Tuning Server Performance and Adding Internal Hard Drives

Apple Computer, Inc.

This manual and the software described in it are copyrighted, with all rights reserved. Under the copyright laws, this manual or the software may not be copied, in whole or part, without written consent of Apple, except in the normal use of the software or to make a backup copy of the software. The same proprietary and copyright notices must be affixed to any permitted copies as were affixed to the original. This exception does not allow copies to be made for others, whether or not sold, but all of the material purchased (with all backup copies) may be sold, given, or loaned to another person. Under the law, copying includes translating into another language or format.

You may use the software on any computer owned by you, but extra copies cannot be made for this purpose.

The Apple logo is a registered trademark of Apple Computer, Inc. Use of the "keyboard" Apple logo (Option-Shift-k) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

Every effort has been made to ensure that the information in this manual is accurate. Apple is not responsible for printing or clerical errors.

© Apple Computer, Inc., 1994
20525 Mariani Avenue
Cupertino, CA 95014-6299
(408) 996-1010

Apple, the Apple logo, AppleShare, A/UX, HyperCard, LaserWriter, LocalTalk, Macintosh, and Macintosh Quadra are trademarks of Apple Computer, Inc., registered in the United States and other countries.

Finder and MacTest are trademarks of Apple Computer, Inc.

Adobe, Adobe Illustrator, and PostScript are trademarks of Adobe Systems Incorporated, which may be registered in certain jurisdictions.

Electrocomp 2000 is a trademark of Image Graphics Inc.

Helvetica, Linotronic, and Times are registered trademarks of Linotype Company.

Microsoft is a registered trademark of Microsoft Corporation.

NFS is a trademark of Sun Microsystems, Inc.

QuarkXPress is a registered trademark of Quark, Inc.

Retrospect is a registered trademark of Dantz Development Corporation.

UNIX is a registered trademark of UNIX System Laboratories, Inc.

X Window System is a trademark of the Massachusetts Institute of Technology.

Simultaneously published in the United States and Canada.

Mention of third-party products is for informational purposes only and constitutes neither an endorsement nor a recommendation. Apple assumes no responsibility with regard to the performance of these products.

Contents

Preface: About This Document / *v*

1 Measuring Server Performance / 1

Two methods to measure performance / 2

Method used to evaluate the Apple Workgroup Server 95 / 2

Increased performance you can expect from AppleShare Pro / 3

Server efficiency and the number of clients / 4

Efficiency of enumeration (file listing) operations / 5

2 Allocating RAM for Best Performance / 7

Basic process for allocating RAM in the server / 8

When to add RAM / 8

Redistributing RAM allocations / 9

The A/UX buffer cache / 9

The AppleShare Pro caches / 10

The file caches / 11

The folder cache and icon cache / 11

Relationship between A/UX buffer cache and AppleShare Pro caches / 13

Macintosh virtual memory / 13

Running UNIX-based services / 14

A performance pitfall: Paging through virtual memory / 15

3 Optimizing Performance When Running AppleShare Pro and NFS / 17

NFS and NIS software preinstalled / 18

Operating as both an AppleShare Pro and an NFS server / 19

Running AppleShare Pro and operating the server as an NFS client / 19

Operating as an NFS server / 20

4 Preparing Your Server for Another Service / 21

Running the `config_server` script / 22

Changes made by the script / 22

Appendix A Adding Internal Drives / 23

Before you start / 24

Installation procedure / 24

Index / 35

Preface About This Document

This document discusses the practice of measuring server performance, issues you should consider when tuning components of your server, and procedures for tuning the server to best match the needs of your site.

This document augments the suite of manuals delivered with the Apple Workgroup Server 95. Use this document in conjunction with those manuals, when you want to make any of the following changes to your server:

- Adjust the RAM allocations in the server.
- Run both NFS and AppleShare Pro on the server.
- Prepare the server to switch from file and print services to database services, or from database services to file and print services.
- Add internal hard drives.

Internal hard drives should be installed only by authorized Apple service providers in order to maintain the warranty on your server.

1 Measuring Server Performance

Measuring the performance of a server is a complicated task because no standard procedure exists for deriving a benchmark for file and print service. (A benchmark is a quantifiable measure of the performance of a computer or computer-related product, used for comparison with the performance of similar products.) Opinions abound as to what constitutes good server performance. This chapter describes the two methods considered by Apple Computer to arrive at a benchmark for the Apple Workgroup Server 95, the method chosen, and the reasons for that choice.

Two methods to measure performance

The performance of a server can be measured in two ways. You can measure the user response time of the server, or you can measure the throughput of the server. Both methods result in a benchmark for comparison of speed with other servers, and both have their advantages and disadvantages.

- User response time

  Measuring user response time is an appealing method because it measures what the user actually sees. For example, this method measures the time required to copy a 1 MB file from the server to a client computer. However, the result will be skewed by the behavior of the particular application used in the task, such as how HyperCard handles files, or how the Finder buffers data before copying it. Thus, this method may produce inaccurate results.

- Throughput capacity

  Measuring the throughput capacity of the server does not imitate the actual experience of the user, but it does give an accurate measure of the server's capability to read data from and write data to the disk.

Benchmarks are often cited in publications reporting on computer products. Various computer publications perform tests and arrive at benchmarks in different ways. Macintosh-oriented publications primarily measure any computer product by its user response time, and PC-oriented publications primarily measure server performance by the throughput capacity method.

Method used to evaluate the Apple Workgroup Server 95

Apple Computer used the throughput capacity method to measure the performance of the Apple Workgroup Server 95. The figures in the following table are the results of these tests. If you want to replicate the tests on your server, matching Apple's test criteria, you need to use the throughput capacity method with 10 clients actively using AppleShare.

Increased performance you can expect from AppleShare Pro

The following table shows the results of tests with AppleShare running on a Macintosh Quadra 950, and with AppleShare Pro running on the Apple Workgroup Server 95. As an example of the tests, the read operation involved opening a set of large files and sequentially reading from each one. The numbers in the table reflect the aggregate throughput of the server, which is the sum of all the server's processing for all of its clients.

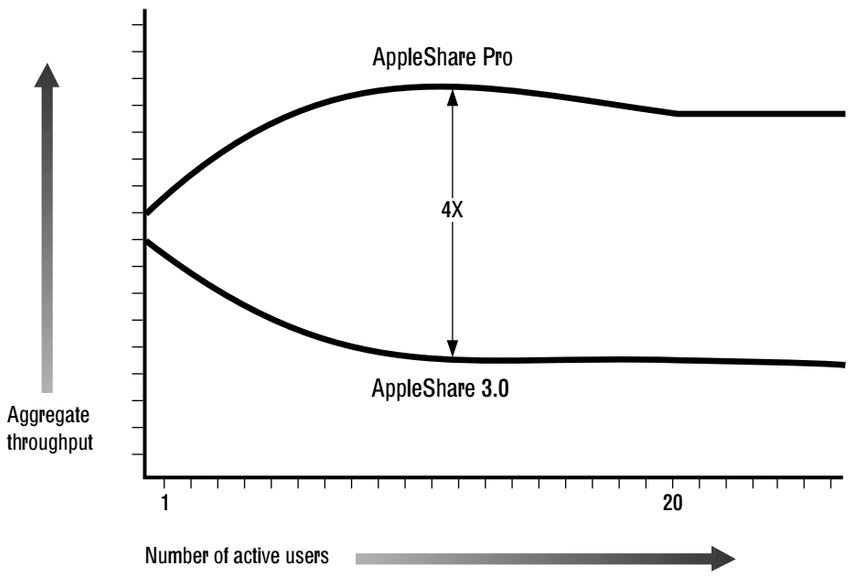
Table 1-1 Comparison of performance of AppleShare 3.0 and AppleShare Pro (with 10 active clients)

	AppleShare 3.0.1 on Macintosh Quadra 950	AppleShare Pro 1.0 on AWS 95	AppleShare Pro 1.1 on AWS 95
Sequential read operations	193 Kbits/sec.	851 Kbits/sec.	951 Kbits/sec.
Sequential write operations	160 Kbits/sec.	348 Kbits/sec.	594 Kbits/sec.
Enumeration (file listing) operations	98 items/sec.	132 items/sec.	295 items/sec.

The aggregate throughput of read and write operations performed by AppleShare Pro is four times faster than that of corresponding operations performed by AppleShare on a Macintosh Quadra 950. The improvement in aggregate throughput in AppleShare Pro as compared to AppleShare is largely due to the multitasking of the server; while the server is waiting for a request from one client, it processes a request from another client. (Tests of directory enumeration operations are discussed later in this chapter.)

Server efficiency and the number of clients

With AppleShare Pro, a single client on the server does not utilize the server to its highest capacity. As you add clients to AppleShare Pro, performance improves. The opposite is true with AppleShare 3.0; as you add clients, performance degrades. The following illustration shows this contrast.



With AppleShare Pro, optimum performance is achieved with 10 active clients, and adequate performance is maintained with 50 active clients. An active client is a user who is reading data from or writing data to the server. Typically, only 25% of clients are active at any one time. Thus, with AppleShare Pro the server maintains adequate performance with as many as 200 clients.

Once the number of active clients exceeds 50, the performance of the server drops below a level determined as adequate by Apple's user studies. The Apple Workgroup Server 95 reaches its maximum speed of 950 KB/second for read operations with 10 active clients. If this aggregate throughput is divided among 10 clients, the performance each client experiences is 95 KB/second. Apple's user studies have shown that the server provides adequate performance at speeds greater than 15 KB/second; with 50 active clients, the server runs at 850 KB/second: 850/50 = 17. Thus, Apple decided on a suggested limit of 50 active clients for AppleShare Pro.

In contrast, AppleShare 3.0 reaches the limit of useful performance with approximately 10 to 12 active clients. It has an aggregate throughput of 210 KB/second; if the number of active clients exceeds 12, its performance reaches an unacceptable level.

Efficiency of enumeration (file listing) operations

When you use the Finder to display a list of filenames and information about the files, such as the file size and the date and time of creation, the operation that the Finder performs is called a *directory enumeration operation*. Tests performed by Apple show that AppleShare Pro is approximately three times as fast as AppleShare 3.0 in performing this operation, as opposed to four times faster in the read and write operations.

2 Allocating RAM for Best Performance

Improving the performance of your server can require adding physical memory (also called *RAM*) or adjusting the existing RAM to your server's best advantage. This chapter describes the components of the server that compete for RAM and the circumstances under which adjusting these components is recommended.

Basic process for allocating RAM in the server

To optimize the performance of your server, you need to make intelligent decisions regarding where to allocate available RAM. Here is the basic process to use when making your decisions.

1. Allocate 7 MB of RAM to the A/UX kernel and the A/UX core processes (processes started by default).
2. Allocate RAM for the Macintosh applications and for UNIX-based services such as NFS, if you run them.
3. Allocate the remaining RAM to the A/UX buffer cache. Adjust this formula according to the specific usage patterns of your site, as described in the following sections.

When to add RAM

Your system performance will improve as a result of the addition of RAM only if your problem is due to limited allocations of RAM to one or more of the following system components (each of which is discussed in detail later in this chapter):

- the A/UX buffer cache
- the Macintosh virtual memory
- the various caches used by AppleShare Pro

Surprisingly, adding RAM will not always increase a system's performance. Although the additional memory strengthens the I/O capacity of the system, server performance may be limited by other factors. For example, adding RAM will not help if poor server performance is due in part to network cables and connectors that lack the speed of Ethernet cables and connectors. If poor server performance is due to an extremely busy network, increasing the amount of RAM in the server will not enhance performance.

Redistributing RAM allocations

The RAM in your server is allocated among the following components:

- the A/UX kernel and A/UX core processes (daemons)
- the A/UX buffer cache
- the Macintosh virtual memory for the Macintosh Finder environment, in which the following applications run:
  —AppleShare Pro
  —other Macintosh applications such as the Finder and Retrospect
- UNIX applications; for example, a database application that runs under the UNIX operating system

The following sections describe the circumstances under which you may benefit from adjusting the RAM for these components, and provide guidelines for making the adjustments.

The A/UX buffer cache

A full description of the A/UX buffer cache is beyond the scope of this document. Briefly, it is a portion of RAM set aside as a cache for data that has been recently requested. The assumption is that data recently requested will soon be requested again. The kernel could read data to and write data from the disk for all file-system accesses, but performance would be poor because of slow disk-transfer rate. The buffer cache minimizes the frequency of disk access.

The number of useful buffers in the cache is constrained by the amount of memory required to execute the core A/UX processes. If too much memory is used for buffers, the system may slow down because of excessive swapping of processes in virtual memory, or paging, which is explained in a later section in this chapter. By using the buffer cache, A/UX can respond to approximately 80% to 85% of read and write requests without accessing the disk.

Guidelines for resizing the A/UX buffer cache

Follow these rules when adjusting the A/UX buffer cache.

File server: If the server is used primarily to serve files, rather than to run a database application, make the A/UX buffer cache as large as you can with the available RAM. This is the way your server was configured when shipped from the factory. For example, as shipped from the factory, a server with 32 MB of RAM allocates approximately 7 MB of RAM to the A/UX kernel and core processes, and 16 MB of RAM to Macintosh virtual memory. The remaining 9 MB of RAM is allocated to the A/UX buffer cache.

Database: If the server is used for a database application, the optimal size of the A/UX buffer cache depends on the type of database application. Most Macintosh-based applications, which store data in a Macintosh file system, use the A/UX buffer cache to store data, and the size of the buffer cache as shipped from the factory is the optimal size. Many UNIX-based applications, which store data in a UNIX file system, use the A/UX buffer cache to store data, and again, the size of the buffer cache as shipped from the factory is the optimal size. However, some UNIX-based applications allow you to bypass the UNIX file system for storing data. If your database application has this capability, then reduce the size of your A/UX buffer cache. Set the cache to an amount that is 10% of the RAM in your system, and you will be providing enough memory for the A/UX core processes.

The AppleShare Pro caches

AppleShare Pro allows you to set aside RAM for temporarily caching frequently used files, folders, and icons. You adjust the settings for these caches in the File Server Cache Preferences window. A brief discussion of these caches and the guidelines for resizing them are provided in this section.

The file caches

As does A/UX with its buffer cache, AppleShare Pro sets aside a portion of RAM for two file caches to improve server performance. One file cache stores data being read from the server, and the other cache stores data being written to the server. These caches store data in read-ahead and write-behind buffers. For both buffers, AppleShare Pro anticipates that a request to the server will soon be followed by a request for the next contiguous block of data from the file. The data requested and the data anticipated to be requested are stored in these buffers. When the buffers are full, AppleShare Pro takes the buffered data and transfers it to the disk. The end result is faster throughput, because the system issues a single, large read or write request to the disk rather than many, small requests. If AppleShare Pro notices that the access patterns to a particular file are random rather than sequential, it will disregard this anticipatory scheme.

Guidelines for resizing the file caches. Follow these rules when adjusting the number and size of the AppleShare Pro file caches:

- By default, there are 10 buffers of 64 KB each.
- Allow one buffer for each active client. An active client is a person who is *currently accessing* a file, not someone merely logged in to the server.

The folder cache and icon cache

AppleShare Pro has a folder cache that stores the data from directory enumeration operations (listings of filenames and the date and time of creation for each file). AppleShare Pro also has an icon cache that stores the bitmaps of icons. You can adjust the size of both caches. The circumstances in which adjusting them is advisable are described in the following sections.

Guidelines for resizing the folder cache. Follow these rules when adjusting the size of the folder cache:

- The size of the folder cache should equal the total number of files and folders being shared, multiplied by .2 KB. For example, if your server shares 1000 files and folders, adjust the folder cache to 200 KB in size.

- If it's known that only a percentage of the files and folders being shared are usually active, then decrease the cache size by this percentage. For example, if 1000 files and folders are available for sharing, and only 50% are usually active, decrease the size of the folder cache from 200 KB to 50% of 200 KB, or 100 KB.

Guidelines for resizing the icon cache. Follow these rules when adjusting the icon cache:

- If your server shares many applications, you should increase the size of the icon cache to hold their icons. Increase the icon cache to approximately 25% of the size of the total file cache. If your server shares few applications, decrease the size of the icon cache to approximately 10% of the size of the total file cache.
- If clients of the server don't navigate by using the Finder, and none of the applications on the server display icons, then don't allocate RAM to the icon cache. Either set the icon cache to zero or deselect the icon cache. Both actions are performed in the File Server Cache Preferences dialog box, as described in the next section.

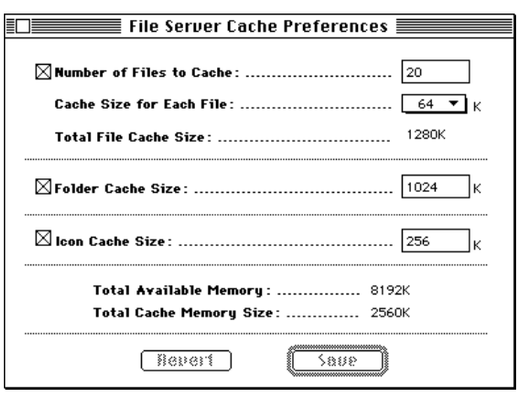
How to resize the AppleShare Pro caches

Follow this procedure to resize the caches:

1 Restart the AppleShare Pro Admin Program.

2 From the Server menu, choose File Server Cache Preferences.

The following dialog box appears.



3 **Configure the caches, as described in the preceding sections.**

For example, resize the caches, or deselect the icon cache if it is not needed. For a complete description of the settings in this dialog box, see *AppleShare Pro Administrator's Guide*.

4 **Restart AppleShare Pro File Server program so that the changes take effe**

Relationship between A/UX buffer cache and AppleShare Pr

The A/UX buffer cache is the primary disk cache on the Apple Workgroup Server 95. It caches all data accessed on both UNIX file systems and Macintosh hierarchical file systems. It is possible (and on some systems, probable) that for brief periods of time, data in AppleShare Pro's directory and icon caches will also be found in the A/UX buffer cache. For this reason, if memory constraints require the reduction of any of these caches to less than its default setting, it would be preferable in most situations, to reduce the size of the AppleShare Pro directory and icon caches than to reduce the size of the A/UX buffer cache.

In situations where the directory and icon caches are being utilized very well, and other file activity causes the A/UX buffer cache to be less efficient than the directory or icon caches would be in caching the data, reducing the size of the AppleShare Pro directory and icon caches may not be the best solution. Such a situation is most likely to occur if your server offers both AppleShare Pro and UNIX-based services, such as NFS.

Macintosh virtual memory

A/UX is the operating system of the Apple Workgroup Server 95. Therefore, all Macintosh applications run in the virtual Macintosh environment. In the virtual Macintosh environment, all Macintosh services, such as those that display windows, are provided by a set of A/UX processes. Thus, AppleShare Pro is run in the virtual Macintosh environment and receives services from A/UX. You can change the amount of memory available to the Macintosh applications running on the server by adjusting the size of the Macintosh virtual memory.

Guidelines for resizing Macintosh virtual memory

Follow these rules when resizing Macintosh virtual memory:

- By default, the segment of RAM allocated to Macintosh virtual memory is 16 MB in size. This setting is optimal if AppleShare Pro is the only Macintosh application run every day on the server, and if you haven't increased the size of any of the AppleShare Pro caches.
- If your day-to-day operations include running other Macintosh applications in addition to AppleShare Pro, such as a print service, you should increase the size of the Macintosh virtual memory.

How to resize Macintosh virtual memory

Follow this procedure to resize Macintosh virtual memory:

1. **Choose Control Panels from the Apple menu.**
2. **Open the Memory control panel.**
3. **Click on an arrow to increase or decrease the amount of memory allocated to segment of RAM.**

   Apple advises against allocating more memory to the Macintosh virtual memory than is built into your system. When virtual memory exceeds RAM, paging occurs regularly and degrades system performance. For more information on paging, see "A Performance Pitfall: Paging Through Virtual Memory" later in this chapter.

Running UNIX-based services

If your server provides UNIX-based services such as NFS, Mail, the X Window System, or FTP and TELNET, you need to take action to compensate for this additional activity. If you don't take precautionary steps, your clients may suffer from performance degradation due to the occurrence of paging. Paging, described more fully in the next section, is a process that is helpful in handling unusual loads on the server, such as the use of an infrequently run application, but is not to be used for day-to-day services.

You can minimize the occurrence of paging by taking one or more of these steps:

- Reduce the size of the A/UX buffer cache (described earlier in this chapter).
- Reduce the size of the AppleShare Pro caches (described earlier in this chapter).
- Add RAM to the system.

Your objective with these actions is to prevent the memory requirements of the UNIX processes and the running Macintosh applications from exceeding the amount of memory allocated to them.

All memory *not* used by the A/UX kernel and the A/UX buffer cache is available for use by other UNIX services.

A performance pitfall: Paging through virtual memory

Virtual memory is representational memory; it is not physical memory as is the RAM on a chip. The purpose of virtual memory is to extend the amount of memory in which programs can run. A/UX divides virtual memory into units called *pages* to facilitate copying virtual memory into RAM. Each page is stored in secondary memory (on a disk) until it is needed. The copying of virtual memory into RAM is known as *paging* or *swapping*. Each time an A/UX or Macintosh process is in secondary storage, paging occurs. Lots of paging incurs lots of disk I/O activity. Thus, virtual memory extends the RAM, but too much paging makes for poor system performance. Therefore, you should install and allocate RAM in your server so that paging occurs only in unusual circumstances, and not in the everyday operations of your server.

3 Optimizing Performance When Running AppleShare Pro and NFS

This chapter explains precautions you should take when configuring the Apple Workgroup Server 95 as both an NFS server and an AppleShare Pro server. It also alerts you to a problem that can occur when the server is configured as an NFS client and shares an NFS mounted file system by way of AppleShare Pro.

NFS and NIS software preinstalled

The Apple Workgroup Server 95 now ships with NFS and NIS software preinstalled, whereas the first release of the server did not. Both of these services extend the networking capabilities of the server. For more information on these services, see *Server Administration With A/UX 3.0.1*.

Table 3-1 lists the software packages preinstalled in this version of the Apple Workgroup Server 95 and indicates which packages were not preinstalled in the first version of the server. (You can reinstall the software that was preinstalled by using the Easy Install option in the Installer.) Note that in this version of the server, the preinstalled packages are the same for file and print servers as they are for database servers.

The AppleTalk Printer Access Protocol (PAP) UNIX Library is now preinstalled for the file and print servers and database servers. This library is included in the Basic C Programming package.

For information on completing the software setup for NFS, NIS, and TCP/IP, see *Setting Up and Managing Your Server,* Chapter 4, "Setting Up TCP/IP Networking." That chapter also describes the steps you must follow to configure your server as an NFS server or as an NFS client.

Table 3-1 Software packages preinstalled for the file server or database server

Packages preinstalled	Location	Size
Macintosh OS and A/UX Startup	MacPartition	10 MB
Core A/UX System	Root&Usr	45 MB
*Networking Capability	Root&Usr	4 MB
*Network Server Capability	Root&Usr	1 MB
*Manual Pages	Root&Usr	4 MB
*Basic C Programming	Root&Usr	4 MB
*Extended C Programming	Root&Usr	2 MB
*More UNIX Utilities	Root&Usr	1 MB
*UNIX Printing Utilities	Root&Usr	1 MB
*Debugging & Version Control	Root&Usr	1 MB

* Preinstalled in this release and not in the first release.

Operating as both an AppleShare Pro and an NFS server

When using the Apple Workgroup Server 95 as both an AppleShare server and an NFS server, be sure that the two services don't share the same files, folders, and volumes (file systems). If they do, you'll incur considerable problems because the two services use different means to assign and manage users, groups, and access privileges. You can use AppleShare to access an NFS shared file system without encountering any conflicts. The potential problem arises when an attempt is made to create or modify files that are shared by AppleShare and NFS and that reside on the same volume.

In addition, sharing the same volume through both services creates a security problem for users of NFS. Because AppleShare Pro runs from the root account, all AppleShare Pro users have root-level privileges on the volumes being shared by NFS and AppleShare Pro. Therefore, the AppleShare Pro users can read and make changes to all files on a volume shared by AppleShare Pro and NFS. Be aware of the vulnerability of the NFS files; to avoid potential problems, don't allow the same volume to be shared by both AppleShare Pro and NFS.

Running AppleShare Pro and operating the server as an NFS client

AppleShare Pro is designed to share local volumes; it is not designed to share volumes over the network. Therefore, if you configure the server as an NFS client, don't use it to share NFS mounted file systems. If you do, and the NFS server crashes, AppleShare Pro will become unresponsive.

AppleShare Pro is not programmed to respond to the NFS timeout errors. In A/UX, Macintosh virtual memory is an A/UX process that handles multiple client sessions. One of the many client sessions trying to access the unresponsive server can cause AppleShare Pro and all its clients to become unresponsive. (The other A/UX processes on the Apple Workgroup Server 95 will be unaffected.)

Operating as an NFS server

If your server is heavily used as an NFS server, you need to adjust your system to handle this type of load. Perhaps you've seen the following system messages that indicate you need to make such an adjustment:

```
'm_expand returning 0'
```

```
'panic: out of mbufs'
```

If you see either of these messages, you need to increase the number of buffers allocated for networking. You do this by changing the value of the kernel parameter NMBUFS. The default value for this parameter is 1000.

If you see the message

```
'panic: kmem alloc failed'
```

you need to change the value of the kernel parameter MAXCORE.

You change the value of a kernel parameter by using the `kconfig` (kernel configuration) command. To see the current values of your kernel parameters, open a CommandShell window and type

```
kconfig -av | more
```

For more information on using the `kconfig` command, display the on-line manual page for `kconfig` by typing `man kconfig` in a CommandShell window.

4 Preparing Your Server for Another Service

The Apple Workgroup Server 95 is tuned differently for each type of service it offers: file and print services, and database service. Therefore, if you want to change from running file services to running a database service, or the other way around, you need to prepare your system accordingly. To facilitate the changeover process, Apple has created a script that makes all the necessary changes to your system. This chapter tells you how to run the script and describes the changes that it makes.

Running the `config_server` script

Follow these steps to prepare your system to change from one type of service to the other:

1. **Log in to the root account.**
2. **Open a CommandShell window.**
3. **Type one of the following commands:**

   If you're changing your server to a database server, type

   ```
   config_server db
   ```

   If you're changing your server to a file and print server, type

   ```
   config_server fp
   ```

4. **Restart your system so that the changes take effect.**

You can now install the application program you wish to run.

Changes made by the script

The `config_server` script makes changes to the values of the kernel parameters NBUF, NCALL, NINODE, and NFILE. If you want to see the current values of these and other kernel parameters, enter the following command in a CommandShell window:

```
kconfig -av | more
```

The `config_server` script also affects the file `Autologin`. The presence of this file causes users to be automatically logged in to the root account when they start up the system. Servers configured for the file and print service use this file; hence the script creates this file if you change from a database server to a file and print server. Servers configured for a database service don't use this file; hence the script removes the `Autologin` file if you change from a file and print server to a database server. Subsequently, you must log in to the system manually.

Appendix Adding Internal Drives

Your Apple Workgroup Server 95 comes with either one or two 3.5-inch hard disk drives. The server can accommodate as many as five hard disks. Apple recommends that an Apple authorized service provider install additional drives in order to guarantee the continued coverage of the hardware under the Apple Limited Warranty. Unless other arrangements have been made, service providers should install the hardware and run the MacTest Pro diagnostic to confirm that the hardware is operational.

IMPORTANT If you configure the server with the maximum number of 2 gigabyte (GB) drives (five), the maximum ambient operating temperature is reduced to 35°C. All configurations with four or fewer drives can operate reliably at 40°C.

Before you start

In order to add drives to your server, you need to know how to configure the SCSI address selection indicator, or jumpers. In addition to the 3.5-inch, half-height drive(s) that you're installing, you'll also need these items:

- a grounding wrist strap and a static mat
- a Phillips screwdriver
- four screws to attach the drive to the bracket
- switches or jumpers as needed to set SCSI IDs for the drives you are installing

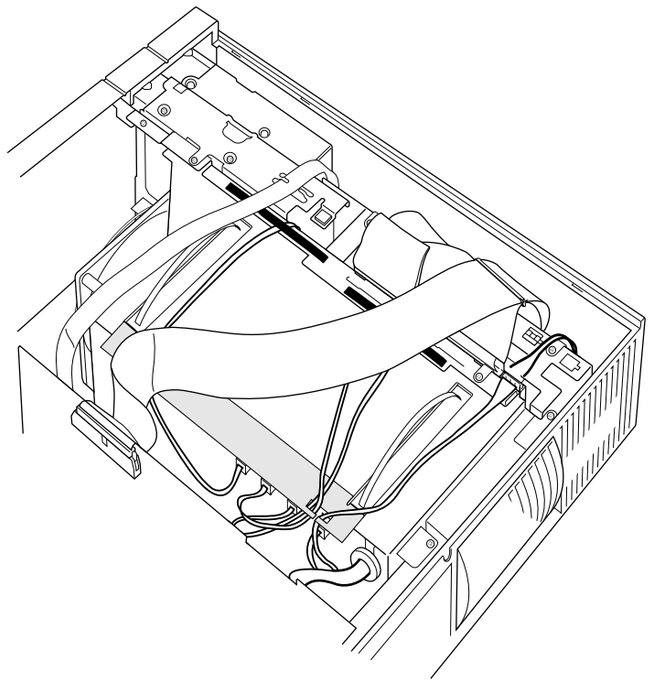
Note: Your Apple Workgroup Server 95 has built-in SCSI termination, so be sure that there are no resistor packs on the drive you are installing.

Installation procedure

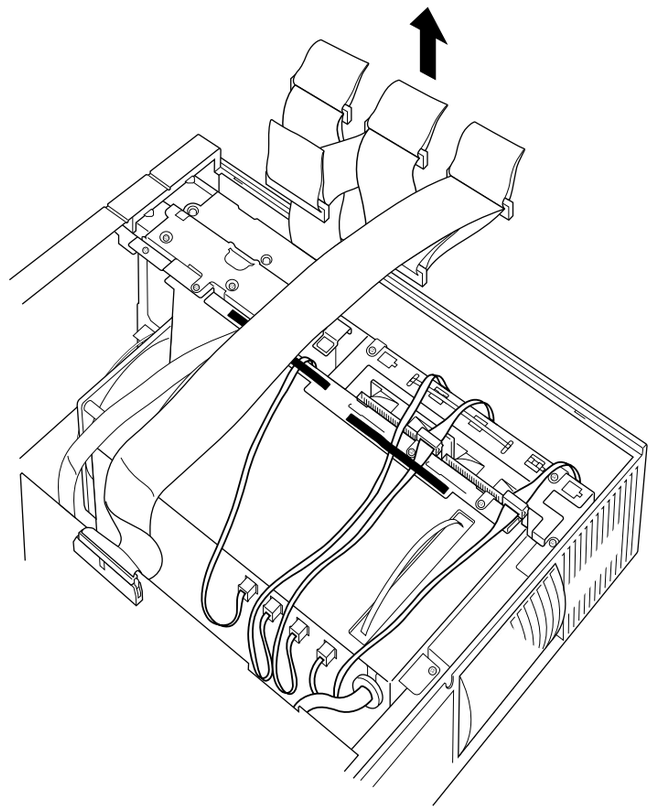
To install an internal hard disk drive:

1. **Turn off and unplug the computer.**
2. **Attach a grounding wrist strap and be sure to use a static mat.**
3. **Remove the cover from the computer.**

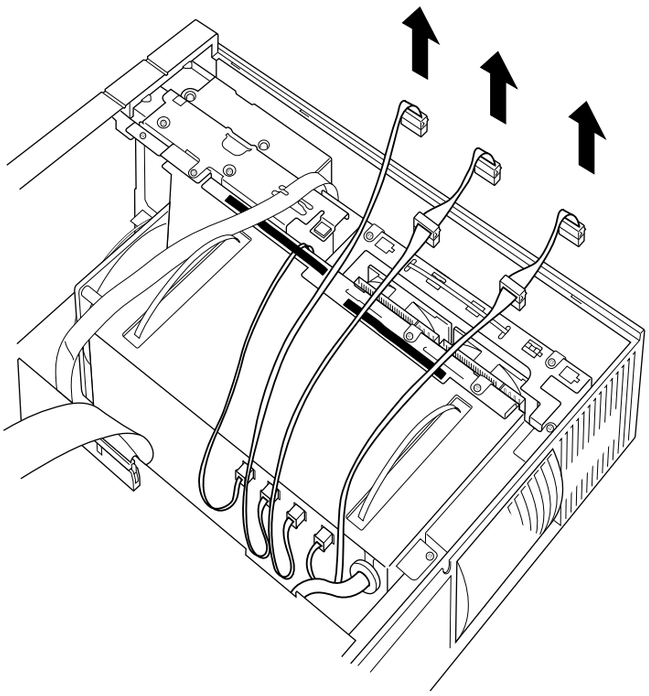
4 Place the computer on its side, with the drive bracket assembly screws



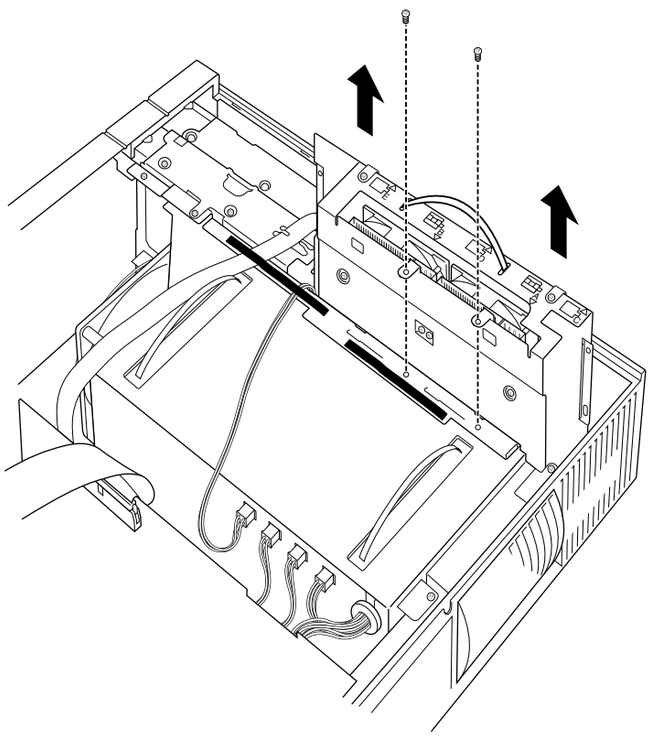
5 **Disconnect the SCSI cables from the existing drives. Your server came with two hard drives preinstalled.**



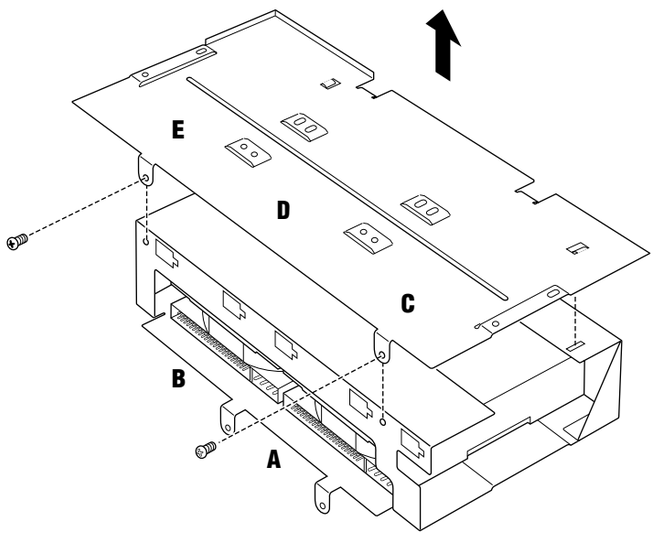
6 Open the cable holder straps and disconnect the power cables from the dr installed in the server.



7 Remove the bottom two drive bracket screws (the screws closest to the cable straps), then slide the drive assembly up and out of the computer.



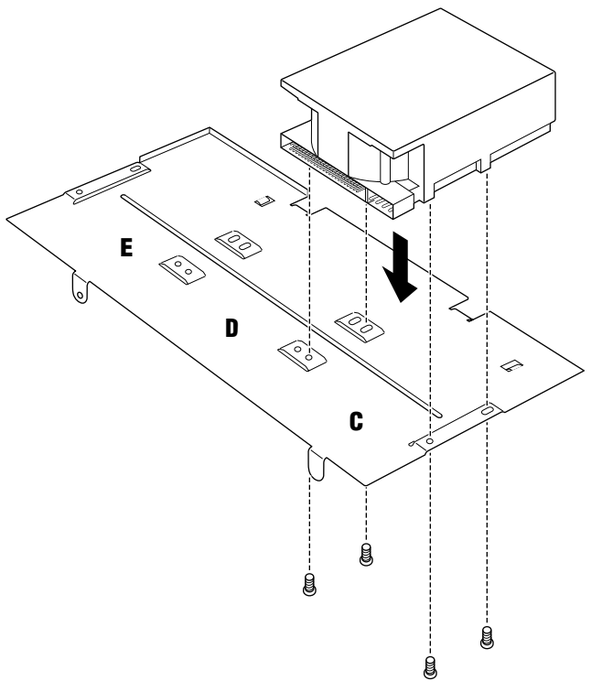
8 **If you are installing a drive in slot C, D, or E, remove the screws that ho of the drive assembly. Note that if you are adding a drive to slot A or B, the drive without removing the top bracket.**



9 **Make sure that you've installed any necessary SCSI ID jumpers or ID swit there are no resistor packs on the drives you are adding. (The Apple Wor Server 95 has built-in SCSI termination.) There are usually three resis they are yellow or orange.**

10 Attach the bottom of the drive to the bracket, using the four screw holes in the bracket plate.

IMPORTANT: Be sure to install drives in the order which they are lettered, filling slot B before slot C; slot C before slot D; and so on. Because of heat considerations, it's important that you use slot E only after the other four slots are filled.

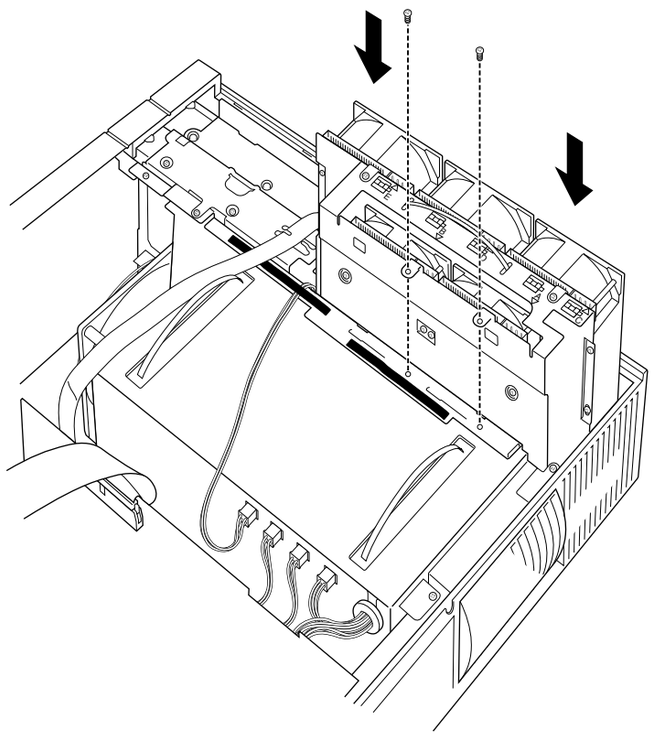


1 1 Set the SCSI ID numbers of the hard drives in such a way that they don't of another device. The SCSI ID numbers always available for hard drives and 6, as shown in the following illustration.

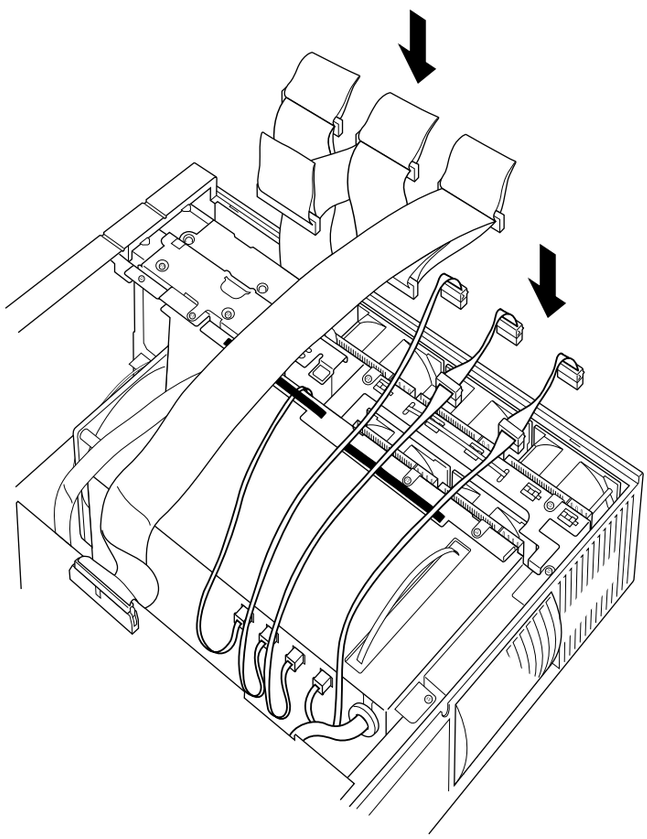
	SCSI ID numbers						
	0	1	2	3	4	5	6
SCSI BUS 3 (Internal/ PDS card)	Startup disk		Tape drive*	CD-ROM drive*			

* If installed.

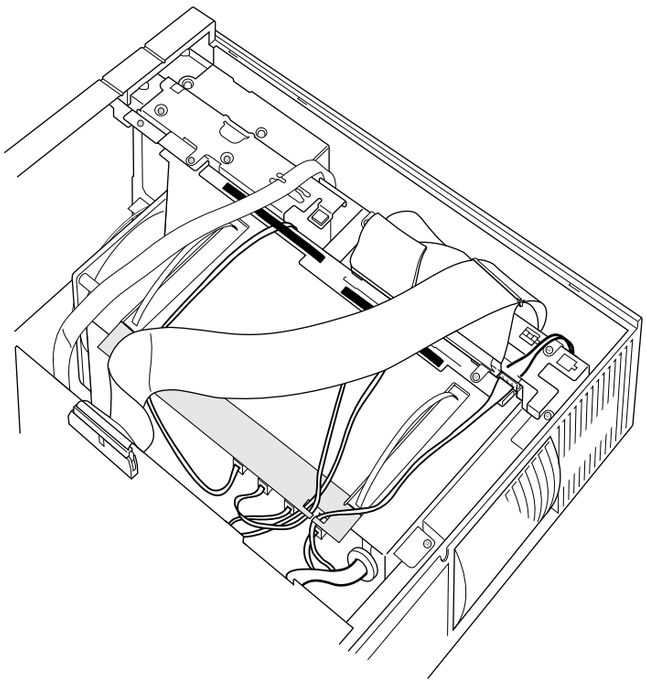
1 2 Slide the drive assembly back into the computer and attach the screws.



1 3 Connect the power cables and the SCSI cables. Note that the lettered tags on t cables match the letters on the drive bracket.



1 4 Secure the cables, as shown below. This step is necessary in order to rep



1 5 Replace the cover and reassemble the server.

Index

A

active client 4
aggregate throughput 3
allocating RAM, basic process for 8
AppleShare
  maximum client load 4
  performance comparison among versions 3
AppleShare Pro
  active clients 4
  buffers 10–12
  cache sizes and UNIX-based services 13
  caches, resizing of 12
  changing the server from a database application to 21–22
  directory enumeration operation 5
  File Server Cache Preferences dialog box 12
  folder cache 11
  icon cache 11–12
  listing files 5
  and NFS client 19
  and NFS mounted file systems 16
  and NFS server 19–20
  reallocating memory to 13
  remote volumes 19
  resizing caches 12
  rules for resizing its caches 11
  speed 3
  throughput capacity 3
A/UX buffer cache
  and AppleShare caches 13
  and database applications 10
  default size of 10
  described 9
  and a file server 10
A/UX kernel, RAM requirements of 8

B

benchmarks 1–2
buffer cache, A/UX 9–10
buffers
  and an NFS server 19
  in AppleShare Pro 10–13

C

caches
  in A/UX 9–10
  in AppleShare Pro 10–13
clients, maximum number for server 4
computer publications and benchmarks 2
`config_server` script 22

D

database applications, and the A/UX buffer cache 10
database service, preparing the server for 22
default software packages installed 18
directory enumeration operation 5

E

enumeration operation 5

F, G, H

file and print service, preparing the server for 22
hard drives, installing internal 23–33

I, J

installing internal hard drives 23–33

K

kernel parameters 20
  changed by `config_server` script 22
`kconfig` 22

L

listing files, speed of 5
load, and paging 14–15
load on server for optional performance 4

M

Macintosh applications, allocating memory for 13
Macintosh virtual environment 13–14
Macintosh virtual memory 13–14
main memory. *See* RAM
measuring performance 2
  and replicating Apple's tests 2
memory. *See* RAM
`m_expand returning 0` message 20

N, O

network connections and performance 8
network software
  preinstalled 18
  UNIX software and AppleShare Pro 19
network throughput capacity 2
NFS 18, 19, 20
NFS and AppleShare Pro 19
NFS server
  and AppleShare Pro 19
  buffer allocations 20
  system messages 20
NFS services and resizing caches 13
NFS system messages 20
NIS 18

P, Q

paging 14
`panic:kmem alloc failed` message 20
`panic:out of mbufs` message 20
performance
  acceptable speed per client 4
  aggregate throughput 3
  and network connections 8
  and number of clients 4
  comparison of AppleShare 3.0 and AppleShare Pro 3
  how achieved by AppleShare Pro 4
  maximum for server 4
  measuring 2
  user studies of 4
physical memory. *See* RAM
preinstalled software packages 18
print service, allocating memory for 14

R

RAM
  allocating for Macintosh applications 13–14
  allocation process 8
  for AppleShare Pro caches 10–13
  basic allocation of 8
  components competing for 9
  how used by the A/UX buffer cache 9, 10
  and paging 14–15
  and swapping 15
  and UNIX-based services 14–15
  and virtual memory 13
  when to add 8
resizing AppleShare Pro caches 11, 12
resizing A/UX buffer cache 10
resizing Macintosh virtual memory 14

S

script to prepare server for other service 22
SCSI IDs and internal drives 24, 31

server capacity 2
sharing files over NFS and AppleShare 19
software, preinstalled 18
swapping. *See* paging
system messages, NFS 20

T

test results of Apple's performance tests 3
throughput capacity 2, 3

U

UNIX kernel parameters 20, 22
UNIX networking software 18
UNIX-based services
    and RAM allocations 14–15
    resizing caches for 14–15
user response time 2

V, W, X, Y, Z

virtual Macintosh environment 13–14
virtual memory, Macintosh 13
virtual memory and paging 15

The Apple Publishing System

Tuning Server Performance and Adding Internal Hard Drives was written, edited, and composed on a desktop publishing system using Apple Macintosh computers, an AppleTalk network system, Microsoft Word, and QuarkXPress. Line art was created with Adobe Illustrator. Proof pages were printed on Apple LaserWriter printers. Final pages were output directly to 70-mm separated film on an Electrocomp 2000 Electron Beam Recorder. PostScript™, the LaserWriter page-description language, was developed by Adobe Systems Incorporated.

Text type is Times®, display type is Helvetica® Narrow, and cover type is AppleGaramond, Apple's corporate font. Ornaments are custom symbols designed for Apple Computer. Some elements, such as computer voice, are set in Apple Courier, a fixed-width font.