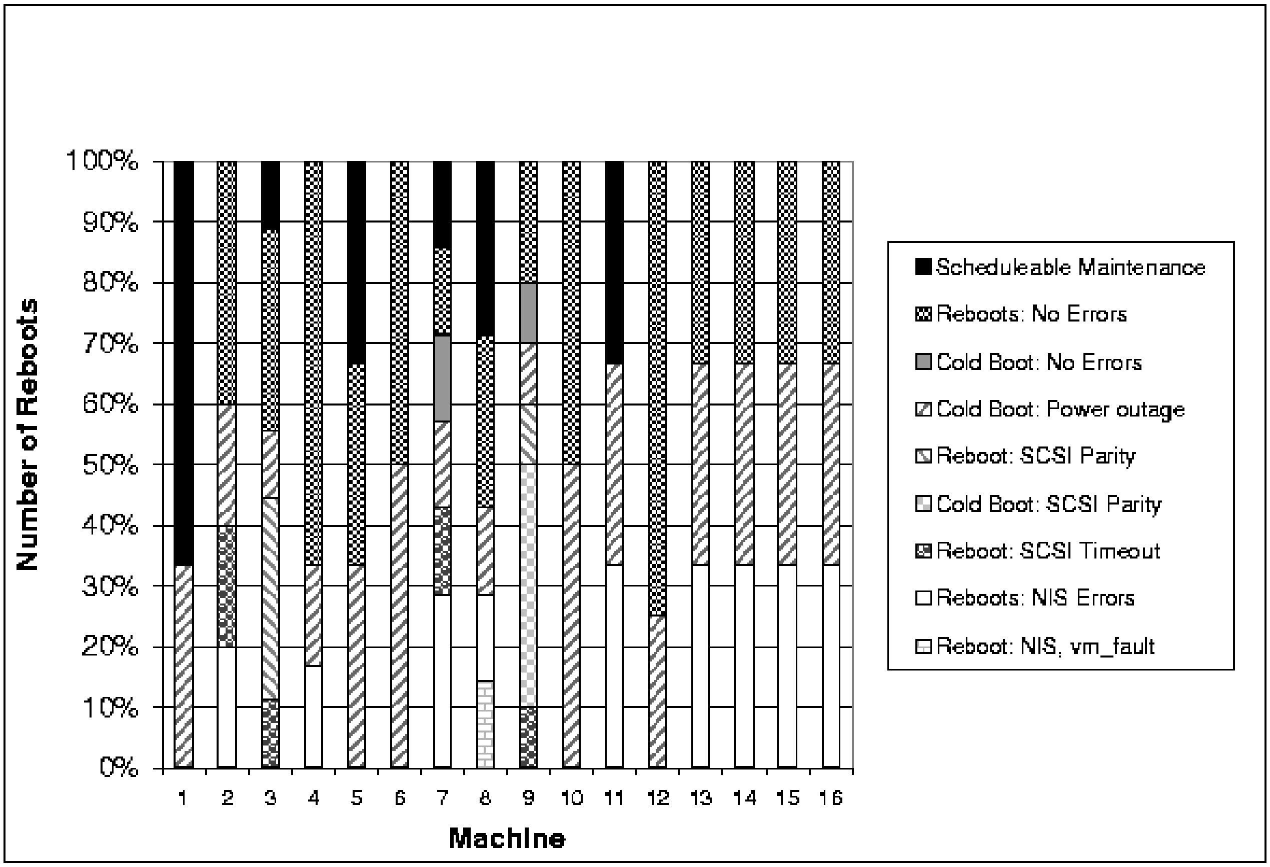


Figure 4: Restarts and their causes. Three types of reboots are shown, Cold Boot (restart with no reboot or shutdown message), Reboot (restart with explicit shutdown or reboot message), and For Schedulable Maintenance (explicit shutdown with no error condition).

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	16	16	17
Restarts	3	5	9	6	3	2	7	7	10	2	3	5	4	3	3	3	3

Table 5a: This table shows how restarts are distributed across machines. Most machines have been restarted between two and three times over the six months, but several machines have been restarted seven to ten times.

Restart Type	Number	% of Total
Schedulable Maintenance	8	11.0%
Reboot: No Errors	24	32.9%
Reboot: Errors	19	26.0%
Cold Boot: No Errors	2	2.7%
Cold Boot: Errors	4	5.5%
Power Outage	16	21.9%

Table 5b: This table shows the frequency of each type of restart.

maintenance; six of the 16 machines had some scheduled maintenance done on them. A single power outage accounts for 22% of all restarts. Another 33% were explicit reboots with no errors in the log; these reboots could have been for software maintenance. It is very unlikely that a machine was explicitly rebooted for no good reason, however we cannot tell from the system logs whether a software upgrade took place. All machines were rebooted without errors. Two machines also received cold boots with no error messages. Finally, the remaining 32% of restarts happened due to errors.

We found only three types of error instances that preceded reboots or cold boots; they were SCSI Timeout, SCSI parity, and NIS errors. Two machines were restarted for SCSI parity problems; one of these is machine 3 that had the failed disk enclosure. Four machines were restarted for SCSI timeout problems. By far, the main cause of reboots and cold boots was NIS errors. All the machines but one were restarted because of network problems. The reason could be that network errors are more fatal to an OS than SCSI errors. While the effects of SCSI errors can be limited to the processes that are reading or writing to the affected drives, the network errors affect all communication between the machine and the outside world.

One interesting point is that no machine restarts happened because of data disk or IDE disk errors. Even though there were hard errors on the three of the 17 system disks, these errors did not cause the operating system to crash. The OS survived hard errors on the internal disk because all of the errors occurred on a user partition that occupied around 80% of the drive.

4.5 Correlations

Sections 4.3 and 4.4 described aggregate data on types of errors and causes of reboots. In this section we examine the time correlation between errors, within and between machines.

Figures 5(a) and 5(b) show the time distribution of errors. The X axis is time and the Y axis is machine numbers. The errors for each machine over time appear on a single horizontal line. A vertical line indicates correlation of errors between machines. Figure 5(a) only

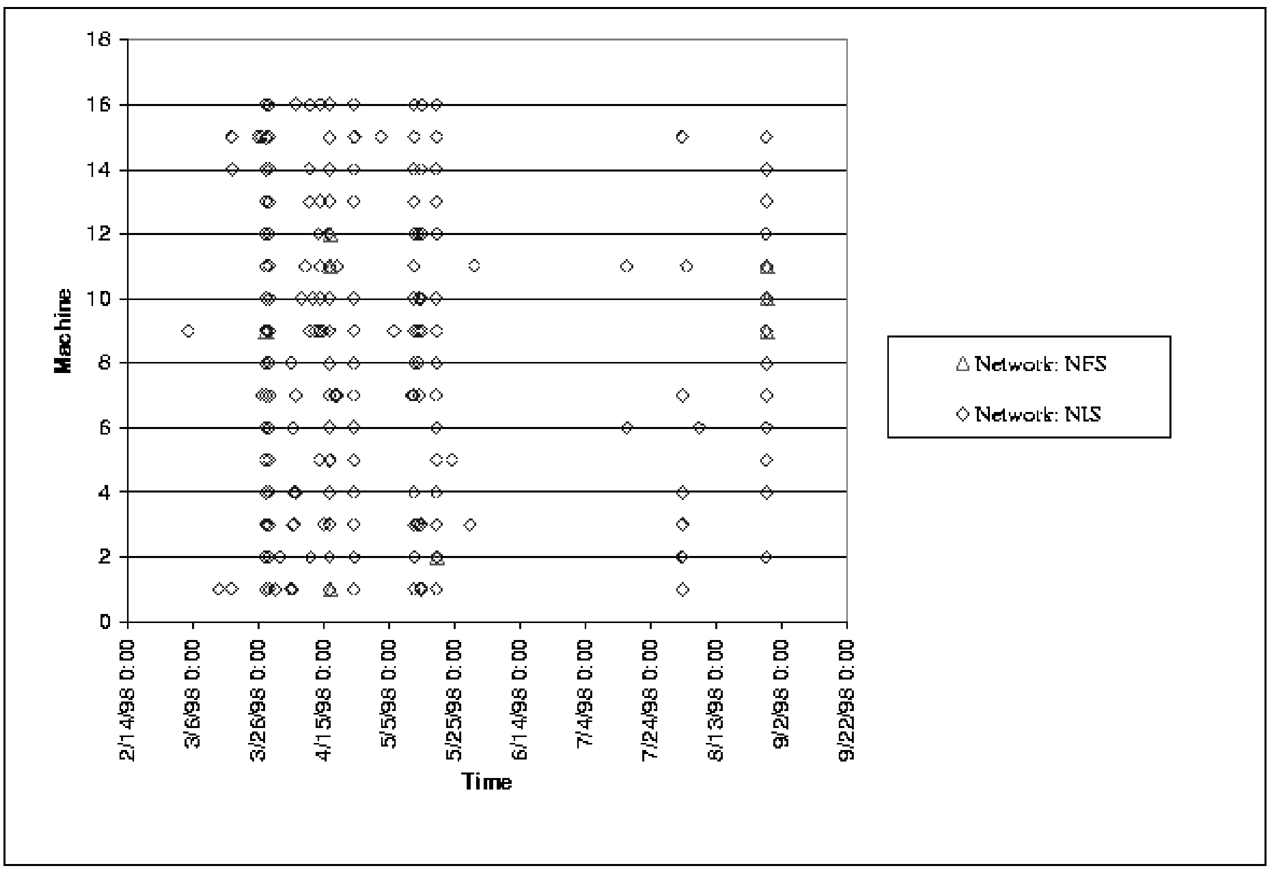


Figure 5(a) Network Errors over time. This figure shows NFS and NIS related errors over time for all 17 machines. The X axis shows time; the errors of each machine are displayed on a horizontal line. The Y axis shows machines. The figure shows that network errors are heavily correlated over machines. This behavior is not surprising as the cause of the errors is an external service.

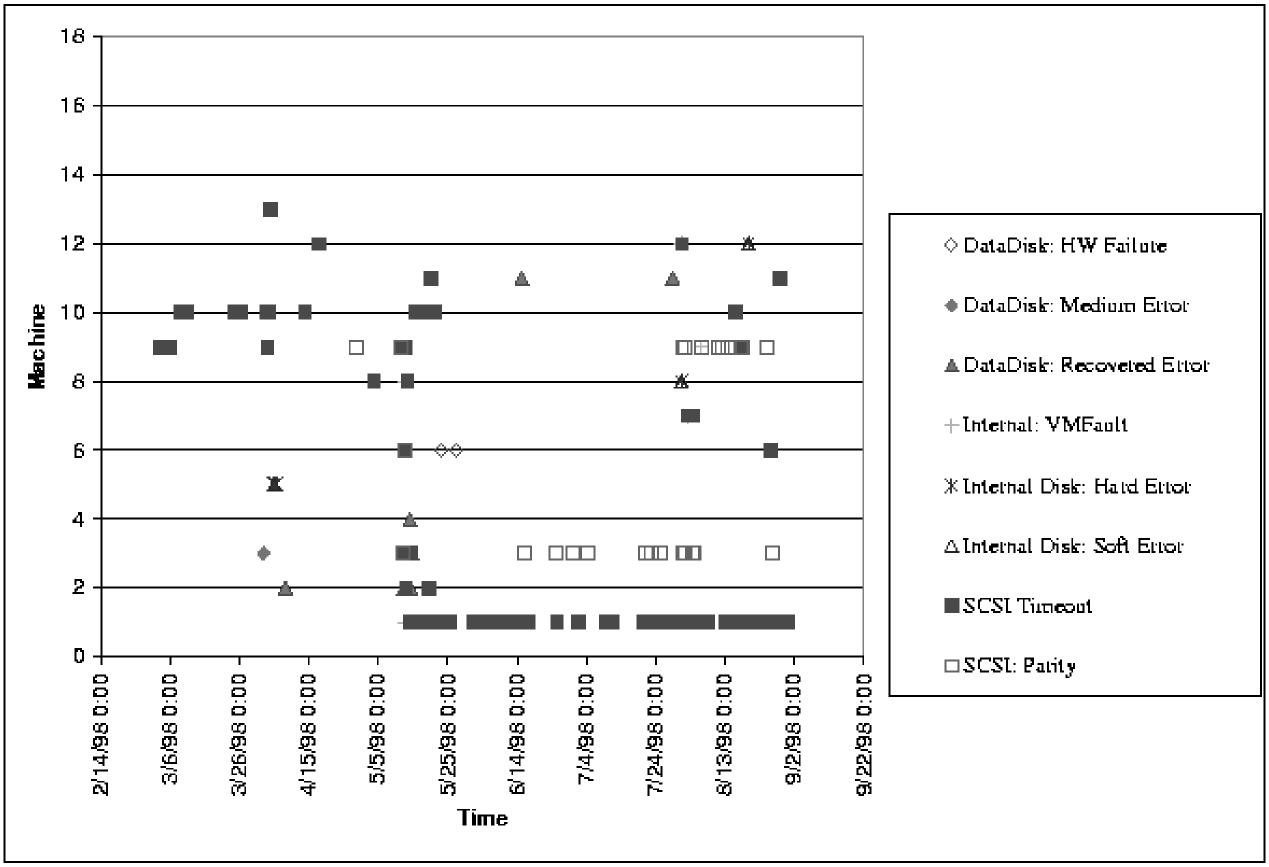


Figure 5(b): Other errors over time. This figure shows the Data Disk, Internal Disk, and SCSI Errors over time. Since there are no shared components between machines, we do not expect these errors to be correlated over time. The figure does show that simultaneous errors do happen.

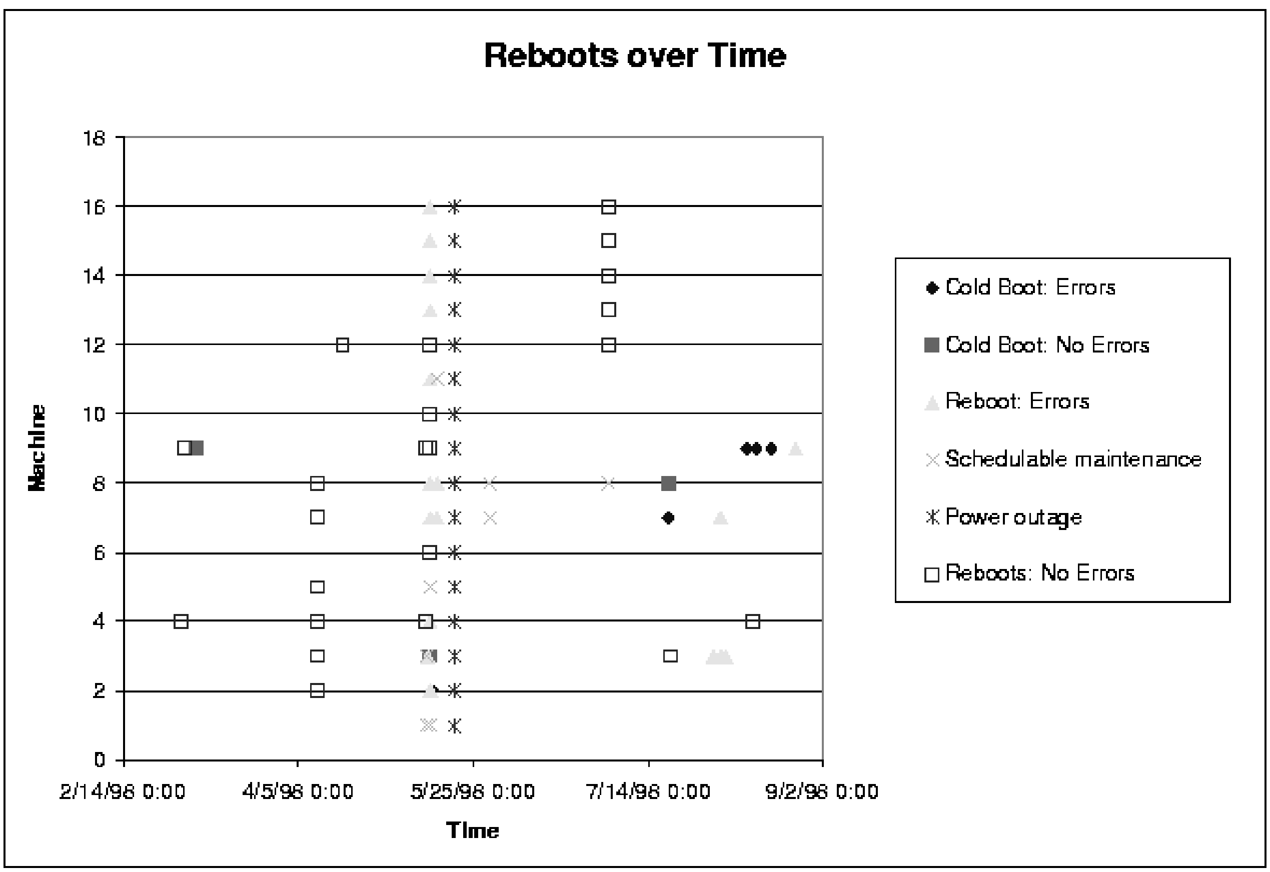


Figure 6: Time Distribution of Reboots. The X axis shows time; each machine's restarts are shown on a separate horizontal line. There are two time instances where nearly all machines were restarted at around the same time. The first is an external network failure. The second is a power outage. The figure also shows that Restarts with No Errors are also correlated between machines. We believe this is because the restarts were part of software maintenance

shows NFS and NIS errors, while Figure 5(b) shows all other errors. It is clear from Figure 5(a) that network errors are correlated between machines. This data reiterates the need to remove all single points of failure from a highly available storage system. The bulk of the errors are NIS errors. When NFS errors occur, they also seem to be correlated with NIS errors.

Figure 5(b) shows all other forms of errors. In this case there is no reason to expect errors on different machines to be correlated; each node is relatively independent of all other nodes. However, the figure shows that even though there is no direct correlation between SCSI errors (no single source), it is possible to have several SCSI errors across different machines at the same time. The figure also suggests that SCSI failures may be predictable; machines 1, 3, and 9 show SCSI parity and timeout errors that escalated over time.

Figures 6(a) and 6(b) shows the time distribution of reboots. The figure indicates that there is a strong correlation between error-free reboots on different machines. This observation further suggests that these reboots were part of software maintenance or upgrade. There are two other heavily correlated groups of reboots between 5/5/98 and 5/25/98. We traced the first back to a NIS service problem. The second was the power outage. All other reboots do not appear to be correlated.

4.6 Case Studies of Data Disk Failure

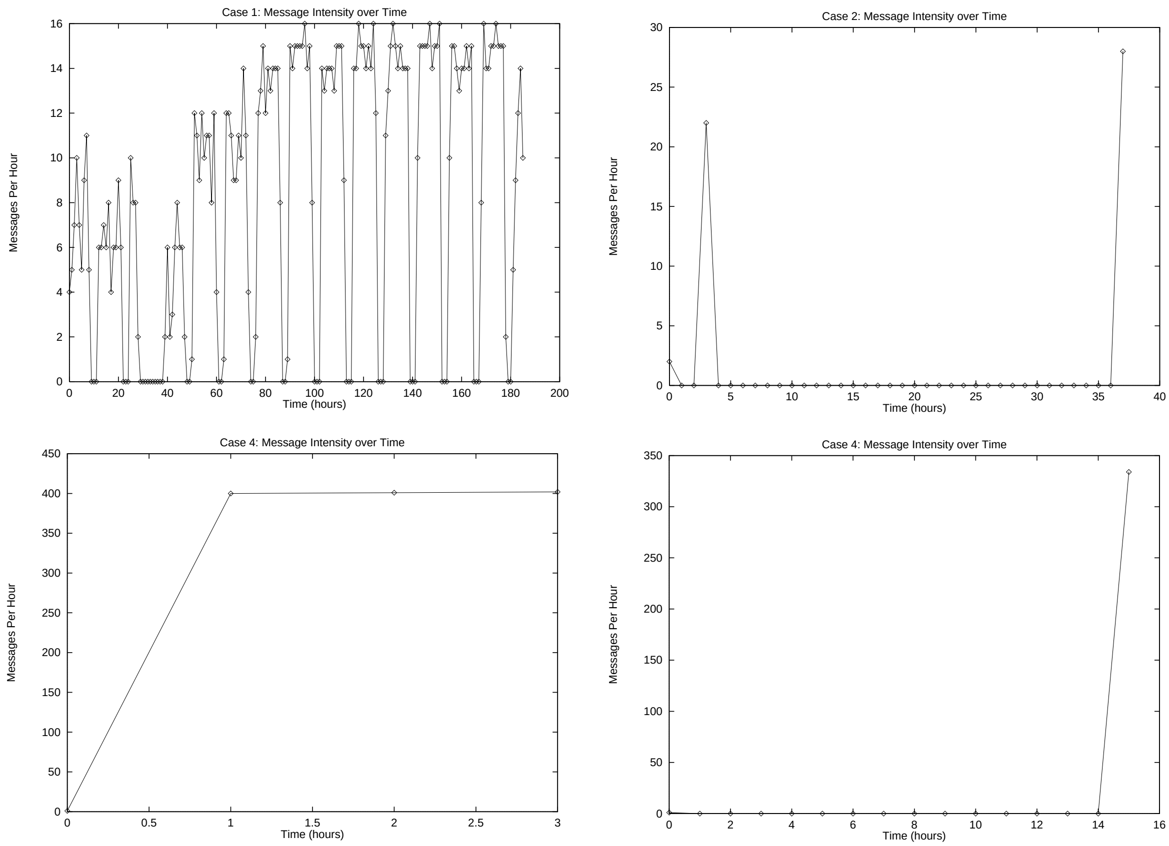
The prior section focused on the failure characteristics of all components except for data disks. For this reason, we focus on data disk failures in this section. We look at four case studies of data disk failures that happened at other times in our system. Even though we had seven disk failures in our prototype, we only had log data for four of the disks at the time of this study. For each case, we study the system log data around the time of the failure and ask the following questions: what types of messages appeared in the log?, for how long did messages appear before the drive was replaced? and did the intensity of these messages increase over time?

Table 3 addresses the first two questions. The table shows the primary and secondary messages that appeared in each case, as well as the time duration of the messages. These messages have already been defined in Section 4.2. The secondary messages are also predefined in the SCSI specification. Like the primary messages, the drive returns a code specifying the secondary error. The messages are fairly self explanatory. The first secondary message, “Peripheral device write fault”, indicates that a fault occurred while writing. “Diagnostic failure” means that diagnostic checks of the device indicated a failure. Finally, “Failure Prediction Threshold Exceeded”, means that the drive maintained statistics and tests indicate that the unit has a unacceptable chance of failure [17,18].

In the first case, the Hardware Failure message appeared at intervals for 186 hours (over 7 days) before the drive was replaced. Therefore, even though the drive was responding, many individual requests were failing. The second case also had error messages over almost two days. These messages, however, were Disk Not Ready,

Case	Primary Message	Secondary Message	Time Duration (hours)
1	Hardware Failure	Peripheral device write fault field replaceable unit	186.0
2	Not Ready	Diagnostic failure: ASCQ = Component ID field replaceable unit	39.7
3	Recovered Error	Failure Prediction Threshold Exceeded Field Replaceable Unit	4.3
4	Recovered Error	Failure Prediction Threshold Exceeded Field Replaceable Unit	16.3

Table 6: Error Log messages that appeared during four disk failures. The table shows the primary and secondary messages that appeared in the system logs during a disk failure.



Figures 7 a, b, c, and d. These graphs show the intensity of error messages over time for four of the five disk cases. The X axis shows time in hours and the Y axis shows the number of messages per hour. Note: the graphs are not to the same scale. As the figures show, in general, the intensity of error messages does go up during a disk failure. However, the shape of the curve in each case is quite different.

indicating that the disk had completely stopped working. In the third and fourth cases, Recovered Error messages appeared over several hours. In these cases, all requests were still satisfied with retries and error correction.

The table shows that for all disk failure cases, the error logs contained messages lasting several hours. The first case is the most extreme, with the messages appearing over seven days. Although this data is useful to understand how failures evolve, we note that the cases cannot be compared directly on the basis of time duration for the following reasons. First, the error messages appeared when requests to the disk had problems. Looking back over the logs, we do not know how many requests were successfully completed by the disks. Also, in our system, disks are often replaced before they completely stop working. The data in the error log only show the time between the first error message and the time that the drive is replaced. Because of these inconsistencies, we don’t directly compare the failure behavior of the drives, but rather attempt to gain some general understanding of the nature of drive failures by looking at what is common in all cases.

Next we look at the intensity of the error messages over time. Note that we use ErrorMessages, not Error Instances, in this analysis. We count all messages, even those that appeared less than 10 seconds apart. Figures 7(a), (b), (c) and (d) show the number of messages per hour for each case. As the figures show, the shape of the curves in each case is quite different. Figure 7(a) shows a slow and steady increase. Figure 7(b) shows two peaks, over a day apart. Figures 7(c) and 7(d) both show a sudden increase in errors, but Figure 7(d) has a single error message (not visible in the graph), that shows up around 12 hours before the sudden increase in errors. Although the shapes of the curves are totally different in the four cases, each shows an increase in error messages over time. We believe that this increase is due to the disk failure and not some change in workload. There is no reason to believe that all four disks experienced a workload increase in the time intervals shown in Figure 7.

Table 3 and Figure 7 suggest that disk failures 3 and 4 happened in much the same way. The error messages reported were the same, and both disks show the same escalation in error messages over time. Both failures also occurred over a fairly short time. The second case is the only one among the four that seems to be fail-stop. The error message that is reported is Disk Not Ready, indicating that the disk has completely stopped responding. In this case, the second peak in message intensity may have occurred while an operator was testing the drive.

In section 4.5, Figure 5(b) showed that some disk error messages do occur from time to time without any disk failure. The main differences between instances of disk error messages with or without failure are the intensity, duration and type of message. Although Recovered Error and Hardware failure messages and do occur from time to time, the secondary message (explaining the cause for the error) is often different in cases where disk failure is imminent. The Recovered Error and Hardware Failure messages mentioned in Section 4.3 had follow up messages “Sector Re-allocated” and “No Seek Complete”, while the same high level messages in this case were followed by “Pheripheral Write Fault” and “Failure Prediction Threshold Exceeded”. This difference indicates that the type of message reported by the drive may be useful in failure prediction. Also, the errors in section 4.3 were few in number (less than 5 per disk in all cases), and did not increase over time, while the failing disks reported hundreds of error messages, with more messages per hour as time went by.

Evaluating failure prediction techniques is beyond the scope of this paper, but the shape of the curves suggests that a simple scheme that triggers after the error intensity passes some threshold, or a scheme that attempts to capture the error arrival process [9] may work to predicting these failures. Our initial experiments with the DFT technique described in [9] suggest that it predicts both disk and SCSI component failures quite well.

5.0 Discussion

We can draw several conclusions from the data in Section 4. First, the data supports our intuition that failures are not instantaneous. The time correlation data in Section 4.5 showed that several machines showed bursts of SCSI errors, escalating over time. In Section 4.6, we saw that in three of the four cases, error messages appeared over hours or days. This data suggests that a sequence of error messages from the same device will suggest imminent failure. A single message is not enough, Section 4.3 showed that components report hardware errors from time to time without failing entirely.

Second, SCSI errors happen often in a large storage system. Section 4.3 showed that over six months, SCSI errors made up almost 50% of all errors in the system. Even though the SCSI parity errors were relatively localized, appearing in only three of the 16 machines studied, the SCSI timeout errors were not. SCSI timeout errors appeared in 10 of the 16 machines. SCSI timeouts affect system performance for two reasons. First, a timeout typically indicates that devices that wish to use the bus are not able to use it, delaying requests. Second, as the SCSI controller regains control by issuing a BUS RESET, a timeout can cause the controller to abort all active requests on the bus. When there a large number of disks on the bus and each disk has several tagged commands outstanding, a SCSI timeout can severely degrade performance. The data also suggests that failures of SCSI components are predictable. Disks already provide some warning that failure is imminent; the data in Sections 4.3 and 4.5 suggest that SCSI failures may also be predictable. Since many disks are dependent on a single SCSI bus, it would be very useful to predict the failures of SCSI buses. It may also be possible to avoid the degraded performance that occurs before a SCSI bus has an absolute failure.

Third, the data also shows that data disks are among the most reliable components in the system. Section 4.1 showed that data disks had the lowest percentage failure rate of all components that failed in one year. This data suggests that work in the literature that has focused on disk reliability do not adequately reflect real systems. Section 4.6 supports prior work showing that disk failure is predictable. Newer drives have technology to send detailed information on the errors and mechanisms to warn the operating system of imminent failure [16].

6.0 Related Work

There has been little data available on the reliability of storage system components. An earlier study [15] suggested that system error logs can be used to study and predict system failures. This work focused on filtering noise and gathering useful information from a system log. The authors introduced the “tuple concept”; they defined a tuple as a group of error records or entries that represent a specific failure symptom. A tuple contains the earliest recorded time of the error, the spanning time, an entry count, and other related information. The work described a Tuple Forming Algorithm, to group individual entries into Tuples, and a Tuple Matching Algorithm to group tuples representing the same failure symptom. The study did not attempt to characterize the failure behavior of devices, and was not specifically target at storage systems. Our log analysis used a simplified version of the tuples described in [15]; we classified error messages of the same type into tuples if they were occurred less than ten seconds apart. In future work we plan to use slightly more sophisticated tuples, for example, to take into account the time duration of a single error tuple. Follow up work characterized the distributions of various types of errors and developed techniques to predict disk failures [17]. In this study, the system was instrumented to collect very detailed information on error behavior [19]. This work, again, did not focus on storage systems with large numbers of disks.

A second study associated with the RAID effort [1] presented factory data on disk drive failure rates. This study focused on determining the distribution of disk drive lifetimes. The authors found that disk drive lifetimes can be adequately characterized by an exponential distribution. An analysis of availability of Tandem systems was presented in [20]. This work found that software errors are an increasing part of customer reported failures in the highly available systems sold by Tandem. Most recently, disk companies have collaborated on the S.M.A.R.T (Self, Monitoring, Analysis and Reporting Technology) standard [8]. SMART enabled drives monitoring a set of drive attributes that are likely to degrade over time. The drive notifies the host machine if failure is imminent.

7.0 Summary

This paper presented an analysis of hardware errors in a large storage system. We show results from six months of system logs on 16 machines, absolute failure data for the entire prototype for eighteen months, and four case studies of disk drive failures. The data showed that data drives are among the most reliable components in the system, while SCSI components generated a considerable number of errors. The data shows that no failure happens instantly, and that there are performance consequences when operating with degraded components. The data also supported the idea that it is possible to predict the failure of both disk drives and SCSI components. Our future work includes exploring the space of failure prediction algorithms.

8.0 References:

[1] Gibson, G. *Redundant Disk Arrays: Reliable Parallel Secondary Storage*. The MIT Press, Cambridge Massachusetts, 1992.

[2] Chen. P.M. Lee, E.K., Gibson, G.A, Katz, R.H. Patterson, D.A. RAID: High Performance Reliable Secondary Storage. *ACM Computing Surveys* June 1994 vol.26 {no.2):145-88

[3] Ng, S. Crosshatch disk array for improved reliability and performance. *Proceedings the 21st Annual International Symposium on Computer Architecture* April 1994 p. 255-64.

[4] Burkhard, W. Menon, J. Disk Array Storage System reliability. *Proceedings 23rd annual International Symposium on Fault Tolerant Computing*, June 1993

[5] Cheetah Disk Drive Specification, Seagate Corporation, 1997

[6] Hartman, J. Ousterhout J. The Zebra Striped Network File System, *ACM Transactions on Computer Systems*. August 1995.

[7] Cao, P. Lim, S.B. Venkataraman, S. Wilkes, J. The TickerTAIP Parallel RAID Architecture. *Proceedings 20th Annual International Symposium on Computer Architecture*, May 1993

[8] Self Monitoring. Analysis and Reporting Technology (S.M.A.R.T) Frequently Asked Questions `http://www.seagate.com:80/support/disc/faq/smart.shtml`

[9] Lin, T-T. Siewiorek, D. Error Log Analysis: Statistical Modeling and Heuristic Trend Analysis. *Proceedings of the IEEE Transactions on Reliability.* Vol 39. No 4. October 1990.

[10] Talagala, N. Asami, S. Patterson, D. Access Patterns of a Web Based Image Collection. To Appear: *Proceedings of the 1999 IEEE Symposium on Mass Storage Systems*.

[11] FreeBSD Library Functions Manual, Version 2.2

[12] Tsao, M. Trend Analysis and Fault Prediction. PhD. Dissertation, Technical Report CMU-CS 83/130, Computer Science Division, Carnegie Mellon University, 1983.

[13] Schulze, M.E Considerations in the Design of a RAID Prototype. Technical Report UCB/CSD 88/448, Computer Science Division, University of California at Berkeley, 1988

[14] The SCSI-2 Interface Specification.

[15] Worthington, B.L.; Ganger, G.R.; Patt, Y.N.; Wilkes, J. On-line extraction of SCSI disk drive parameters. *1995 Joint International Conference on Measurement and Modeling of Computer Systems*.

[16] Predictive Failure Analysis, IBM Corporation. `http://www.storage.ibm.com/storage/oem/tech/pfa.htm`

[17] *Personal Communication*. Mike Smith, FreeBSD SCSI Group.

[18] *Personal Communication*. Kenneth Merry, FreeBSD SCSI Group.

[19] T-T Lin, Design and Evaluation of an on-line predictive diagnostic system. Ph.D Thesis, Technical Report, Electrical and Computer Engineering, CMUCSD-88-1. Carnegie Mellon University. April 1988.

[20] Gray, J. A Census of Tandem System Availability Between 1985 and 1990. *Proceedings of the IEEE Transactions on Reliability.* Vol 39. No 4. October 1990.