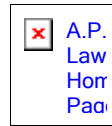


Search
FAQ
New to Sco
New to Unix
New Here
SCO News
Unix Articles
Unixware
Linux
Security
Code
Tests
Links
Books
Reviews
Opinion
Find a Consultant
Find a Job
Site Forum
Donations
Contact Info
Services
Sales
Consulting Rates
Advertising Rates
Site Map



A.P. Lawrence Home

Over 10,600 Unix/Linux users so far this month (48,400+ page views)



Adding more space

[More Articles](#)

Sooner or later it happens: you type "df -v" and see that the "%used" is pushing 99%. Or, worse, you get awful messages on the console: No space on device 1/40.

(If you are out of space right now, follow this link for suggestions as to [what to do right now](#)).

So, it's time to add more disk space. There are two basic ways to approach this problem. You could replace the current disk with something larger, or simply add another disk. Both methods have their own positive and negative aspects. Let's consider some of them:

Replace the current disk

This can be the simplest solution of all, particularly if you are using emergency boot and recovery disks such as can be made by [Microlite Edge](#) or other similar products. Do a backup, pop out the old disk, in with the new, boot with the Recovery disks, and very quickly you have a running system.

But this assumes that you either have some other use for your current disk, or that it's old and worthless enough that you are willing to toss it out. That may be the case, but in those situations, you may have a further problem if you have to replace the old disk with a different architecture. You may have an old IDE disk, and want to replace it with a new SCSI Raid system, for example. In those situations, the Emergency Recovery disks may not be enough, and you may have to start with a fresh install of the OS and proceed from there.

Add another disk

Adding storage can, of course, involve similar trade-offs. You could add a SCSI drive to an existing IDE drive, for example, but generally not vice-versa. You also need to decide what information you are going to store on that new space, and how you are going to get it there. Some programs have the ability to make use of new space fairly simply, but more often than not, you are going to have to do some data rearranging.

See [Adding a disk drive](#) for details on how to add the disk

Dividing the disk

Whichever way you decide to go, you need to start thinking about partitions before you do anything. Partitions, as you may now, are the areas that your hard drive is divided into. It can be a little confusing, because there are really two kinds of divisions: "fdisk" partitions and "divvy" partitions (which are called "divisions" by SCO, but "partitions" by some other Unices, which is why it gets confusing).

If you have an older version of SCO, you may have to be concerned about fdisk partitions for very large drives. More modern versions are able to support about anything you could conceivably purchase (assuming proper drivers, of course). If you have SCO 3.2v4.2, you will probably need or want **uod429a** for EIDE or large SCSI drives. This supplement has some confusing aspects to it, so read the README file carefully and completely.

I'll assume for the moment that you have no issues that cannot be resolved by proper drivers or supplements, and that you will be creating one Unix fdisk partition. It's now time to think about divvy divisions.

These divisions divide your Unix partition into filesystems or raw disk partitions. Raw partitions are strictly for the use of programs like Informix that are written to use these partitions, generally to obtain higher performance. Without such programs, your divisions will be used for ordinary Unix filesystems. Older systems usually had just a root division or possibly a root and a **/u** (The "root" division is the one that shows up as "/" mounted on "/dev/root"). Release 5 systems usually have a root and a small **/stand**. You need to decide what partitions you will create, and how large they will be.

On older releases, the OS constrains your choice to some degree because 3.2v4.2 and below simply cannot handle a filesystem larger than 2 gigabytes. If you had a raw partition, that could be larger, but any filesystem prior to 3.2v5.x has to be 2 gigabytes or less.

Why divide?

Assuming Release 5, or a 4.2 system with a 2 gig drive, why not dispense with extra divisions and simply have one large file system?

You could do that. With release 5, you'll probably still want a small (15-20MB) /stand filesystem, because Unix can't boot from an HTFS filesystem, and that's what you would want the root to be. The /stand would be an (older) EAFS filesystem, which Unix can boot from, and the root would be the higher performance and more reliable HTFS type.

Even on 3.2v4.2, you could have one large root filesystem (up to 2 gigabytes). It's very convenient to do this; you don't have to worry about how much space to allocate to what, you just let the whole disk be one big filesystem.

For several reasons, you probably don't want to do that. On the older releases (and even on the newer, under some conditions), big file systems can take a long time to clean and repair in the event of an unexpected crash. It is also true that damage to a hard drive usually confines itself to one file system. In other words, if a defective hard drive trashes your root filesystem, a

separate /u file system may escape damage entirely. This can make recovery simpler and more complete, because the data on /u may be current right up to the crash.

Deliberately limiting space

You may also want to create file system divisions to specifically limit how much space can be used. At first, this may seem silly, but consider the often seen case where a naive user keeps submitting the same large print job over and over, not understanding that it is not printing for some other reason (printer offline?) and that each submission is using up more disk space. Consider also a runaway program that keeps creating a larger and larger file. Wouldn't it be nice to let a separate file system run out of space before the whole drive is used up? In the case of the printing, you could create a small division and mount it at /var/spool/lp/temp.

Note that you have to be a little bit careful about creating partitions that will be used for system directories, like /user, and on Release 5, the mess of symbolic links means you have to tread even more carefully. With enough care and forethought, you **can** do whatever you need to do. You have to watch out for things that need to be present when the system is booting; if you are planning on moving anything like that elsewhere, you will have to leave some of it behind, ultimately to be hidden beneath your new file system. This can get tricky, so be careful or hire competent help.

Deciding what to move

You may have one large area that will continue to grow. For example, /usr/mas90, /appl/filepro, and /usr/rwc65 are common SCO programs that have defined directory structures. If you have or can identify such an area, then this is the data you will move.

You may, however, have two or more areas to move. You can solve this by creating multiple file systems in the new space, or by using one file system and symbolic links to point to the new data areas.

There are things you cannot move. You cannot, for example, move /usr or any part of it that needs to be present as the machine boots or in single user mode. **DO NOT MOVE ANYTHING THAT ISN'T YOURS** and you'll be safe.

Moving Data

If you are installing a replacement drive, you are not concerned with moving data. If, for example, your old system had a large /data filesystem, you could simply specify a larger division when divvy'ing the new disk. The restore will simply put the old data in the new, larger, division.

But when adding a new hard drive, you may want to move the contents of a directory to the new space. The following procedure is useful:

Create the new space with "mkdev hd" as required. Relink, and reboot. Rename the directory you are going to move, using the "mv" command. For example:

```
mv /usr/data /usr/data.safe
```

Note the ownership and permissions of `/usr/data.safe`, and create a new `/usr/data` directory with the same permissions and ownership. Example:

```
# cd /usr
# mv data data.safe
# ls -ld data.safe
drwxr-xr-x  2 root      sys          512 Nov 16 18:47 data.safe
# mkdir data
# chown root data
# chgrp sys data
# chmod 755 data
# ls -ld data
drwxr-xr-x  2 root      sys          512 Nov 16 18:47 data
```

Run "mkdev fs", giving the name of the directory you want to move as the mount point (`/usr/data`). Then type:

`mountall` to mount the new space at `/usr/data` (or simply type "mount `/usr/data`").

Type the following to copy the data:

```
# cd /usr/data.safe
# find . -depth -print | cpio -pdlmv ../data
```

After this is done, and you are completely satisfied that everything is working as it should be, you can "rm -r `/usr/data.safe`"

If you are using symbolic links, the procedure is similar. First, create a new directory on the other drive:

```
mkdir /bigdrive/newdata
```

Assign appropriate permission to `newdata`.

Copy the data as before, and mv the oldname aside:

```
cd /usr/data
find . -depth -print | cpio -pdlmv /bigdrive/newdata
mv /usr/data /usr/data.sf
```

Now create the symbolic link:

```
ln -s /bigdrive/newdata /usr/data
```

Again, when you are satisfied that everything is working, come back and "rm -r `/usr/data.safe`".

See also:

- [Filesystems](#)
- [Disk space](#)

Was this information valuable to you? Consider making a [contribution](#) to help pay for the costs of bringing it to you.

[Amazon Honor System](#) it's fast, safe and easy. It's also very private- Amazon doesn't share your information with anyone: I don't even get to find out who contributed!



Related Articles:

[RedHat RPM Source Packages](#) (Linux, Administration, Programming)
[User Friendly Date Script](#) (Linux, Administration, Code)
[Connecting to the Internet](#) (Networking, Security, Administration, Windows)
[Synchronics Counterpoint on Red Hat Linux](#) (Linux, Administration, Windows)
[Unix Backup](#) (Administration, Backup)
[Shell Bashing](#) (Administration, Basics, Linux)
[SSH](#) (Administration, Security, Networking)
[Hylafax for OSR5](#) (Administration, Reviews)
[Transferring Data from an Old Hard Drive](#) (Disks/Filesystems)
[Installing 5.0.6](#) (Administration, Reviews)
[Monitoring Windows NT with Unix syslog](#) (Reviews, Windows, Administration, Networking)
[Adding a Hard Drive to Linux](#) (Linux, Basics, Disks/Filesystems)
[Hard Disks](#) (Basics, Disks/Filesystems)
[Loose Cable](#) (Disks/Filesystems)
[Floppies](#) (Basics, Disks/Filesystems)
[Configuring anti-spam in RedHat 7.0](#) (Mail, Administration, Security)
[Monitor users with Peek](#) (Modems/Terminals, Administration)
[Mail in a Virtual Domain](#) (Mail, Administration)
[Kernel Link Failures](#) (Administration, Kernel/Internals)
[Installing SNMP](#) (Administration, Networking)
[Changing System Name](#) (Administration, Networking)
[General Trouble Shooting](#) (Basics, Administration)
[The Gimp](#) (Misc., Administration)
[Windows Network Configuration](#) (Networking, Windows, Administration)
[Majordomo](#) (Administration, Mail)
[Bootting OSR5- Filesystems](#) (Kernel/Internals, Disks/Filesystems)
[LDAP Basics](#) (Networking, Administration)
[Bootting OSR5](#) (Kernel/Internals, Disks/Filesystems)
[Bootting OSR5- Definitions](#) (Kernel/Internals, Disks/Filesystems)
[Bootting OSR5- Swap and Dump](#) (Kernel/Internals, Disks/Filesystems)
[Data Transfer](#) (Misc., Administration)
[Star Office on SCO](#) (Administration)
[Setting up Internet Mail](#) (Administration, Mail)
[Understanding RAID](#) (Disks/Filesystems, Basics)
[Understanding SCSI](#) (Disks/Filesystems, Basics)
[Adding Memory](#) (Administration)
[DPT Raid Controller](#) (Reviews, Disks/Filesystems)
[Configuring a News Server](#) (Administration, Mail)
[Setting up a Small Office Network](#) (Networking, Administration, Security, Windows)
[OSR5 PPP Setup](#) (Networking, Administration)
[Installing 5.0.5](#) (Administration)

[Veritas \(Online Data Manager\)](#) (Unixware, Disks/Filesystems)
[Performing OS Upgrades](#) (Administration)
[Configuring SCO's GUI](#) (Misc., Administration, Basics)
[Setting up High Speed Modems](#) (Modems/Terminals, Administration)
[Out of Disk space](#) (Disks/Filesystems)
[Adding another hard drive](#) (Administration, Disks/Filesystems)
[bfind.c- Finds which file contains a block](#) (Programming, Disks/Filesystems, Kernel/Internals, Code)
Books: [SSH- The Secure Shell](#) (Security, Administration, Networking)
Books: [Hacking Exposed \(Second Edition\)](#) (Web/HTML, Security, Linux, Windows, Networking, Administration)
Books: [Perl for System Administration](#) (Administration, Perl, Windows)
Books: [Unix and Windows 2000 Integration](#) (Administration, Windows)
Books: [Using Samba \(O'Reilly\)](#) (Administration, Windows)
Books: [\(Sun\) Resource Management](#) (Administration)
Books: [Samba](#) (Administration, Windows)
Books: [Sendmail Theory and Practice](#) (Administration)
Books: [Windows NT and Unix Integration](#) (Administration, Windows, Networking)
Books: [UnixWare 7 System Administration](#) (Administration, Sco Specific)
Books: [Essential SCO Unix Administration](#) (Administration, Sco Specific)
Books: [Sendmail Pocket Reference](#) (Administration)
Books: [Unix for the Impatient](#) (Administration, Good Starting Books, Programming)
Books: [The Unix Programming Environment](#) (Administration, Programming)
Books: [SCO Performance Tuning Handbook](#) (Administration, Sco Specific)
Books: [Essential System Administration \(Unix\)](#) (Administration, Good Starting Books)
Books: [The Devil Book](#) (Administration, Kernel/Internals)
Books: [Configuration and Capacity Planning](#) (Administration, Kernel/Internals)
Books: [SCO Companion](#) (Administration, Good Starting Books, Sco Specific)
Books: [The Design of the Unix Operating System](#) (Administration, Kernel/Internals)

© 1998 Anthony Lawrence. All rights reserved.

This article is copyrighted material. You have permission to use it for any purpose, commercial or non-commercial, as long as it is kept intact and no modifications, additions, or deletions are made except as allowed herein.

You may publish it in paper or electronic form. That includes magazine, newsletters, and web pages, both internal and external, for profit or not. Banner ads and other graphics may be removed, but all other text, hyperlinks and copyright notices, including this, must remain (but see below also). You may not delete text, alter it, or add to it in any way that does not clearly delineate what is yours and what comes from this site. You may alter fonts, font sizes and the like and reformat text as is appropriate for your use.

You may select specific paragraphs or sections, but if you do so, you must include this entire notice also, noting that you have not published the entire article, or simply note that the paragraphs you have published are part of a larger article and give the http address of the actual article.

My main concern is that no one would be confused that you wrote something I wrote or vice-versa. If your use meets that concern, and allows people to find the original article here, I have no objection.

This general permission specifically does NOT apply to test questions and answers.

Some articles at <http://www.aplawrence.com/> and <http://www.pcunix.com/> are copyrighted by other individuals or corporations; these paragraphs do not apply to those articles even if accidentally included.

We do appreciate being advised of any such use: Email: tony@aplawrence.com.