Redundant Arrays of IDE Drives

D. A. Sanders, *Member, IEEE*, L. M. Cremaldi, *Member, IEEE*, V. Eschenburg, C. N. Lawrence, C. Riley, *Member, IEEE*, D. J. Summers, D. L. Petravick

Abstract **We report tests of redundant arrays of IDE disk drives for use in offline high energy physics data analysis. Parts costs of total systems using commodity EIDE disks are now at the $4000 per Terabyte level. Disk storage prices have now decreased to the point where they equal the cost per Terabyte of Storage Technology tape silos. The disks, however, offer far better granularity; even small institutions can afford to deploy systems. Our tests include reports on software RAID-5 systems running under Linux 2.4 using Promise Ultra 100^{TM}disk controllers. RAID-5 protects data in case of a single disk failure by providing parity bits. Tape backup is not required. Journaling file systems are used to allow rapid recovery from crashes. Our data analysis strategy is to encapsulate data and CPU processing power. Analysis for a particular part of a data set takes place on the PC where the data resides. The network is only used to put results together. We explore three methods of moving data between sites; internet transfers, hot pluggable IDE disks in FireWire cases, and DVD-R disks.**

Keywords— **RAID, EIDE, FireWire.**

I. Introduction

We report tests of redundant arrays of IDE disk drives for use in offline high energy physics data analysis [1]. Parts costs of total systems using commodity IDE disks are now at the $4000 per Terabyte level. A revolution is in the making. Disk storage prices have now decreased to the point where they equal the cost per Terabyte of 300 Terabyte Storage Technology tape silos. The disks, however, offer far better granularity; even small institutions can afford to deploy systems. The faster access of disk versus tape is an added bonus. Our tests include reports on software RAID-5 systems running under Linux 2.4 using Promise Ultra 100^{TM}disk controllers. RAID-5 protects data in case of a single disk failure by providing parity bits. Tape backup is not required. Journaling file systems are used to allow rapid recovery from crashes. We also report on using FireWire (IEEE 1394) to PCI interfaces. With three PCI cards and sixty-three 160 Gigabyte disks per card one could attach 30 Terabytes to a single PC. FireWire is also hot pluggable.

Our data analysis strategy is to encapsulate data and CPU processing power. Data is stored on many PCs. Analysis for a particular part of a data set takes place on the PC where the data resides. The network is only used to put results together. Alternate analysis schemes may be used with somewhat lower efficiency. Commodity 5-port 10/100 ethernet switches combined with a single high end, fast backplane switch (we use a Lucent Cajun®P550 [2]) would allow one to connect a thousand PCs, each with perhaps a Terabyte of disk space.

We explore three methods of moving data between sites; internet transfers, hot pluggable IDE disks in FireWire cases, and DVD-R disks. Writable 4.7 GB DVD-R disks are now available for $6. They can be read by $60 DVD-ROM drives and written by the $500 Pioneer DVR-A03 drive [3].

RAID [4] stands for Redundant Array of Inexpensive Disks. Many industry offerings meet all of the qualifications except the inexpensive part, severely limiting the size of an array for a given budget. This may change. The different RAID levels can be defined as follow:

- RAID-0: "Striped." Disks are combined into one physical device where reads and writes of data are done in parallel. Access speed is fast but there is no redundancy.
- RAID-1: "Mirrored." Fully redundant but the size is limited to the smallest disk.
- RAID-4: "Parity." For N disks, 1 disk is used as a parity bit and the remaining $N-1$ disks are combined. Protects against a single disk failure but access speed is slow since you have to update the parity disk for each write.
- RAID-5: "Striped-Parity." As with RAID-4, the effective size is that of $N-1$ disks. However, since the parity information is also distributed evenly among the N drives the bottleneck of having to update the parity disk for each write is avoided. Protects against a single disk failure and the access speed is fast.

RAID on EIDE (Enhanced Integrated Drive Electronics) disks under Linux software which both stripes data across disks for speed and provides parity bits for data recovery (RAID-5) is now available [5]. With redundant disk arrays, tape backup is not needed to recover from the failure of one disk in a set. This removes a major obstacle to building large arrays of EIDE disks.

II. Large Disks

In today's marketplace, the cost per Terabyte of disks with EIDE interfaces is about a third that of disks with SCSI (Small Computer System Interface). The EIDE interface is limited to 2 drives on each bus and SCSI is limited to 7 (14 with wide SCSI). The only major drawback of EIDE disks is the limit in the length of cable connecting the drives to the drive controller. This limit is nominally 18 inches; however, we have successfully used 24 inch long cables [6]. Therefore, one is limited to 10 disks per box for an array (or perhaps 20 with a "double tower"). To get

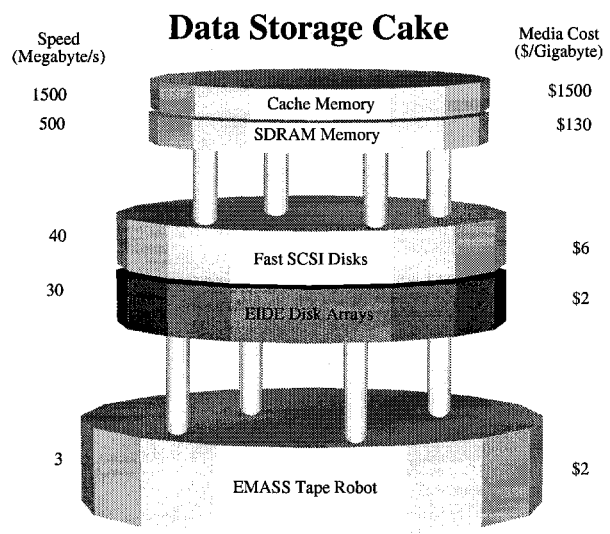
Manuscript submitted to IEEE Transactions On Nuclear Science, November 25, 2001. This work was supported in part by the U.S. Department of Energy under Grant Nos. DE-FG05-91ER40622 and DE-AC02-76CH03000.

D. A. Sanders, L. M. Cremaldi, V. Eschenburg, C. N. Lawrence, C. Riley, and D. J. Summers are with the University of Mississippi, Department of Physics and Astronomy, University, MS 38677 USA (telephone: 662-915-5438, e-mail: sanders@phy.olemiss.edu.)

D. L. Petravick is with the Fermi National Accelerator Laboratory, CD-Integrated Systems Development, MS 120, Batavia, IL 60510-0500 USA (telephone: 630-840-3935, e-mail: petravick@fnal.gov)

a large RAID array one needs to use large capacity disk drives. There have been some problems with using large disks, primarily the maximum addressable size. Historically the limits have been 2^{22}, 2^{24}, 2^{26}, and now 2^{28} 512 byte blocks. This has limited disk sizes to 2.1 GB, 8.6 GB, 34.4 GB, and now 137.4 GB. The computer hardware and software would not address disks larger than these limits. The hardware solutions to this was either to upgrade the motherboard BIOS, limit the disk capacity with a jumper, or use a PCI disk controller card that did not have the limits in its BIOS. We have addressed these problems in an earlier paper [7], and because we wanted to put more drives in an array than could be supported by the motherboard we opted to use PCI disk controller cards. We tested both Promise Technologies [8] ULTRA 66™and ULTRA 100™disk controller cards, which each support four drives.

Using arrays of disk drives, as shown in Table I, the cost per Terabyte is similar to that of cost of Storage Technology tape silos. However, RAID-5 arrays offer far better granularity since they are scalable down to a Terabyte. Thus, even small institutions can afford to deploy systems. Therefore, as seen in the Figure, "you can have your cake and eat it too".



III. Software RAID Arrays

We considered both hardware and software RAID. Hardware RAID should have faster access times since the RAID is handled by a separate CPU; however, the presence of this CPU increases the cost, from about $50 to $600 for the 3ware Escalade 7850 card [9]. The Escalde 7850 adds 2 Megabytes of memory to the base Escalade 7810 for enhanced RAID 5 write performance. In the future, 3ware intends [10] to provide a software upgrade that allows the Escalade 7850 to exploit disks larger than 137 Gigabytes. The Escalade 7850 controls eight EIDE disk drives. We decided to first concentrate on software RAID and we have extensively tested RAID-5 arrays using software RAID [5], [11].

A. Hardware

We have examined both Maxtor DiamondMax™[12] and IBM DeskStar™[13] hard disks. For RAID-5 the disk partitions must be all of the same size; therefore, the only trouble we had was when Maxtor changed the capacity for the 80 GB disk from 81.9 GB to 80 GB. One GB is defined as 1000MB and not 1024MB. The drives we consider for use with a RAID-5 array are compared in Table I. In general, the internal I/O speed of a disk is proportional to the product of its rotational speed and platter capacity. When assembling an array we had to be concerned with

TABLE I
Comparison of Large EIDE Disks for a RAID-5 Array

Disk Model	GB	RPM	Cost per GB	GB per platter	Spin-Up Current at 12V
Maxtor D740X	80	5400	$2.11	20	2.00A
Maxtor D536X	100	5400	$2.27	33	0.64A
Maxtor D540X	160	5400	$1.85	40	1.80A
IBM 75GXP [13]	75	7200	$3.19	15	2.00A
IBM 120GXP [14]	120	7200	$2.91	40	2.00A

a few other things. We had to worry about the "spin-up" current draw on the 12V part of the power supply. With 8 disks in the array (plus the system disk) we would have exceeded the capacity of the power supply that came with our tower case, so we decided to add a second off-the-shelf power supply rather than buying a more expensive single supply. We have measured the power consumption for the whole disk array box described below. It uses 276 watts at startup and 156 watts during normal sustained running. We used the hardware shown in Table II for our array test. Many of the components we chose are generic; many components from other manufacturers also work.

To install the second power supply we had to modify our tower case with a jigsaw and a hand drill. We also had to use a jumper to ground the green wire in the 20-pin block ATXPWR connector to fake the power on switch. When installing the 2 disk controller cards care had to be taken that they did not share interrupts with other highly utilized hardware such as the video card and the ethernet card. We also tried to make sure that they did not share interrupts with each other. When we tried to use a disk as a "Slave" on a motherboard EIDE bus, we found that it would not run at the full speed of the bus and slowed down the access speed of the entire RAID-5 array. This problem was not in evidence when using the disk controller cards. Therefore, we decided that rather than take a factor of 10 hit in the access speed we would rather use 8 instead of 9 hard disks.

B. Software

For the actual tests we used Linux kernel 2.4.5 with the RedHat 7 distribution (we had to upgrade the kernel to this level). The latest stable kernel version is 2.4.14. We

TABLE II
700 GB RAID 5 CONFIGURATION

System Component	Unit Price
100GB Maxtor system disk [12]	$227
8 – 100GB Maxtor RAID5 disks	$227
2 – Promise ATA/100 PCI cards [8]	$27
4 – StarTech 24" ATA/100 cables [6]	$3
AMD Athlon 1.4 GHz/266 CPU [15]	$120
Asus A7A266 motherboard, audio [16]	$132
2 – 256MB DDR PC2100 DIMMs	$35
In-Win Q500P Full Tower Case [17]	$77
Sparkle 15A @ 12V power supply [18]	$34
2 – Antec 80mm ball bearing case fans	$8
110 Alert temperature alarm [19]	$15
Pine 8MB AGP video card [20]	$20
SMC EZ card 10/100 ethernet [21]	$12
Toshiba 16x DVD, 48x CDROM	$54
Sony 1.44 MB floppy drive	$12
KeyTronic 104 key PS/2 keyboard	$7
DEXXA 3 button PS/2 mouse	$4
Total	$2682

needed the 2.4.x kernel to allow "Journaling" file systems. Journaling file systems are used to allow rapid recovery from crashes. We tested 2 different Journaling file systems; ReiserFS [22] and ext3 [23]. We opted on using ext3 for two reasons 1) At the time there were stability problems with ReiserFS and NFS (this has since been resolved with kernel 2.4.7) and 2) it was an extension of the standard ext2fs (it was originally developed for the 2.2 kernel) and, if synced properly could be mounted as ext2. Since we planned to use NFS software to connect these disks arrays to other computers, including those that cannot run Linux 2.4.x. We have successfully used NFS to mount this disk array on the following types of computers: a DECstation 5000/150 running Ultrix 4.3A, a Sun UltraSparc 10 running Solaris 7, a Macintosh G3 running OSX, and various Linux boxes with both the 2.2 and 2.4 kernels. We are currently using two of these RAID-5 boxes to run analysis software with the BABAR KANGA code and the CMS CMSIM/ORCA code.

We have performed a few simple speed tests. The first was "hdparm -tT /dev/xxx". On a single drive we saw read/write speeds of about 30 MB/s. On the whole array we saw a drop to 28 MB/s. When we tried writing a text file using a simple FORTRAN program (We wrote "All work and no play make Jack a dull boy" 10^8 times), the speed was 22.34 MB/s. While mounted via NFS over 100 Mb/s ethernet the speed was 2.12 MB/s.

We also tested what actually happens when a disk fails by turning the power off to one disk in our RAID-5 array. One could continue to read and write files, but in a "degraded" mode without the parity safety net. When a blank disk was added to replace the failed disk, again one could continue to read and write files in a "degraded" mode while the system rebuilt the missing disk as a background job. For more details, see reference [11].

The performance of Linux IDE software drivers are improving. The latest standards [24] include support for command overlap, READ/WRITE DMA QUEUED commands, scatter/gather data transfers without intervention of the Central Processor, and elevator seeks. Command overlap is a protocol that allows devices that require extended command time to perform a bus release so that commands may be executed by the other device on the bus. Command queuing allows the host to issue concurrent commands to the same device. Elevator seeks minimize disk head movement by optimizing the order of I/O commands.

We did encounter a few problems. We had to worry about sharing of IRQs. Because we wanted to maximize performance we wanted to have the EIDE disk controller cards have unique IRQs that were not shared by any other hardware device. We had to modify "MAKEDEV" to allow for more than eight IDE devices, that is to allow for disks beyond "/dev/hdg". For version 2.x one would have to actually modify the script; however, for version 3.x we just had to modify the file "/etc/makedev.d/ide". Another problem was the 2 GB file size limit. By their very nature 32 bit processors can not normally address files larger than 2 GB (2^{31}). There are patches to the Linux 2.4 kernel and glibc but there are still some problems with NFS and not all applications use these patches.

We have found that the current underlying file systems (ext2, ext3, reiserfs) do not have a 2 GB file size limit. The limit for ext2/ext3 is in the petabytes. The 2.4 kernel series supports large files (64-bit offsets). Current versions of GNU libc support large files. However, by default the 32-bit offset interface is used. To use 64-bit offsets, C/C++ code must be recompiled with the following as the first line:

```
#define _FILE_OFFSET_BITS 64
```

or the code must use the *64 functions (i.e. open becomes open64, etc.) if they exist. This functionality is not included in GNU Fortran (g77); however, it should be possible to write a simple wrapper C program to replace the OPEN statement (perhaps called open64). We have succeeded in writing files larger than 2 GB using a simple C program with "#define _FILE_ OFFSET_ BITS 64" as the first line. This works over NFS version 3 but not version 2.

While RAID-5 is recoverable for a hardware failure, there is no protection against accidental deletion of files. To address this problem we suggest a simple script to replace the "rm" command. Rather than deleting files it would move them to a "/raid/Trash" or better yet a "/raid/.Trash" directory on the RAID-5 disk array (similar to the "Trash can" in the Macintosh OS). The system administrator could later purge them as space is needed using an algo-

rithm based on criteria such as file size, file age, and user quota.

IV. FireWire

We got FireWire (IEEE 1394/i.LINK) working a Linux box by following the following steps:

1. Bought an inexpensive PCI FireWire controller, for a cost of $25. It was an OHCI-1394 card with a VIA controller. OHCI chipsets apparently are the best-supported under Linux and are the most common; TI's PCILynx chipset also works.
2. The kernel used is Linux 2.4.12 as released by Linus and Alan Cox's -ac3 patch. Alan's patches can be downloaded at http://www.bz2.us.kernel.org/pub/linux/kernel/people/alan/linux-2.4/. The -ac series is basically what Red Hat and other distributions base their kernels on, and includes drivers not in stock 2.4.12.
3. Enabled FireWire support in "make config" by answering "M" to these prompts in "make config":

```
IEEE 1394 (FireWire) support (EXPERIMENTAL)
  OHCI-1394 support
  SBP-2 support (Harddisks etc.)
```

(The RAWIO driver is not necessary for storage devices. In addition, you will need the SCSI disk driver enabled in the kernel, even if you don't have a real SCSI interface on the machine. This is because FireWire is treated as a SCSI channel.)

4. After rebooting with the new kernel, some recent distributions should detect the FireWire card and install the correct drivers. If not, the following modules need to be manually loaded, in this order:

```
ohci1394
sbp2
```

The sbp2 driver is somewhat finicky; it helps to have a few seconds' delay between the two modprobes. The command "cat /proc/scsi/scsi" should list the attached storage devices (disks, CD-ROMs, etc.):

```
Attached devices:
Host: scsi1 Channel: 00 Id: 00 Lun: 00
  Vendor: Maxtor  Model: 1394 storage   Rev: 60
  Type:   Direct-Access  ANSI SCSI revision: 02
```

Some of the output may not make sense if an IDE-FireWire (1394) bridge is in use; we noticed the non-Maxtor drive had strange output.

5. At the moment, the devices are added in more-or-less random order. The only way to guarantee ordering is to manually hot-plug them. We don't know if this is a software limitation or an artifact of the plug&play nature of FireWire (there's no permanent ID setting like IDE or SCSI have). Presumably if one writes a volume header label (e.g. with tune2fs -L) to each disk you could get around this problem.
6. Hot plugging seems to work fine. However, DO NOT UNPLUG A DEVICE WITHOUT UNMOUNTING IT FIRST. Once unmounted, disconnect the device physically and then run "rescan-scsi-bus.sh -r". To add new devices, plug them in and run "rescan-scsi-bus.sh". The script can be downloaded at
http://www.garloff.de/kurt/linux/rescan-scsi-bus.sh

We configured two FireWire disks as a RAID-5 array successfully. One of the disks used the new Oxford 911 FireWire to EIDE interface chip [25], [26].

V. High Energy Physics Strategy

Our data analysis strategy is to encapsulate data and CPU processing power. Data is stored on many PCs. Analysis for a particular part of a data set takes place on the PC where the data resides. The network is only used to put results together. Alternate analysis schemes may be used with somewhat lower efficiency. NFS software is used to connect these disks arrays to computers which cannot run Linux 2.4.

What would be required to build a petabyte system? Start with eight 160 GB Maxtor disks in a box. The Promise Ultra133 card allows one to exceed the 137 GB limit [27]. Each box provides $7 \times 160\,\text{GB} = 1120\,\text{GB}$ of usable RAID5 disk space in addition to a CPU for computations. A petabyte is reached with 900 boxes. Use 300 commodity 5-port 10/100 ethernet switches ($60 each) to connect the 900 boxes to a 300-port, high end, fast backplane ethernet switch [2]. The boxes consume 156 watts each while running for a total of 141 kilowatts. Two dozen 24 000 BTU window air conditioners would suffice to remove the heat load. The volume occupied is about a hundred cubic meters. If each disk were housed in its own hot pluggable FireWire case [26], replacing failed disks might be easier.

For small amounts of data and to update analysis software one can use internet file transfers, preferably via "rsync". The program "rsync" remotely copies files and uses a remote-update protocol to greatly speedup file transfers when the destination file already exists. This remote-update protocol allows "rsync" to transfer just the differences between two sets of files across the network link, using an efficient checksum-search algorithm. Some of the additional features of "rsync" are: support for copying links, devices, owners, groups and permissions; can use any transparent remote shell, including "rsh" or "ssh"; and does not require root privileges. The only problem is the available bandwidth. Internet2 may solve this problem but given the prevalence of Napster-like programs competing with data transfers, this is not a certainty. The other method would be to use some form of removable, and universally readable media. Two new methods are hot pluggable IDE disks in FireWire cases [26], and DVD-R disks. Since FireWire works on Linux, Windows 98SE, and Macintosh OS9 and OSX, one can use hot pluggable EIDE disks in FireWire cases as a simple method of transferring reasonable amounts of data or even full sets of analysis software.

Writable 4.7 GB DVD-R disks are now available for $6. They can be read by $60 DVD-ROM drives and written by the $500 Pioneer DVR-A03 drive [3]. Linux is capable of writing DVD-Rs. However, the software to do so is not available under a free license. It is an enhanced version

of "cdrecord", the free program that writes CDs, CD-Rs, and CD-RWs. A demo version that will write up to 1 GB is available from the author's FTP site [28]. According to folks on the linux1394 mailing list, it works over FireWire. We have succeeded in using the Pioneer DVR-A03 drive to write a CD-RW (from an iso image produced by "mkisofs") over FireWire with Linux using the program "cdrecord". The reliability of DVD-R media needs to be explored.

VI. Conclusion

We report on tests of redundant arrays of IDE disk drives for use in offline high energy physics data analysis. Parts costs of total systems using commodity IDE disks are now at the $4000 per Terabyte level. A revolution is in the making. Disk storage prices have now decreased to the point where they equal the cost per Terabyte of Storage Technology tape silos. The disks, however, offer far better granularity; even small institutions can afford to deploy systems. The faster access of disk versus tape is an added bonus. Our tests included reports on software RAID-5 systems running under Linux 2.4 using Promise Ultra 100^{TM}disk controllers. RAID-5 protects data in case of a single disk failure by providing parity bits. Tape backup is not required. Journaling file systems are used to allow rapid recovery from crashes. We also reported on using FireWire to PCI interfaces. With three PCI cards and 63 160 Gigabyte disks per card one could attach, in principle, 30 Terabytes to a single PC. FireWire is also hot pluggable. Our data analysis strategy is to encapsulate data and CPU processing power. Data is stored on many PCs. Analysis for a particular part of a data set takes place on the PC where the data resides. The network is only used to put results together. Alternate analysis schemes may be used with somewhat lower efficiency. Commodity 5-port 10/100 ethernet switches combined with a single high end, fast backplane switch would allow one to connect a thousand PCs, each with a Terabyte of disk space. We explored three methods of moving data between sites; internet transfers, hot pluggable EIDE disks in FireWire cases, and DVD-R disks.

References

[1] For an example of the recording and reconstruction of a 50 Terabyte high energy physics data set see:
S. Amato et al., "The e791 parallel architecture data acquisition system," *Nucl. Instrum. Meth.*, vol. A324, pp. 535–542, 1993, arXiv:hep-ex/0001003; and
Steve Bracker, Krishnaswamy Gounder, Kevin Hendrix, and Don Summers, "A simple multiprocessor management system for event parallel computing," *IEEE Trans. Nucl. Sci.*, vol. 43, pp. 2457–2464, 1996, arXiv:hep-ex/9511009.

[2] The Lucent Cajun P550 Switch has 23 Gigabits per second of backplane capacity. Up to six cards may be installed in this switch. One option is a card with 48 full duplex 10/100 Base-TX ports.
Available: http://lucent.netlabs.net/ins/products/p550.html

[3] Available: http://www.pioneer-eur.com/ products/multimed/optical/dvrA03.htm

[4] D. Patterson, G. Gibson, and R. Katz, "A case for redundant arrays of inexpensive disks," 1988, In Proc. of the 1988 ACM SIGMOD Conf. on Management of Data, June 1988, pp. 109-116.

[5] Miguel de Icaza, Ingo Molnar, and Gadi Oxman, "The linux raid-1,4,5 code," 1997, 3rd Annual Linux Expo '97, April 1997.

[6] Available: http://www.startech.com/
The ATA/100 standard uses an 80 wire cable to transmit up to 100 Megabytes per second.

[7] David Sanders, Chris Riley, Lucien Cremaldi, Don Summers, and Don Petravick, "Working with arrays of inexpensive eide disk drives," 1999, Presented at International Conference on Computing in High- Energy Physics (CHEP 98), Chicago, Illinois, 31 Aug - 4 Sep 1998, arXiv:hep-ex/9912067.

[8] Available: http://www.promise.com/ Products/Default.htm and http://www.promise.com/ Products/Ultra_Tx2/U100_TX2_Data_Sheet.pdf.
Each ATA/PCI Promise card controls four disks.

[9] Available: http://www.3ware.com/news/7450_7850.pdf and http://www.3ware.com/ products/EscaladeReturnLetter.asp

[10] Kent Abendroth, kent.abendroth@3ware.com, Escalade Global Sales Manager, personal communication.

[11] Jakob Østergaard, (Jan. 2000), "The software-RAID HOWTO,", Available: http://www.linuxdoc.org/ HOWTO/Software-RAID-HOWTO.html.

[12] Available: http://www.maxtor.com/

[13] Available: http://www.storage.ibm.com/ hdd/prod/ds75gxp40gv.htm

[14] Available: http://www.storage.ibm.com/ hdd/desk/ds120gxp.pdf

[15] Available: http://www.amd.com/
We bought our AMD CPU boxed with a fan.

[16] Available: http://www.asus.com/ Products/motherboard/socketa/a7a266/

[17] Available: http://www.in-win.com/framecode/index.html
A 300 watt supply comes with the Q500P case.

[18] Available: http://www.sparklepower.com/ pdf/FSP300-60BTV.pdf
The Sparkle FSP300-60BTV is used as a second 300 watt supply. At 12 volts it gives 15 amps.

[19] Available: http://www.pcpowercooling.com/ products/alarmandaccesories/alert/index.htm

[20] Available: http://www.pinegroup.com/ and http://www.nvidia.com/
The Pine 8MB AGP NVIDIA VANTA LT video card is used to run a monitor for diagnostics.

[21] Available: http://www.smc.com/ drivers_downloads/library/SMC1211.pdf and http://www.smc.com/drivers_downloads/library/rtl8139.c
The SMC 1211TX EZ PCI Card uses an rtl8139 10/100 ethernet software driver.

[22] Available: http://www.reiserfs.org/

[23] Available: http://www.uow.edu.au/~andrewm/linux/ext3/

[24] Available: http://www.t13.org/

[25] Available: http://www.oxsemi.com/press/dec00/index.html ; http://www.oxsemi.com/products/IEEE1394/oxfw911ov.pdf ; http://www.oxsemi.com/products/IEEE1394/fw911ds.pdf ; and http://www.oxsemi.com/products/an/oxan1.pdf

[26] Other World Computing, 5.25" Firewire Disk Case with Oxford 911 chip, $90
Available: http://eshop.macsales.com/ Catalog_Item.cfm?ID=1445&Item=casekitfw

[27] Promise Technology's Ultra133 TX2 PCI controller card uses a wider 48-bit data address versus the older 28-bit address, which is limited to 2^{28} 512 byte blocks or 137 Gigabytes. The card controls four disks and has a $59 list price.

[28] Available: ftp.fokus.gmd.de/pub/unix/cdrecord/ProDVD/