Revised: January, 2002

Advanced File System Quick Reference

Advanced File System (AdvFS) Quick Reference

Welcome. This site provides a quick overview for working with your AdvFS file system.

AdvFS Quick Reference is hands on and task oriented. Additional documentation, often including detailed examples, is referenced at the end of most sections.

Click on the titles in the left-hand column to navigate.

The Tru64UNIX publications page contains pointers to Tru64 UNIX Documentation, where you can access manuals and reference pages for all versions of the operating system. *AdvFS Quick Reference* assumes that you are familiar with AdvFS. If you are not, see the documentation for your version of the operating system and follow the links to System and Network Management Documentation Bookshelf and then *AdvFS Administration*.

Advanced File System Quick Reference

Planning

Each installation is unique. AdvFS provides the flexibility to handle a wide range of user environments.

Configuration and Layout

- Domain and fileset organization
- Volume organization
- RAID configurations
- Limiting disk usage
- Clusters
- Data consistency
- Using SysMan for system management

New Features Version 5 Operating System Software

- Improved on-disk structure
- Enhanced `vdump` and `vrestore` with remote capability
- Increased quota limits
- Direct I/O
- The `vdf` utility
- Metadata display utilities

New Features Version 5.1a Operating System Software

- The `fixfdmn` command
- Temporary atomic-write data logging
- Object safety (object reuse) protection
- Frag file on/off

Advanced File System Quick Reference

Planning:Configuration and Layout

Before allocating file system space, it is useful to plan what the system will look like. The following are some items to consider when determining system structure:

- Domain and fileset organization
- Volume organization
- RAID configurations
- Limiting disk usage
- Clusters
- Data consistency
- Using SysMan for system management

Domain and fileset organization

There are many ways to organize your file system. Domains are the physical storage layer of AdvFS. This storage is managed separately from the directory structure, so you can expand and contract the size of a domain, and you can move files between disks in the domain without changing file path names.

A fileset is the mountable portion of the directory hierarchy. A fileset name must be unique within its domain. Filesets share the available space on the volumes of the domain and share a transaction log. Each fileset has its own directory structure.

You can create one domain that contains numerous filesets or you can create a domain for each fileset; you can also structure your system anywhere between these two extremes. If you have AdvFS Utilities, you can distribute your domains on more than one volume.

If you have enough drives, use an entire disk for a domain. For example, do not place partition a in one domain and partition b in another domain. Add one partition (typically, partition c) to a domain rather than several partitions on one disk (such as a, b, g, h).

The following table examines the tradeoffs in configuring domains and filesets.

Domain and Fileset Configuration Tradeoffs

Configuration	Advantage	Disadvantage
If your configuration is large enough, use entire disk(s) for a domain, not small partitions on several physical devices	Efficiency; device's I/O stream is not shared	Waste of space if small domains are on large volumes
Configure domain on multiple volumes (requires Advanced Utilities) rather than on a single volume	Distributes I/O load; increases throughput	Failure of a single volume makes the domain inaccessible
Many smaller domains instead of one large domain (assuming the number of volumes stays the same)	Faster recovery; improved throughput; might decrease some administrative tasks	Might increase administrative overhead

Many filesets in a domain instead of one large one	Can set quota limits on each fileset for finer-grained control of resources	Administrative overhead; all filesets must be mounted to run some utilities
Place transaction log on fast or uncongested volume	Prevents log from becoming I/O bottleneck	Cost of hardware
Configure root fileset as AdvFS	Fast recovery after system crash	None

In addition:

- Use the `showfsets` command to display the number of filesets in a domain and the size of the filesets.
- Use the `showfile` command to determine which volume a file resides on. Make sure that busy files are not located on the same volume.
- Use the `migrate` command to move files to different volumes.
- If you are using Logical Storage Manager (LSM), use either multiple, small LSM volumes or a single, large striped LSM volume. Either balances I/O across volumes. Avoid using concatenated volumes.
- The `defragment` utility processes each volume in parallel. Thus, creating a domain on several volumes can speed up the defragmentation process.
- Use software (LSM) or hardware RAID to mirror the volume on which the log resides. This insures that the domain is not compromised in the event of a disk failure.

For limits on the number of active filesets, domains, and so forth, see the Software Product Description for your version of the operating system.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Configuring the File System"
Reference Pages: `defragment(8)`, `showfile(8)` and `showfsets(8)`

Version 4.x: Advanced Products CD contains the reference page for the AdvFS Utilities `migrate(8)` command.

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Configuring the File System"
Reference Pages: `defragment(8)`, `migrate(8)`, `showfile(8)` and `showfsets(8)`

Top of Page

Volume organization

A volume is the entity that is used by AdvFS to add and remove space in a domain. A volume is a block special device (such as a single disk partition, an entire disk, an aggregate volume provided by LSM, or hardware RAID). Each volume can be assigned to only one domain. If you have the optional AdvFS Utilities, you can add more than one volume to a domain.

The following table examines the tradeoffs in configuring domains on AdvFS volumes.

AdvFS Volume Configuration Tradeoffs

Configuration	Advantage	Disadvantage
Volume is a disk partition	Uses small portion of disk	Limited size; I/O contention with other partitions
Volume is a single disk	More efficient than using single partitions	Limited size
Volume is a Logical Storage Manager (LSM) or RAID volume	Can use mirroring, striping, parity to improve performance and reliability; increased capacity	Cost: requires LSM license or RAID hardware

Top of Page

RAID configurations

RAID (redundant array of independent disks) can be implemented either in hardware or software (through LSM). RAID technology improves the reliability and availability of your system through disk mirroring, striping, and parity calculations.

Software RAID is a good choice for small configurations because it does not require expensive devices. See also RAID in Setup:Volumes.

Hardware RAID solutions range from low-cost backplane RAID array controllers to high-performance array subsystems that support not only a range of RAID functions, but also support scalability, battery backup, and the ability to swap damaged components on line.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
Logical Storage Manager
System Configuration and Tuning

Top of Page

Limiting Disk Usage

Set quotas to limit the amount of space a user, group, or fileset might consume. AdvFS quota information is always available (that is, you cannot delete quota files), but quotas may be enabled or disabled. If you do not set fileset quotas, any fileset can capture all resources in a domain.

The root user is not limited by quotas.

For more information see Setup:Quotas.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration

"Managing Quotas"

Version 5.x Tru64 UNIX publications page

System and Network Management Documentation Bookshelf:

AdvFS Administration

"Managing Quotas" Top of Page

Clusters

AdvFS is the primary file system supported in the cluster. All file system entities are available on any node of the cluster.

For most operations, clusters are transparent to AdvFS. For more information, see the TruCluster documentation.

Top of Page

Data consistency

You can change the way AdvFS performs its transaction logging and data caching to improve data consistency in the event of a system crash. The following table shows the advantages and disadvantages of each scheme.

Improving Data Consistency

I/O Method	Advantage	Disadvantage
Asynchronous I/O (default) - data cached and written to disk later	High performance	Possibility of inconsistent user data after crash
Asynchronous atomic-write data logging - data written asynchronously to the log in 8-kilobyte increments	If crash occurs, guarantees 8-kilobyte increments of user data are written to disk and no out-of-order disk writes occur	Might be slower than asynchronous write performance because data is written to the log as well as to the file
Synchronous I/O - data cached and written to disk	Guarantees that user data was written to disk before the write return	Slower than asynchronous I/O because the call does not return until the data has been written to disk
Synchronous atomic-write data logging - same as asynchronous atomic-write but the logged data is also flushed to disk	If crash occurs, guarantees 8-kilobyte increments of user data are written to disk and no out-of-order disk writes occur. If `write()` call returns before crash, guarantees all user data was written to disk.	Slower than asynchronous write performance because data is written to the log as well as to the file

Version 5.1A: Atomic-write data logging can be turned on and off temporarily. Use the `mount -o adl` command to set an in-memory flag that activates atomic-write data logging in a fileset for the duration of the mount. Persistent atomic-write data logging, which is turned on and off with the `chfile` command on a per-file basis, takes precedence over temporary atomic-write data logging while the file is open.

See also:

For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page

System and Network Management Documentation Bookshelf:

AdvFS Administration
"Improving Data Consistency"
"Temporary Atomic-Write Data Logging"
System and Configuration Tuning
"Forcing Synchronous Writes"
"Preventing Partial Data Writes"

Version 5.1A Tru64 UNIX publications page
Reference Pages: `chfile(8)` and `mount(8)`

Top of Page

Using SysMan for system management

System management utilities are accessible from the command line and from the Common Desktop Environment (CDE) front panel. You can use SysMan utilities to perform routine administrative tasks.

From the SysMan menu double-click to choose Storage then File Systems Management Utilities. The following utilities are available:

- General File System Utilities
  - Dismount a File System
  - Display Currently Mounted File Systems
  - Mount File Systems
  - Share Local Directory (/etc/exports)
  - Mount Network Directory (/etc/fstab)
- Advanced File System (AdvFS) Utilities
  - Manage an AdvFS Domain
  - Manage an AdvFS File
  - Defragment an AdvFS Domain
  - Create a New AdvFS Domain
  - Create a New AdvFS Fileset
  - Recover Files from an AdvFS Domain
  - Repair an AdvFS Domain

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
System Administration
Reference Pages: `sysman(8)`

Top of Page

Advanced File System Quick Reference

Planning:New Features in Version 5 Operating System Software

The following features have been added to AdvFS:

- Improved on-disk structure
- Enhanced `vdump` and `vrestore` with remote capability
- Increased quota limits
- Direct I/O
- The `vdf` utility
- Metadata display utilities

Improved on-disk structure

Version 5.0 and later operating system domains have a new metadata structure that increases performance for directories containing thousands of files. All domains that were created prior to Version 5.0 are recognized by later versions but are not automatically upgraded to the new structure.

The Version 5.0 operating system is the first for which a new domain carries a domain version number (DVN) of 4. Domains created earlier carry a DVN of 3. The improved directory structure increases the speed of file creation and access.

To create a new Version 5 domain see Creating a domain in Setup:Domains. To upgrade a domain to the new structure see Upgrading a domain to DVN4 in Setup:Domains.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Version 5.0 Domains"

Top of Page

Enhanced vdump and vrestore with remote capability

The `vdump` and `vrestore` dump file formats are forward compatible for Version 4.x and Version 5.x operating systems. This means that if you have a DVN3 domain (that is, created on a Version 4.x operating system) running on a Version 5 system, you can dump it as you would any DVN4 domain.

Version 5.0 and later `vdump` and `vrestore` utilities identify the filesets that are saved. This allows you to check that you are restoring from the correct fileset.

The `rvdump` command backs up files from a single mounted fileset or a clone fileset to a remote storage device. You must be able to execute the `rsh` command on the remote node to which you are dumping. The `rvdump` command has the same options as the `vdump` command, but you must specify the node name for the device that you are backing your files to. The `rvrestore` command works in the same manner.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
Reference Pages: `rsh(1)`, `vdump(8)`, and `vrestore(8)`

Top of Page

Increased quota limits

Limits are set on disk usage (number of blocks) or on number of files (inodes) or both. Quota size limits are as follows:

Quota Limits

	User and Group	Fileset
Disk Usage	8 billion terabytes*	4 billion terabytes
# Files	4 billion	4 billion

* Prior to Version 5.0, the disk usage limit for user and group quotas was 2 terabytes. If your domain was created prior to Version 5.0, and if you are now running Version 5.0 or later, you must upgrade your domain (see Upgrading a domain to DVN4 in Setup:Domains) if you want to take advantage of the new limits.

Top of Page

Direct I/O

Direct I/O, which is new for Version 5 operating systems, gives you the performance benefits and speeds of raw disks within the file system structure. When direct I/O is enabled for a file, read and write requests on it are executed synchronously to and from disk storage through direct memory access (similar to raw I/O), bypassing AdvFS caching. This might improve the speed of the I/O process for applications that access data only once.

Direct I/O is particularly suited for files that are used exclusively by a database. However, if an application tends to access data multiple times, direct I/O can adversely impact performance because, without caching, I/O cannot be pooled. As soon as you specify direct I/O, it takes precedence and any data already in the buffer cache for that file is automatically flushed to disk.

The best performance with direct I/O occurs when the requested transfer is aligned on a disk sector boundary and the transfer size is an even multiple of the underlying sector size. Larger transfers are generally more efficient than smaller ones, although the optimal transfer size depends on the underlying storage hardware.

To open a file for direct I/O, use the `open()` function and specify the `O_DIRECTIO` flag. If the file is already open for direct I/O or is in cached mode, the new mode is direct I/O and remains so until the last close of the file. Note that direct I/O, atomic-write data logging, and memory mapping are mutually exclusive modes. Therefore, if the file is already open for atomic-write data logging or is memory mapped, then calling the `open` function to initiate direct I/O fails.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Improving Data Transfer Rate with Direct I/O"
Programmers Guide
"Using Direct I/O on AdvFS Files"

Top of Page

The vdf utility

The `vdf` utility displays disk usage for domains and filesets. It reformats output from the `showfdmn`, `showfsets`, and `df` commands to display information about the disk usage of AdvFS domains and filesets. It clarifies the relationship between a domain's disk usage and its fileset's disk usage.

Because the values displayed by `vdf` are gathered by executing a sequence of different system utilities, a busy system (files being added, removed, modified, and so on) can produce inconsistent `vdf` output.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Displaying Used and Available Disk Space for Domains and Filesets"
Reference Pages: `vdf(8)`

Top of Page

Metadata display utilities

There are a number of disk structure dumping utilities that enable you to examine metadata in a domain. The commands can save or display the formatted contents of a metadata file. The `savemeta` command is used to save all the metadata information from a domain in a named directory for later examination by the other utilities.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
Reference Pages: `savemeta(8)`

Top of Page

Advanced File System Quick Reference

Planning:New Features in Version 5.1A Operating System Software

The following features have been added to AdvFS:

- The `fixfdmn` command
- Temporary atomic-write data logging
- Object safety (object reuse) protection
- Frag file on/off

The `fixfdmn` command

The `fixfdmn` utility puts a domain into a usable (mountable) state. It scans on-disk metadata looking for corruptions and either corrects them or deletes the corrupt metadata and associated files. See Running `fixfdmn` in Troubleshooting:Recovery Procedures.

The `fixfdmn` utility is available in patch kits for earlier versions of the operating system. See your customer service representative if you are unable to access it in this manner.

See also:
For your version of the operating system, see:

Version 5.1A Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Fixing On-Disk Metadata Corruptions"
Reference Pages: `fixfdmn(8)`

Top of Page

Temporary atomic-write data logging

Atomic-write data logging (see Data consistency in Planning:Configuration and Layout) can be activated on per file basis for the duration of the fileset mount. All applications that have the file open are affected. However, persistent atomic-write data logging takes precedence over temporary atomic write data logging while the file is open.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Temporary Atomic-Write Data Logging"
Reference Pages: `mount(8)`

Top of Page

Object safety (object reuse) protection

When the object safety option is enabled with the `chfsets` command, the pages on disk belonging to the fileset are zero-filled and forced to disk before they are available to a file. This prevents old data from being visible if a system crash occurs while the file is being written.

Because object safety degrades performance (the pages on disk are zero-filled, then the zeroes are overwritten with data), performance impact must be measured against the improved security.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Changing the Characteristics of a Fileset"
Reference Pages: `chfsets(8)`

Top of Page

Frag file on/off

Files or parts of files that are less than 8 kilobytes are stored in the frag file for the fileset. Fragging minimizes wasted space in the fileset. If fragging is turned off, I/O is more efficient, but storage requirements increase. See Disabling frag files in Upkeep:I/O Tradeoffs.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Improving Performance by Disabling the Frag File"
Reference Pages: `chfsets(8)` and `mkfset(8)`

Top of Page

Advanced File System Quick Reference

Setup

After you have planned your system, choose the types of volumes you want, create domains and filesets on these volumes, then set quotas to limit usage.

Volumes

- License registration
- Direct-connect disks
- Logical Storage Manager (LSM)
- RAID

Domains

- Creating a domain
- Upgrading a domain to DVN4
- Adding volumes to a domain
- Configuring the `root` domain as AdvFS
- Converting a UFS file system to AdvFS
- Domain bookkeeping

Filesets

Quotas

Using a Graphical User Interface (GUI)

See also: Troubleshooting:Avoiding Problems

Advanced File System Quick Reference

Setup:Volumes

An AdvFS volume is anything that behaves like a UNIX block device. It can be a disk or disk partition. AdvFS can function with Logical Storage Manager (LSM) or redundant array of independent disks (RAID) volumes as well. Both AdvFS and LSM have expanded volume functionality obtainable only by license.

License registration

AdvFS is the default file system for Tru64 UNIX. Some functionality requires a separately licensed product (AdvFS Utilities). These utilities provide additional processing capabilities including the option to create domains containing more than one volume. LSM, which is also licensed, includes software RAID capabilities. Both AdvFS Utilities and LSM are included in the Compaq StorageWorks™ package. You must register a license product authorization key (PAK) before you can use them. Contact your software support organization for additional information.

Top of Page

Direct-connect disks

You can set up local system disks and run your system without LSM volumes or RAID.

Top of Page

Logical Storage Manager

AdvFS treats LSM volumes just like any other volume such as a disk or a disk partition. You can use LSM to create and manage the volumes that you use in AdvFS domains.

To use LSM with AdvFS:

1. Create the LSM volumes with the desired attributes.
2. Configure AdvFS using LSM volumes.

This configuration gives you the RAID capabilities of the LSM software. AdvFS does not require that all domains use LSM volumes or that domains use only LSM volumes. However, software RAID is available only to those domains that do.

DO NOT increase the size of a domain by increasing the size of a volume with an LSM command. AdvFS does not handle this reliably.

LSM and AdvFS together provide some performance enhancements:

- Mirroring with LSM volumes and/or RAID allows access to the same information on more than one volume. This provides failover in the case of a disk failure.
- Mirroring can improve the read throughput because files can be accessed from either volume depending upon I/O load.
- Detailed information on disk I/O activity is available with LSM disk monitoring.

- LSM's volume striping is useful when large files are shared and when the transaction log is spread over multiple disks. AdvFS stripes individual files.

DO NOT use AdvFS striping with any other form of striping. This might degrade system performance.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
Logical Storage Manager

Top of Page

RAID

You can use RAID technology in a storage configuration for high performance and high data availability. You can configure RAID functionality by using LSM or a hardware-based RAID subsystem.

If you choose hardware RAID, see your appropriate vendor manuals.

To set up software RAID configurations, first configure LSM volumes with RAID. Create domains using these volumes.

You can add an LSM volume with the `addvol` command (requires AdvFS Utilities).

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
Logical Storage Manager
System Configuration and Tuning

Top of Page

Advanced File System Quick Reference

Setup: Domains

Creating a domain is the first step in setting up an AdvFS file system.

- Creating a domain
- Upgrading a domain to DVN4
- Adding volumes to a domain
- Configuring the `root` domain as AdvFS
- Converting a UFS file system to AdvFS
- Domain bookkeeping

Creating a domain

See Using SysMan for system management in Planning:Configuration and Layout, or use the `mkfdmn` command to create a domain and to assign an initial volume to it. A domain is not a complete file system that you can mount. In order to mount an AdvFS file system, the domain must contain one or more filesets. You can access files as soon as you mount one or more filesets. See Setup:Filesets.

The `/etc/fdmns` directory is created and maintained automatically by AdvFS. It defines the domains by providing a subdirectory for each domain you create. The subdirectories contain symbolic links to every volume in the domain (see Domain bookkeeping).

Version 4.x: Domains containing a large number of files created on versions prior to Version 4.0D (and very rarely on later Version 4 operating systems) might have inadequate metadata allocation. This causes an "Out of disk space" error message although there is disk space available. To avoid this, preallocate metadata with the `-x` and `-p` options of the `mkfdmn` command.

Version 5.x: By default, Version 5.0 and later operating systems have a new on-disk format, so adequate metadata allocation is not a problem. Version 5 disk formats are marked with a Domain Version Number (DVN) of 4. Version 5 operating systems recognize the Version 4 operating system on-disk format (DVN3), but the older operating systems do not recognize the DVN4 domains.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"File Domains"
System Configuration and Tuning
Reference Pages: `mkfdmn(8)` and `sysman(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Domains"
System Configuration and Tuning
Reference Pages: `mkfdmn(8)` and `sysman(8)`

SysMan "Create a New AdvFS Domain"

Top of Page

Upgrading a domain to DVN4

Version 5.0 and later operating system domains have a new metadata structure (see Improved on-disk structure in Planning:New Features in Version 5 Operating System Software). If you are running Tru64 Version 5.0 or later, it might be useful to upgrade your DVN3 domains. If you are getting "out of disk space" error messages when the domain still appears to have space available, or if you are running an application that requires quota limits larger than 2 terabytes or that uses directories containing thousands of files, you can improve performance by upgrading your old file domain to a domain with a DVN of 4.

If you perform an update installation (not a full installation) to the Version 5 operating system, all domains transferred in the installation process remain DVN3. Only newly created domains are DVN4.

If you perform a full installation, the new system domains (root, usr, var) are DVN4. Domains transferred from earlier version of the operating system remain DVN3 until you upgrade them.

To upgrade a domain to DVN4 if you have enough space for both the old and new domains:

1. Create a new, uniquely named domain with the `mkfdmn` command. The new domain is DVN4.
2. Create new filesets with the `mkfset` command. These can have the same names as the old.
3. Pipe the output of the `vdump` command on the filesets in the old domain to the `vrestore` command for the filesets in the new domain.
4. Remove the old domain with the `rmfdmn` command. This also removes the old filesets.
5. Rename the directory that has the same name as the new domain to the name of the old domain.

```
# cd /etc/fdmns
# mv new_domain_name old_domain_name
```

To upgrade a domain to DVN4 if you do not have enough space hold both the old and new domains:

1. Back up the filesets in the domain with the `vdump` command. It is a good idea to use the `-x` option for additional protection from saveset errors. You lose all the data in your domain if you cannot restore it after creating the new domain.
2. Remove the old domain with the `rmfdmn` command. This also removes the old filesets.
3. Create a new domain with the `mkfdmn` command. The new domain is DVN4.
4. Create the new filesets with the `mkfset` command.
5. Restore the filesets in the new domain with the `vrestore` command.

See also:
For your version of the operating system, see:

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Upgrading a Domain to DVN4"

Top of Page

Adding volumes to a domain

If you have AdvFS Utilities (see License registration in Setup:Volumes), you can transform a single-volume domain into a multivolume domain by adding one or more volumes (that is, disks, disk partitions, or logical volumes) with the `addvol` command. Unless you are running a cluster configuration, the `root` domain is restricted to a single volume.

Create a multivolume domain to increase the disk capacity of an existing domain. Adding volumes to a domain does not affect the logical structure of the filesets within the domain. You can add volumes to a domain without reconfiguring the

directory hierarchy layer of the file system.

You can add volumes immediately after creating a domain, or you can wait until the domain requires additional space. You can add a volume to a domain while its filesets are mounted and in use.

DO add one partition (typically, partition `c`) to a domain rather than several partitions on one disk (such as `a`, `b`, `g`, `h`).

To correctly configure the volume to be added, see Reusing a volume in Upkeep:Domain Upkeep.

DO NOT add a volume containing data that you want to keep. The old file system is partially overwritten during the `addvol` procedure. Although the previous information becomes inaccessible, it might be possible to recover some of the data with the `salvage` utility.

It is a good idea to limit the number of volumes you add unless you are running a mirror or parity configuration. Disk errors can cause the entire domain to become inaccessible.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"License Registration" and "Adding Volumes"

Version 4.x: Advanced Products CD contains the reference page for the AdvFS Utilities `addvol(8)` command.

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"License Registration" and "Increasing the Size of a Domain"
Reference Pages: `addvol(8)`

TruCluster documentation

SysMan "Manage an AdvFS Domain" utility

Top of Page

Configuring the `root` domain as AdvFS

During initial installation of Tru64 UNIX, you can install `root` on AdvFS. If you do not, you can later convert the `root` file system to AdvFS in order to boot your system from an AdvFS domain.

Unless you are running a cluster configuration, the AdvFS `root` domain must reside on a single disk. Check the size of the existing UFS `root` partition. The target AdvFS `root` domain partition must be at least 5% larger than the UFS partition.

You must have root user privilege and AdvFS installed on your system to convert the `root` domain.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Converting the root File System"

Reference Pages: `disklabel(8)`, `mkdir(1)`, `mkfdmn(8)`, `mkfset(8)`, `mount(8)`, and `vdump(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Converting the root File System"
Reference Pages: `disklabel(8)`, `mkdir(1)`, `mkfdmn(8)`, `mkfset(8)`, `mount(8)`, and `vdump(8)`

TruCluster documentation

Top of Page

Converting a UFS file system to AdvFS

Converting a UFS file system to AdvFS reduces the amount of time your system is down after a system failure because you save the time it would take to run the `fsck` utility. You also acquire all the advantages of a log-based file system.

Back up the existing file system to tape or to a file and restore it to an AdvFS environment. The converted system requires about 5% more disk space.

By converting the `root` file system to AdvFS, you can boot your system from an AdvFS domain. The `root` domain must reside on a single disk unless you are running in a cluster configuration.

Once you have converted to AdvFS, you can modify your file-system configuration by updating your domains (see Upgrading a domain to DVN4) to meet changing system requirements.

You must have root user privilege to perform the conversion.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Converting File Systems"

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Converting File Systems"

TruCluster documentation

Top of Page

Domain bookkeeping

To recognize a domain, the system must have a directory entry in `/etc/fdmns` that defines the name of the domain. The directory contains symbolic links to the physical devices associated with the domain.

Normally the directory entry is created when you enter the command (`mkfdmn`) to make the domain. In situations where you add a domain without using the standard AdvFS commands, for example when physically moving disks containing a domain from a damaged machine, you must update the `/etc/fdmns` directory by hand.

If you have a current copy of the directory, it is preferable to use it. To manually update domain bookkeeping:

1. Create an entry for the domain in `/etc/fdmns` with the `mkdir` command.
2. Use the `ln` command with the `-s` option to create symbolic links for all volumes belonging to the domain.

3. To enable automatic mounting of the filesets at reboot, add domain and fileset information to the `/etc/fstab` file.

You can reconstruct the `/etc/fdmns` directory with the `advscan` utility. The utility can:

- Determine if a partition is an AdvFS partition.
- List partitions in the order they are found on disk.
- Read the disk label to determine which partitions are in the domain and if any are overlapping.
- Scan all disks found in any `/etc/fdmns` directory.
- Recreate missing domain directories. The domain name is created from the device name. You can then rename the domain to the original name.
- Fix the domain count and links for a domain.

Note that this procedure does not create a new domain. It only enables the system to recognize an existing domain whose bookkeeping has been altered.

DO NOT run the `mkfdmn` command to reconstruct an existing domain. Doing so destroys the files in your domain. If you have destroyed your files, you might be able to recover some files by running the `salvage` utility with the `-S` option. See Salvaging a damaged domain in Troubleshooting:Recovery Procedures.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Restoring the /etc/fdmns Directory"
Reference Pages: `advscan(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Restoring the /etc/fdmns Directory"
Reference Pages: `advscan(8)` Top of Page

Advanced File System Quick Reference

Setup:Filesets

A fileset is the mountable portion of the directory hierarchy of a file system. Each fileset is a collection of directories and files that form a subtree structure.

To create a fileset, see Using SysMan for system management in Setup:Configuration and Layout, or:

1. Create a domain with the `mkfdmn` command.
2. Create a fileset with the `mkfset` command.
3. Create a mount-point directory with the `mkdir` command.
4. To activate the domain, mount the fileset with the `mount` command.

A domain must contain at least one fileset to be mountable. A single fileset can consume all of the storage available in the domain, or you can create multiple filesets within a domain that share the storage pool established for the domain (see Domain and Fileset Organization in Planning:Configuration and Layout).

In a domain each fileset can be mounted and unmounted independently. You can limit fileset growth within a domain by assigning fileset quotas (see Setup:Quotas).

You can add filesets to the `/etc/fstab` file in the same manner that you add any file system. Once added, filesets are mounted each time you reboot the system.

- - - - - - - - - -

For example, to add the `sales_dmn` domain and the `nwsales_fs` fileset with quotas enabled, enter:

sales_dmn#nwsales_fs /nwsales advfs rw,userquota,groupquota 0 2

- - - - - - - - - -

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Filesets"
Reference Pages: `mkfset(8)` and `sysman(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Filesets"
Reference Pages: `mkfset(8)` and `sysman(8)`

SysMan

- "Create a New AdvFS Domain"
- "Create a New AdvFS Fileset"
- "Mount File System"
- "Dismount a File System"

Top of Page

Advanced File System Quick Reference

Setup:Quotas

Quotas limit the amount of disk space, that is, the number of files and the number of blocks, that users can access. You can set two types of quotas:

- User and group quotas (similar to UFS quotas)
- Fileset quotas (to restrict the space that a fileset can use)

You can also set two types of quota limits: hard limits cannot be exceeded and soft limits can be exceeded for a period of time called the grace period. If you set no grace period, the grace period remains at the AdvFS default of 7 days.

See the TruCluster documentation if you are using quotas in a clustered environment.

Comparing quota types

	User and group	Fileset
Quota files	In files named `quota.user` and `quota.group` in the `root` directory. (1)	Part of fileset metadata, not visible
Setting quota limits	1. Change the mount options in `/etc/fstab` file to **rw,userquota,groupquota** 2. Mount the fileset. 3. Run the `edquota` command with the `-u` or `-g` option. (2)	Run the `chfsets` command.
Setting grace period	Use the `edquota` command with the `-ut` or `-gt` option.	Same as group grace period
Enabling quotas	- Run the `quotaon` command. - For quota enforcement at reboot, use the `rcmgr` command to add `QUOTA_CONFIG="yes"` to the `/etc/rc.config.common` file.	Setting quota limits turns them on.
Disabling quotas	- Use the `quotaoff` command. This turns quotas off until next `quotaon` command. - Remove `QUOTA_CONFIG="yes"` from the `/etc/rc.config.common` file.	Use the `chfsets` command to set quota limits to 0.
Checking quota limits	Use the `edquota` command with `-u` or `-g` options.	Use the `chfsets` or `showfsets` command.

(1) These files are created when the fileset is created. They are indexed by user ID and group ID, respectively. You cannot delete quota files, but you are not required to enforce quotas. To examine quota files, see Checking fileset and domain disk space in Upkeep:Domain Upkeep.

(2) You can set a separate quota for each user or each group of users for each fileset.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Managing Quotas"
Reference Pages: `chfsets(8)`, `edquota(8)`, `quotaon(8)`, `quotaoff(8)` and `showfsets(8)`

Version 5.x Tru64 UNIX publications page

System and Network Management Documentation Bookshelf:
AdvFS Administration
"Managing Quotas"
Reference Pages: `chfsets(8)`, `edquota(8)`, `quotaon(8)`, `quotaoff(8)` and `showfsets(8)` Top of Page

Advanced File System Quick Reference

Setup:Using a Graphical User Interface (GUI)

If you have AdvFS Utilities (see License registration in Setup:Volumes), you can build and manage your domains with *dtadvfs*, the AdvFS GUI or Version 5.x: the Logical Storage Manager Storage Administrator (LSMSA).

To run the AdvFS GUI, the agent, *advfsd*, must be running. To start the agent and GUI, enter:

```
# /sbin/init.d/advfsd start
# /usr/bin/X11/dtadvfs
```

The Version 4.x agent might slow performance because it polls all domains periodically. The Version 5.x agent operates on demand. You can turn off the AdvFS agent, and turn it on again when you want to run the GUI.

To stop the agent, enter

```
# /sbin/init.d/advfsd stop
```

You can also use SysMan to accomplish many of the tasks facilitated by the graphical interface.

See also:
For your version of the operating system, see:

Version 4.x: Advanced Products CD contains the reference pages for the AdvFS Utilities `advfsd(8)` and `dtadvfs` commands.

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
- *AdvFS Administration*
  - "Managing the Advanced File System with the AdvFS GUI"
- *Logical Storage Manager*

Reference Pages: `advfsd(8)`, `dtadvfs(8)`, `lsmsa(8)`, and `lsmsad(8)`

Top of Page

Advanced File System Quick Reference

Upkeep

Regular maintenance is vital to running an efficient and reliable system.

Domain Upkeep

- Checking fileset and domain disk space
- Verifying domain consistency
- Moving the disks containing a domain to different hardware
- Reusing a volume
- Moving filesets from one domain to another
- Replacing a volume in a domain
- Increasing the size of the root domain

Backup

- The vdump and vrestore commands
- Cloning
- Planning for backup

Monitoring Performance

- Collect
- The vmstat command
- The advfsstat command
- The iostat command

I/O Tradeoffs

- Examining I/O
- Adding faster devices
- Distributing files with high I/O requirements
- Moving the transaction log
- Balancing the domain
- Reducing file fragmentation
- Creating more smaller domains
- Changing I/O mode
- Version 5.1A: Object safety (object reuse) protection
- Version 5.1A: Disabling frag files

See also: Troubleshooting

Advanced File System Quick Reference

Upkeep:Domain Upkeep

There are a number of domain upkeep procedures.

Checking fileset and domain disk space

There a number of ways to check the space used by your domain. For the most accurate results, be sure that your system is quiescent or at least has low system activity. If you do not, you may get inaccurate or out-of-sync information.

Checking space

	Description	Notes
df	Displays the available disk space and the disk space used for each fileset in the domain. (1) (2) (4) (5)	In domains with multiple filesets and no fileset quotas, the total capacity of all filesets can be greater than 100%. The available space value used in the calculation is all the space in the domain because it is available to each fileset.
du	Displays block allocation for named files or directories. (1) (4)	Use the `-a` option to display information about individual files.
ls	Displays the space used by the files. (4)	Use the `-l` option to show the space spanned by the file. For sparse files such as quota files, use the `-s` option to obtain the actual on-disk storage for the file.
showfdmn	Displays the attributes of each volume in the domain. (3) (4)	To display information about all the domains on a system: **# cd /etc/fdmns** **# showfdmn ***
showfile	Displays block usage and volume information for a file or for the contents of a directory.	Use the `-x` option to display the extent map for the file.
showfsets	Displays information about the filesets in a domain, including number of files, number of blocks, fileset clone, if present, object safety, fragging status, and fileset quota limits. (4) (5)	To examine user and group quota limits, run the `edquota` command with the `-u` or `-g` options.
Version 5.x **vdf**	Displays information about the disk usage and the sizes of metadata files in a domain or fileset. (1) (4) (5)	Reformats output from the `showfdmn`, `showfsets`, and `df` commands.

(1) For more accurate results, run the `quotacheck` command first. It is particularly important that you do this on a quiescent system. If you do not have user and group quotas enabled, do so temporarily (see Quotas in Setup:Quotas).

(2) Disk space usage is calculated somewhat differently from calculations for traditional UNIX file systems:

- If a fileset quota has been set, the command displays the amount of space available for the fileset.
- When both soft and hard quota limits are set, the command calculates the disk space available using the smaller of the two limits.
- If there is less space in the domain than is allowed by the fileset quota, the command displays the actual space available in the domain.
- If fileset quotas have not been established, the command displays the domain size in the Available field.

(3) If your domain is low on space (more than 85-90% in use), consider adding another volume to the domain (see Adding volumes to a domain in Setup:Domains).

(4) Use the `-k` option to display in kilobytes.

(5) The `df` and `showfsets` commands do not include the space used for metadata and quota files, so the value in the Used field may be smaller than that actually used by the domain. The `vdf` command displays disk usage that includes metadata and quota files.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Managing Quotas"
Reference Pages: `df(1)`, `du(1)`, `ls(1)`, `showfdmn(8)`, `showfile(8)`, and `showfsets(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Managing Quotas"
Reference Pages: `df(1)`, `du(1)`, `ls(1)`, `showfdmn(8)`, `showfile(8)`, `showfsets(8)`, and `vdf(8)` Top of Page

Verifying domain consistency

The `verify` command checks the AdvFS metadata structure. It is a good idea to run this command:

- When problems are evident (corruptions, domain panic, lost data, I/O errors).
- When an event has occurred that you feel has put media at risk (such as a power spike).

Consider running the command:

- After a system crash.
- Before an update installation.
- If your files have not been accessed in three to six months or longer and you plan to run utilities such as `balance`, `defragment`, `migrate`, `quotacheck`, `repquota`, `rmfset`, `rmvol`, or `vdump` that access every file in a domain.

Run the `verify` command with the `-f` option and unmount all filesets in the domain. This option attempts to repair certain types of metadata inconsistencies and allows the most complete recovery. If you want to, run the command again without the `-f` option to ensure your domain is consistent. Version 5.x: The `verify` command with the `-a` option performs some, but not all, of the checks on mounted domains.

If you mirror your AdvFS domains, you can run the `verify` command on the split mirror when it is necessary to validate an AdvFS domain. You do not have to take your system down, and there is only a minor performance hit while the mirror resynchronizes after the command completes. This is a far less intrusive procedure than taking the whole domain offline to run the utility. See *Logical Storage Manager*, "Removing a Plex" and "Reattaching a Plex."

If `verify` finds problems:

1. Fix the problems on the split mirror.
2. Run an incremental backup on the primary domain from the time the mirror was split off.
3. Restore the incremental backup to the fixed domain.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Verifying File System Consistency"
Reference Pages: `verify(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Verifying File System Consistency"
Reference Pages: `verify(8)` Top of Page

Moving the disks containing a domain to different hardware

To recognize a domain, the system must have a directory entry in `/etc/fdmns` that defines the name of the domain. The directory contains symbolic links to the physical devices associated with the domain. Moving an existing domain containing one or more volumes to a different physical machine requires that you create a directory and symbolic links to the volumes.

To move a domain to a new system:

1. Create a directory entry for the new domain in the `/etc/fdmns` directory on the new system.
2. Create a symbolic link in the directory for each volume in the domain.

For more information see Domain bookkeeping in Setup:Domains.

DO NOT run the `mkfdmn` command. Doing so destroys the files in your domain. If you have destroyed your files, you might be able to recover some files by running the `salvage` utility with the `-S` option. See Salvaging a damaged domain in Troubleshooting:Recovery Procedures.

Top of Page

Reusing a volume

You can only reuse a volume marked "unused." If your volume is part of an existing domain, remove it from the old domain with the `rmvol` command. This marks the volume "unused." You can also mark a volume "unused" with the `disklabel` command. Then add the volume to the new domain with the `addvol` command.

All information that exists on the volume you are adding becomes unavailable once you run the `addvol` command. If you want to retain information and move the volume, see Moving the disks containing a domain to different hardware.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration

"Filesets"
Reference Pages: `mkfdmn(8)`, `mkfset(8)`, `rmfset(8)`, `vdump(8)`, and `vrestore(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Filesets"
Reference Pages: `mkfdmn(8)`, `mkfset(8)`, `rmfset(8)`, `vdump(8)`, and `vrestore(8)` Top of Page

Moving filesets from one domain to another

If you have a domain with many filesets, you can create another domain and move some of the filesets to it. For the pros and cons see Domain and fileset organization in Planning:Configuration and Layout.

To move an existing fileset to a new domain:

1. Create the new domain, if needed, with the `mkfdmn` command.
2. Create a new fileset with the same name in the new domain with the `mkfset` command. Create a mount-point directory for the fileset with the `mkdir` command.
3. Mount the new fileset.
4. Create an entry in the `/etc/fstab` file for the new domain and fileset. See Setup:Filesets if you want the fileset to be mounted at reboot.
5. Move the contents of the fileset to the new fileset using the `vdump` and `vrestore` commands.
6. Delete the fileset from the old domain with the `rmfset` command.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Filesets"
Reference Pages: `mkfdmn(8)`, `mkfset(8)`, `rmfset(8)`, `vdump(8)`, and `vrestore(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Filesets"
Reference Pages: `mkfdmn(8)`, `mkfset(8)`, `rmfset(8)`, `vdump(8)`, and `vrestore(8)` Top of Page

Replacing a volume in a domain

You can replace a volume in a domain without taking the domain offline:

1. You might first want to run the `verify` command (see Verifying domain consistency).
2. Add the new volume with the `addvol` command.
3. Remove the old volume with the `rmvol` command.
4. If you are concerned that the domain is balanced, use the `showfdmn` command on the domain to determine if the percentage of used space on each volume is acceptable. If the domain is not well distributed, use the `balance` command to distribute the files.

See also:

For your version of the operating system, see:

Version 4.x: Advanced Products CD contains the reference pages for the AdvFS Utilities `addvol(8)`, `balance(8)`, and `rmvol(8)` commands.

Version 5.x Tru64 UNIX publications page
Reference Pages: `addvol(8)`, `balance(8)`, and `rmvol(8)`

Top of Page

Increasing the size of the root domain

The AdvFS root domain is limited to one volume unless you are running in a cluster configuration. If you want to increase the size of the root domain, you must recreate the domain on a larger volume. The easiest way to do this is to recreate the root domain on a different device.

You need the following:

- Current operating system CD-ROM or access to it from a Remote Installation Service (RIS) Server
- Backup device
- Information about console commands

To move root to a larger volume:

1. Install a new device for the domain.
2. Configure the volume for use as the root domain (use `mkfdmn -r`).
3. Backup the current root domain.
4. Recreate the root domain on the new volume.
5. Edit the `/etc/fstab` file to specify mounting the new `root_domain#root` on `/`

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Increasing the Size of an AdvFS root Domain"
System Administration
"Restoring or Duplicating a System (Root) Disk"

TruCluster documentation

Top of Page

Advanced File System Quick Reference

Upkeep:Backup

AdvFS provides extended file-system backup capabilities with the `vdump` and `vrestore` commands and Version 5.x: remote backup with the `rvdump` and `rvrestore` commands. On active domains you can use the `clonefset` utility to make a static copy of a fileset, which you can then back up.

You can also back up your files with the `dump`, `cpio`, `tar`, and `nsr` commands. The tools you use to back up and to restore files must be compatible.

The vdump and vrestore commands

The `vdump` (Version 5.x also `rvdump`) and `vrestore` (Version 5.x also `rvrestore`) commands can be used to back up not only AdvFS filesets but also UFS and other standard file systems.

If you use the `vdump` command to back up a file system, you must use the `vrestore` command to restore saved files.

With the `vdump` command you can:

- Save mounted filesets.
- Choose the subdirectory that you want to back up. You do not need to dump an entire fileset.
- Compress files to minimize the saveset size.
- Specify the number of in-memory buffers. You can maximize throughput by choosing a number compatible with your storage device. Specify the same number when you run the `vrestore` command.
- Limit your display to error messages. You do not need to display warning messages.
- Display the names of files as they are backed up.
- Configure output with an error-protection system that allows you to recover data even if there is a read error when you restore.

You do not have to be root user to use the `vdump` and `vrestore` commands, but you must have write permission for the directory to which you want to restore files. AdvFS quota files and the fileset quotas for the fileset can be saved and restored only when the root user initiates the command.

Be aware that when you restore a fileset with an active quota, that quota remains with the fileset. If you restore to a volume of a different size, use the `showfsets` command to check that your quota is the size you want it to be.

DO NOT back up a fileset to an output file in the same fileset.

The `vdump` and `vrestore` dump file formats are forward compatible in Version 4 and Version 5 operating systems.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Backing Up and Restoring"

Reference Pages: `dump(8)`, `restore(8)`, `vdump(8)`, and `vrestore(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Backing Up and Restoring"
Reference Pages: `dump(8)`, `restore(8)`, `vdump(8)`, and `vrestore(8)` Top of Page

Cloning

A clone fileset is a read-only snapshot of the data in an existing fileset. You create a clone with the `clonefset` utility to capture the fileset data at a particular time. Until you delete the clone, as you modify the data in your original files, AdvFS saves the data that existed in the original, page by page, into the clone.

You must license AdvFS Utilities (see License Registration in Setup:Volumes) to create a clone fileset. You can have only one clone per fileset.

Backing up a database with a clone
If your database has an online backup utility, use it to back up the database. If it does not, you can back up the database with a clone:

1. Shut down the database so that all database buffers are flushed and the fileset has a complete, consistent copy of the database files.
2. Clone the fileset and mount the clone.
3. Reactivate the database.
4. Back up the clone fileset.
5. Unmount and delete the clone.

DO NOT use any tools except the database's own utilities to back up an active database. You can use the `vdump` and `vrestore` commands on a clone fileset that contains the database.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Cloning for Online Backup"

Version 4.x: Advanced Products CD contains the reference page for the AdvFS Utilities `clonefset(8)` command.

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Cloning for Online Backup"
Reference Pages: `clonefset(8)`

Top of Page

Planning for backup

Schedule incremental or full backups depending upon your system requirements. See the `vdump(8)` reference page for an example of a backup schedule. This schedule alternates tapes so there are two copies of every backup in case of tape failure.

Full backups save all files in the fileset. Incremental backups capture only the changes that have occurred since a previous, lower-numbered backup.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `vdump(8)` and `vrestore(8)`

Top of Page

Advanced File System Quick Reference

Upkeep:Monitoring Performance

There are a number of ways to monitor system performance:

- Collect
- The `vmstat` command
- The `advfsstat` command
- The `iostat` command

Collect

The Collect utility gathers historical operating system and process data. This is different from operating system utilities that provide a snapshot of the system at a given time. Any subset of the subsystems (Process, Memory, Disk, Tape, LSM Volumes, Network, CPU, File systems, Message Queue, tty, and Header) can be defined as a source of collected information.

Collect has two modes: collection and playback. In collection mode, it gathers operating system and process data and writes it to standard output, to a binary file with a special format, or to both. In playback mode, Collect reads from a previously written binary file and writes to standard output.

Collect is available on all currently supported operating systems. For Version 4.0F see patch kit 3. Beginning with Version 4.0G, Collect is included with Tru64 UNIX.

See the Collect web page for more information.

Top of Page

The `vmstat` command

The `vmstat` utility displays virtual memory statistics. You can specify an interval for your display and the number of reports to be delivered.

The command is useful for obtaining process information and paging statistics. Check *System Configuration and Tuning* for the parameters you might want to modify.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
System Configuration and Tuning
"Monitoring Memory by Using the `vmstat` Command"
Reference Pages: `vmstat(8)`

Top of Page

The `advfsstat` command

The advfsstat utility command displays AdvFS domain and fileset performance statistics. You can specify an interval for your display and the number of reports to be delivered.

The command captures statistics for the buffer cache, for volumes, for the BMT and more. Use this information to determine if it would be beneficial to change any of the parameters that *System and Configuration Tuning* mentions as tunable.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
System Configuration and Tuning
"Monitoring AdvFS Performance Statistics by Using the `advfsstat` Command"
"AdvFS I/O Queues"
Reference Pages: `advfsstat(8)`

Top of Page

The `iostat` command

The `iostat` utility reports I/O statistics. You can track the number of characters written and read by terminals, the number of transfers and bytes transferred by a disk, and the percentage of time the system has spent in user or system mode or spent idling.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
System Configuration and Tuning
"Displaying Disk Usage by Using the `iostat` Command"
Reference Pages: `iostat(8)`

Top of Page

Advanced File System Quick Reference

Upkeep:I/O Tradeoffs

This discussion describes a single-node system. If you are running in a cluster, conditions might vary (see TruCluster documentation).

You can improve throughput if you spread your I/O queues over more than one device, employ faster devices, and have more efficient file storage.

Consider doing one or more of the following:

- Examining I/O
- Adding faster devices
- Distributing files with high I/O requirements
- Moving the transaction log
- Balancing a domain
- Reducing file fragmentation
- Creating more smaller domains
- Changing I/O mode
- Version 5.1A: Object safety (object reuse) protection
- Version 5.1A: Disabling frag files

Examining I/O

If you determine where your I/O bottleneck is occurring, you can place a faster device at that location, move files with high I/O requirements to another volume, and/or move the transaction log to another volume. There are a number of ways to check disk activity:

- The `iostat` utility reports I/O statistics for terminals, disks, and the CPU. It displays the number of transfers per second (tps) and bytes per second in kilobytes (bps). From this you can determine where I/O bottlenecks are occurring. That is, if one device shows sustained high throughput, this device is being utilized more than others.
- Fileset and domain activity is monitored by the `advfsstat` utility. It displays detailed information about them when the command is given and over time. From this you can determine if a domain should be allocated to other devices.
- Collect

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `advfsstat(8)` and `iostat(1)`

Top of Page

Adding faster devices

If you have access to faster rotational SCSI drives or solid-state disks, you might want to substitute them for your existing devices.

If you have AdvFS Utilities (see License registration in Setup:Volumes), add the faster device to the domain with the `addvol` command. Then remove the slower device with the `rmvol` command to migrate the files to the faster device.

If you do not have AdvFS Utilities:

1. Make a domain with a new name on the faster device.
2. Create filesets with the same names as the old.
3. Create a temporary mount-point directory for each fileset.
4. Mount each new fileset at the temporary mount point.
5. Use the `vdump` command to copy the filesets from the old device.
6. Restore the domain to the new volume with the `vrestore` command.
7. Unmount the old and new filesets.
8. Rename the new domain to the old, if desired. If not, update the `/etc/fstab` file (see Setup:Filesets).

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
Reference Pages: `vdump(8)` and `vrestore(8)`

Version 4.x: Advanced Products CD contains the reference pages for the AdvFS Utilities `addvol(8)` and `rmvol(8)` commands.

Version 5.x Tru64 UNIX publications page
Reference Pages: `addvol(8)`, `rmvol(8)`, `vdump(8)` and `vrestore(8)`

Top of Page

Distributing files with high I/O requirements

If a file appears to strain system resources, move it to another volume with the `migrate` utility or distribute it across volumes with the `stripe` utility.

Virtual storage solutions, such as Logical Storage Manager (LSM), RAID, and storage area networks (SAN), stripe whole systems and are usually configured at system setup. AdvFS striping is done per file. Turn on AdvFS striping after you create the file but before you write data to it. If you want to stripe an existing file, make a new file and move the data to it.

DO NOT use AdvFS striping on a file when other striping is in effect. This might degrade system performance.

See also:
For your version of the operating system, see:

Version 4.x: Advanced Products CD contains the reference pages for the AdvFS Utilities `migrate(8)` and `stripe(8)` commands.

Version 5.x Tru64 UNIX publications page
Reference Pages: `migrate(8)` and `stripe(8)`

Top of Page

Moving the transaction log

Monitor performance of the volume that contains the transaction log with the `iostat` utility. Do one of the following if the volume containing the log appears overloaded:

- Move the transaction log to a faster or less congested volume.
- Divide the domain into several smaller domains. Then each transaction log handles transactions for fewer filesets.

To move the transaction log to another volume:

1. Use the `showfdmn` command to determine the location of the log. The letter **L** after the volume number indicates the volume on which the log resides.
2. Use the `switchlog` command to move the log to another volume.
3. Enter the `showfdmn` command again to confirm that the log has been moved.

You can also isolate the transaction log on its own volume by doing the following:

1. Add a small partition (volume) to the domain.
2. Move the log to that partition.
3. Create a dummy file to fill the volume ensuring that only log activity takes place on the volume.

Isolate the log when the system is running in single user mode or when there is little system activity.

The following example isolates the transaction log for the domain `usr_domain` mounted at `/usr`. Version 5.x device names are used, but this feature works for Version 4.x using the appropriate device-naming conventions.

```
# addvol /dev/disk/dsk2d usr_domain
# showfdmn usr_domain

      Id              Date Created     LogPgs Version  Domain Name
37a5e009.0004e280 Aug 2 11:14:33 1999    512       4  usr_domain

Vol 512-Blks  Free  % Used Cmode Rblks Wblks Vol Name
1L  6607056 5492816    17%    on   256   256 /dev/disk/dsk0g
2    256304  200384    22%    on   256   256 /dev/disk/dsk2e
3    256304  256192     0%    on   256   256 /dev/disk/dsk2d
  --------- ------- ------
    7119664 5949392    16%

# switchlog usr_domain 3
# showfdmn usr_domain

      Id              Date Created     LogPgs Version Domain Name
37a5e009.0004e280 Aug  2 11:14:33 1999   512       4 usr_domain

Vol 512-Blks  Free  % Used Cmode Rblks Wblks Vol Name
1   6607056 5492816    17%    on   256   256 /dev/disk/dsk0g
2    256304  200384    22%    on   256   256 /dev/disk/dsk2e
3L   256304  248000     3%    on   256   256 /dev/disk/dsk2d
 ---------- ------- ------
    7119664 5941200    17%
```

Allocate all the free blocks on the volume containing the log to a dummy file, `/usr/foo`:

```
# dd if=/dev/zero of=/usr/foo count=248000
248000+0 records in
248000+0 records out
```

Check the dummy file to ascertain that the volume containing the transaction log is full:

```
# showfdmn usr_domain

      Id              Date Created     LogPgs Version Domain Name
37a5e009.0004e280 Aug  2 11:14:33 1999   512       4 usr_domain
```

```
Vol 512-Blks  Free  % Used Cmode Rblks Wblks Vol Name
1   6607056 5445200     18%   on   256   256 /dev/disk/dsk0g
2    256304       0    100%   on   256   256 /dev/disk/dsk2e
3L   256304  248000      3%   on   256   256 /dev/disk/dsk2d
 ---------- -------  ------
    7119664 5693200     20%
```

If it is not, use the `showfile` command with the `-x` option to determine where the dummy file was placed:

```
 # showfile -x /usr/foo

      Id Vol PgSz Pages XtntType Segs SegSz  I/O Perf File
5f45.8002  2   16 15500   simple   **    ** async 97% foo

 extentMap: 1
   pageOff pageCnt vol volBlock  blockCnt
         0     141   2   198240      2256
       141       1   2   200528        16
       142       4   2   200560        64
         .
         .
         .
     12511      13   2   132320       208
     12524     207   1  1367936      3312
         .
         .
         .
     14971      11   1  1407408       176
     14982     518   1  1407600      8288
 extentCnt: 59
```

If necessary, migrate the dummy file to the volume containing the transaction log (volume 3 in this example):

```
# migrate -s 2 -d 3 /usr/foo
# showfdmn usr_domain

      Id              Date Created   LogPgs Version Domain Name
37a5e009.0004e280 Aug  2 11:14:33 1999 512       4 usr_domain

Vol 512-Blks  Free   % Used Cmode Rblks Wblks Vol Name
1    6607056 5445200    18%    on   256   256 /dev/disk/dsk0g
2     256304  200384    22%    on   256   256 /dev/disk/dsk2e
3L    256304   47616    81%    on   256   256 /dev/disk/dsk2d
  ---------- ------- ------
     7119664 5693200    20%

# migrate -s 1 -d 3 /usr/foo
# showfdmn usr_domain

      Id              Date Created   LogPgs Version Domain Name
37a5e009.0004e280 Aug  2 11:14:33 1999 512       4 usr_domain

Vol 512-Blks  Free   % Used Cmode Rblks Wblks Vol Name
1    6607056 5492816    17%    on   256   256 /dev/disk/dsk0g
2     256304  200384    22%    on   256   256 /dev/disk/dsk2e
3L    256304       0   100%    on   256   256 /dev/disk/dsk2d
  ---------- ------- ------
     7119664 5693200    20%

----------
```

See also:

For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page

System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Improving the Transaction Log Performance"
Reference Pages: `iostat(1)`, `showfdmn(8)`, `switchlog(8)`, `vdump(8)`, and `vrestore(8)`

Version 4.x: Advanced Products CD contains the reference pages for the AdvFS Utilities `addvol(8)`, `migrate(8)`, `rmvol(8)` and `stripe(8)` commands.

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Improving the Transaction Log Performance"
Reference Pages: `addvol(8)`, `iostat(1)`, `migrate(8)`, `rmvol(8)`, `showfdmn(8)`, `stripe(8)`, `switchlog(8)`, `vdump(8)`, and `vrestore(8)`

Top of Page

Balancing a domain

The `balance` utility distributes the percentage of used space evenly among volumes in a multivolume domain. This improves performance and evens the distribution of future file allocations.

Files are moved from one volume to another until the percentage of used space on each volume in the domain is as equal as possible. (Because the `balance` utility does not generally split files, domains with very large files might not balance as evenly as domains with smaller files.)

Generally, run the `balance` utility after adding a volume to move files to it. The utility balances space on the volumes, but I/O performance might be changed by the resulting file layout.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Balance Utility"

Version 4.x: Advanced Products CD contains the reference page for the AdvFS Utilities `balance(8)` command.

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Balancing a Multivolume Domain"
Reference Pages: `balance(8)`

Top of Page

Reducing file fragmentation

The `defragment` utility consolidates free space. Defragmenting benefits both read and write processes. A defragmented file can be read more quickly than one that is not. In addition, the defragment process often makes the free space on a disk more contiguous, so files that are created later have more adjacent blocks available to be written to. This then improves the read/write access of the new files.

Run defragment when administratively necessary and then, preferably, when there is low activity. Running the `balance` utility before you run defragment might speed up the defragmentation process. Balancing a domain after defragmenting might undo some defragmentation and free space consolidation.

There is no benefit to running `defragment` on:

- Any system that has a file system comprised mainly of files smaller than 8 kilobytes, such as mail or news servers.
- Any system not experiencing performance-related problems due to excessive file fragmentation. In many cases even a large, fairly fragmented file incurs no noticeable performance penalty.

There is no level of fragmentation for which running the `defragment` utility is required. Defragment only when it is needed for system performance.

Backing up and restoring also decreases fragmentation.

Version 5.0: The `defragment` utility initiates a thread per fileset on the domain being fragmented. If you want to limit defragmenting to a single thread (that is, to mimic the behavior of a Version 4 operating system) use the `-N 1` option.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Defragment Utility"
Reference Pages: `defragment(8)`, `vdump(8)`, and `vrestore(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Defragmenting a Domain"
Reference Pages: `defragment(8)`, `vdump(8)`, and `vrestore(8)` Top of Page

Creating more smaller domains

If you have a single or very few domains with many, many filesets, I/O improves if you create more domains:

1. Create new domains with the `mkfdmn` command and new mount points with the `mkdir` command.
2. Use `mkfset` to make new filesets.
3. Back up the contents of the old domain with the `vdump` command.
4. Restore the files to the new filesets with the `vrestore` command.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Domains"
Reference Pages: `mkfdmn(8)`, `mkfset(8)`, `vdump(8)`, and `vrestore(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Domains"
Reference Pages: `mkfdmn(8)`, `mkfset(8)`, `vdump(8)`, and `vrestore(8)` Top of Page

Changing I/O mode

The way data is cached and when transactions are logged can be altered to improve user data consistency if a crash should occur. The `chfile` command turns on and off forced synchronous I/O and atomic-write data logging. See Data consistency in Planning:Configuration and Layout for a description and comparison of I/O methods.

Top of Page

Version 5.1A: Object safety (object reuse) protection

When the object safety option is enabled with the `chfsets` command, the pages on disk belonging to the fileset are zero-filled and forced to disk before they are available to the file. This prevents old data from being visible if a system crash occurs while the file is being written.

Because object safety degrades performance (the pages on disk are zero-filled then the zeroes are overwritten with data), performance impact must be measured against the improved security.

See also:
Version 5.1A Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Changing the Characteristics of a Fileset"
Reference Pages: `chfsets(8)`

Top of Page

Version 5.1A: Disabling frag files

Files or parts of files that are less than 8 kilobytes are stored in the frag file for the fileset. Fragging minimizes wasted space in the fileset. If fragging is turned off, I/O is more efficient, but storage requirements increase.

Filesets have frags turned on by default. You can disable them:

- At fileset creation by using the `mkfset` command.
- For an existing fileset by using the `chfsets` command.

Disabling or enabling frags in a fileset does not affect existing files unless they are extended or truncated.

See also:
Version 5.1A Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Improving Performance by Disabling the Frag File"
Reference Pages: `chfsets(8)`

Top of Page

Advanced File System Quick Reference

Troubleshooting

The following section addresses problems and recovery procedures. Some failures can be avoided if you are aware of system limitations.

Advanced File System Quick Reference

Troubleshooting:Problems

Back up your data regularly and check the system log files. Identifying a potential problem can prevent a lot of trouble.

Domain panic

When a metadata write error occurs, or if corruption is detected in a single AdvFS domain, the system generally initiates a domain panic (rather than a system panic) on the domain. This isolates the failed domain and allows the system to continue to serve all other domains. After a domain panic, AdvFS no longer issues I/O requests to the affected domain. Although the domain cannot be accessed, the filesets in the domain can be unmounted.

When a domain panic occurs, the following message is printed to the system log and the console:

```
AdvFS Domain Panic; Domain <name> Id <domain_Id>
An AdvFS domain panic has occurred due to either a
   metadata write error or an internal inconsistency.
This domain is being rendered inaccessible.
```

To recover from a domain panic, perform the following steps:

1. Run the `mount` command with the `-t advfs` option to find the mounted filesets in the affected domain.
2. Unmount all these filesets.
3. Examine the `/etc/fdmns` directory to obtain a list of the AdvFS volumes in the domain that panicked.
4. Save metadata (see Saving metadata in Troubleshooting:Recovery Procedures).
5. Check to see if the problem is a hardware problem. If it is, fix the hardware before continuing. To look for hardware problems:

   Version 4.x: Run the `dia(8)` command.

   Version 5.x: Run Compaq Analyze.
6. Run the `verify` command (see Running `verify` in Troubleshooting:Recovery Procedures).
7. Run the `fixfdmn` command (see Running `fixfdmn` in Troubleshooting:Recovery Procedures) to ensure that the

domain metadata is consistent.

8. If the `verify` or `fixfdmn` command reported problems, and if you cannot restore from backup, run the `salvage` command (see Salvaging a damaged domain in Troubleshooting:Recovery Procedures) to recover as much of your domain as possible.
9. Once your domain is correct, remount the filesets.

You do not need to reboot after a domain panic.

Please file a problem report with your software support organization. Include as much information as possible.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
Reference Pages: `mount(8)` and `vfile(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Recovering from a Domain Panic"
Reference Pages: `mount(8)` and `savemeta(8)` Top of Page

Data corruption

If you suspect data corruption in the absence of I/O errors, follow the complete Recovery procedures.

Please file a problem report with your software support organization.

Top of Page

Failure of the `root` domain

A catastrophic failure of the disk containing your AdvFS root domain requires that you create a new `root` domain to boot your system. Before you recreate your domain, check that the failure is not due to a hardware problem.

DO NOT run the `mkfdmn` or `mkfset` command on the remaining volumes. Doing so destroys your metadata. If you have destroyed your files, you might be able to recover some files by running the `salvage` utility with the `-S` option. See Salvaging a damaged domain in Troubleshooting:Recovery Procedures.

For instructions for your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Recovering from Failure of the root Domain"
System Administration
"Restoring the root and /usr File Systems"

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Recovering from Failure of the root Domain"
System Administration
"Restoring or Duplicating a System (Root) Disk"
Top of Page

Loss of a volume in a multivolume domain

If a volume in your domain fails, try to mount the filesets and remove the volume with the `rmvol` utility. If this is not possible, you must repopulate the domain. See Salvaging a damaged domain in Troubleshooting:Recovery Procedures.

DO restore from backup if you have a recent one. This is most reliable.
DO NOT run the `mkfdmn` command on the remaining volumes of the failed domain. Doing so destroys metadata. If you have run the `mkfdmn` command, you might be able to recover some files by running the `salvage` utility with the `-S` option. See Salvaging a damaged domain in Troubleshooting:Recovery Procedures.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Restoring a Multivolume usr Domain"

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Restoring a Multivolume usr Domain" Top of Page

Filesets mounted read-only

Filesets are mounted read-only when an inconsistency is detected in the mount process. There are two common reasons this occurs:

- A disk is mislabeled on a RAID array.
- A Logical Storage Manager (LSM) volume upon which an AdvFS domain resides has been shrunk from its original size.

When a fileset is mounted, AdvFS verifies that all the volumes in a domain can be accessed. If the size recorded in the domain's metadata for each volume matches the size of the volume, the mount proceeds. If a volume is smaller than the recorded size, AdvFS attempts to read the last block marked in use for the volume. If this block can be read, the mount succeeds, but the fileset is marked as read-only. If the last in-use block for any volume in the domain cannot be read, the mount fails.

To recover, if you have AdvFS Utilities, and if the domain consists of multiple volumes and has enough free space to remove the offending volume:

1. Remove the volume from the domain using the `rmvol` command. (This automatically migrates the data to the remaining volumes.)
2. Correct the disk label of the volume with the `disklabel` command. (This step is not necessary if you are using an LSM volume.)
3. Return the corrected volume to the domain with the `addvol` command.
4. Run the `balance` command to distribute the data across the new volumes.

`----------`

Version 4.x: For example:

```
# rmvol /dev/rz2c data5
# disklabel -z rz2
# disklabel -rw rz2 rz29
```

```
# addvol /dev/rz2c data5
# balance data5
```

Version 5.x: For an example, see *AdvFS Administration*, "Recovering from Filesets that are Mounted Read-Only."

If you do not have AdvFS Utilities or you are unable to remove the offending volume (that is, you do not have enough storage available to remove your volume and you do not have other storage that you could temporarily add), you must restore your domain from backup.

1. Back up your domain.
2. Remove the domain with the `rmfdmn` command.
3. Correct the disk label of the volume with the `disklabel` command.
4. Make the new domain.
5. Restore the filesets from the backup.

Version 4.x: For example, if `/dev/rz1c` (an `rz28` disk) containing the `data3` domain is mislabeled:

```
# vdump -D data3
# rmfdmn data3
# disklabel -z rz1
# disklabel -rw rz1 rz28
# mkfdmn /dev/rz1c data3
# addvol /dev/rz5c data3
# mkfset data3 fset3
# mount data3#fset3 /data3
# vrestore -xf - /data3
```

Version 5.x: For an example, see *AdvFS Administration*, "Recovering from Filesets that are Mounted Read-Only."

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
Reference Pages for related reference pages

Version 4.x: Advanced Products CD contains the reference pages for the AdvFS Utilities `addvol(8)` and `rmvol(8)` commands.

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Recovering from Filesets that are Mounted Read-Only"
Reference Pages for related reference pages

Top of Page

"Can't clear a bit twice" error message

If you receive a "Can't clear a bit twice" error message in the system message file or on the console, a domain corruption has caused the system to crash. To repair it:

1. Reboot and check which domain does not mount.

2. Set the AdvfsFixUpSBM kernel variable to allow access to the damaged domain. This option is off by default. To turn AdvfsFixUpSBM on:

```
# dbx -k /vmunix /dev/mem
dbx> assign AdvfsFixUpSBM = 1
dbx> quit
```

3. Mount and back up the filesets in the damaged domain.
4. Unmount the filesets in the corrupted domain.
5. Run the `fixfdmn` command (see Running `fixfdmn` in Troubleshooting:Recovery Procedures) to fix errors.
   - If the domain is intact, remount the filesets.
   - If there are errors, recreate the domain and filesets and restore from the backup.
6. Once you have repaired your domain, turn AdvfsFixUpSBM off:

```
# dbx -k /vmunix /dev/mem
dbx> assign AdvfsFixUpSBM = 0
dbx> quit
```

DO NOT continue to run with the AdvfsFixUpSBM variable on. The variable is global. It will cause other corrupted metadata to be hidden instead of panicking the domain.

Top of Page

Version 4.x: BMT space error message

Domains containing a large number of files created on versions prior to Version 4.0D (and very rarely on later Version 4 operating systems) may have inadequate metadata allocation. This causes ENOSPC errors to be returned to applications even though running the `df` or `showfdmn` command shows there is still space available in the domain. To avoid this for large files, preallocate metadata with the `-x` and `-p` options of the `mkfdmn` command.

Domains created on Version 5.0 and later are not affected. The new file metadata structure takes care of this problem. If you have many DVN3 files (see Creating a Domain in Setup:Domains) that potentially have this problem and you are running Version 5 operating system software, see "Upgrading a domain to DVN4" in Setup:Domains.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Creating a Domain for a Large Number of Files"
Reference Pages: `mkfdmn(8)`

Top of Page

I/O Errors

Check the system error log on a regular basis. Hardware I/O errors on user data are reported as EIO errors to the application, and the application may or may not report this to the user. If the hardware I/O errors affect metadata, they might show up as a domain panic. Follow as much of the Troubleshooting:Recovery procedures as necessary to retrieve your data.

Top of Page

Machine failure

If a machine has failed, it is possible to move disks containing AdvFS domains to another computer running AdvFS. Connect the disk(s) to the new machine and modify the `/etc/fdmns` directory (see Domain bookkeeping in Setup:Domains) so the new system recognizes the transferred volume(s).

DO NOT use the `addvol` command on these volumes to add them to a domain or the `mkfdmn` command to recreate the domain on a new machine. Doing so makes the data on the volumes inaccessible. If you have run the `addvol` or `mkfdmn` command, you might be able to recover some files by running the `salvage` utility with the `-S` option. See Salvaging a damaged domain in Troubleshooting:Recovery Procedures.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Moving an AdvFS Disk to an Undamaged Machine"

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Physically Moving an AdvFS Disk" Top of Page

Accidental use of `mkfdmn` or `addvol`

The `mkfdmn` and `addvol` commands initialize a volume for use by AdvFS. If the volume is already part of an AdvFS domain, using these commands will damage the metadata for the domain that existed on that volume.

DO NOT use the `mkfdmn` or the `addvol` command to repair volumes that you think may have been damaged. If you accidentally use one of these commands on a previously initialized volume, you might be able to recover some files by running the `salvage` utility with the `-S` option. This option checks each page on each volume in the domain and takes a long time on large domains. See Salvaging a damaged domain in Troubleshooting:Recovery Procedures.

The more commands you issue after the `mkfdmn` or `addvol` command, the more you alter the original volume content and the less information you can recover with the salvage operation. If you cannot get a reasonable amount of information from running the `salvage` command, your only option is to restore the data from backup.

In some cases, recovery generates names based on the file's tag number. This usually happens for the `root` directory, because the `mkfdmn` command usually overwrites this directory.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `salvage(8)`

Top of Page

The `rmvol` or `migrate` command won't start

If your source volume is 100% full, it might not be possible to migrate files from the full volume or to remove the volume. The solution is to remove some files from the volume:

1. Use the `showfile` command to find a file on the affected volume.
2. Copy the file. This puts it on another volume because the source volume is full.
3. Delete the original file. This should free up enough space for the `rmvol` or `migrate` command to operate.

You might need to move additional files if deleting the copied file does not free up enough space.

See also:
Version 4.x: Advanced Products CD contains the reference pages for the AdvFS Utilities `migrate(8)` and `rmvol(8)` commands.

Version 5.x: For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `migrate(8)` and `rmvol(8)`

Top of Page

Running out of space during `rmvol`

If you are removing a volume from a domain and get a message that there is insufficient space, your domain is not damaged. Files not moved when `rmvol` stops remain on the old volume.

To recover, delete enough files to give adequate space on the remaining volumes or add another volume and repeat the `rmvol` command.

See also:
Version 4.x: Advanced Products CD contains the reference pages for the AdvFS Utilities `rmvol(8)` command.

Version 5.x: For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `rmvol(8)`

Top of Page

Handling `verify` "Out of Memory"

On machines with many millions of files, sufficient swap space must be allocated for the `verify` command to run to completion. If the amount of memory required by `verify` exceeds the kernel variable `proc/max_per_proc_data_size` process variable, `verify` does not complete. To overcome this problem, allocate up to 10% of the domain size in swap space for running `verify`.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Verifying File System Consistency"
System Configuration and Tuning
"Memory and Swap Space Configuration"
Reference Pages: `verify(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Verifying File System Consistency"
System Configuration and Tuning
"Memory and Swap Space Configuration"
Reference Pages: `verify(8)` Top of Page

Advanced File System Quick Reference

Troubleshooting:Recovery Procedures

The following recovery steps apply to many types of system crashes. All recovery steps might not be necessary. Follow these steps as needed:

Saving metadata

In the event of a domain panic, system crash, or other event that makes your domain inaccessible, it is useful for Technical Support Personnel if you save the domain metadata before proceeding with recovery.

Version 4.x: Use the `vfile` command iteratively to save metadata from an unmounted domain. There is only one root tag file and one transaction log file per domain. You can either use the `showfdmn` command to locate them and use the `0 2` and `0 3` options on that volume only, or you can run all `vfile` command options on all volumes and support personnel can select the appropriate output.

Version 5.x: Run the `savemeta` utility to capture the domain metadata. The utility saves the domain's log file, bitfile metadata table, each volume's storage bitmap, the domain's root tag file, and the fileset tag files.

Version 4.x: For example, to save data from `domain_1` on `/dev/rz1c`:

```
# vfile 0 0 /dev/rz1c > sav_bmt
# vfile 0 1 /dev/rz1c > sav_sbm
# vfile 0 2 /dev/rz1c > sav_tag
# vfile 0 3 /dev/rz1c > sav_log
```

Version 5.x: For example, to save data from `domain_1` on `/dev/rz1c`:

```
# savemeta domain_1 domain_meta_1
```

Pass the `vfile` command or `savemeta` command output and crash data, if possible, to Technical Support.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
Reference Pages: `vfile(8)`
Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:

Reference Pages: `savemeta(8)` Top of Page

Saving undamaged filesets

Backup any filesets that are mounted or can be mounted. You can use this backup, other recent backups, and the output of the recovery utilities (`verify`, `fixfdmn`, and `salvage`) to recreate your domain.

Top of Page

Running `verify`

Unmount any mounted filesets that you need to recreate, and run the `verify` utility with the `-f` option to determine if your fileset structure is intact and to fix some problems. You can run the `verify` command again without the `-f` option to determine if all errors have been eliminated.

- If there are no errors or all errors were fixed, mount the filesets you unmounted and resume normal operations.
- If there were metadata errors that were not fixed by the `verify` command, run the `fixfdmn` command.
- If the `verify` command was unable to run (or the `fixfdmn` command was not successful), you have to rely on backup and/or salvaging data from the damaged domain.

You must be the root user to run the `verify` utility.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `verify(8)`

Top of Page

Running `fixfdmn`

The `fixfdmn` utility is available in Version 5.1A and in patch kits for earlier versions of the operating system. See your customer service representative for more information.

If the `verify` command cannot run or if it detects on-disk (metadata) corruption, unmount your filesets and run the `fixfdmn` command.

Remount the filesets. If you are unable to do this, run the `fixfdmn -u` command to undo the process so that the `salvage` command can access the unchanged data.

DO NOT attempt to undo the results of the `fixfdmn` command if you have made any subsequent changes to your domain. Corruption will result.

See also:
Version 5.1A Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Fixing On-Disk Metadata Corruptions"
Reference Pages: `fixfdmn(8)`

Top of Page

Salvaging a damaged domain

The `salvage` command extracts salvageable files from a corrupted domain and places copies of the files in directories created to hold them or on tape. Run the `salvage -d time` utility, where *time* is the date and time of your most recent backup, to recover only the files that have changed since this date. If you have no backups, run the utility without the `-d time` option to recover all the files in the domain. You must be the root user to run the utility.

Enhancements to the `salvage` utility are included in the Tru64 UNIX patch kits. For the latest version, check the latest patch kit or ask your Field Service Representative for the latest patch kit.

Running the `salvage` utility does not guarantee that you recover all of your domain. You might not be able to recover files, directories, names of files, or parts of files. The amount of data you are able to recover depends upon the damage to your domain.

The fastest salvage process is to recover file information to another domain. If there is not enough room on disk for the recovered information, you can recover data to tape, recreate your domain, and then write it back onto your original domain. If you do not have enough disk space, run the `salvage` utility with the `-F` and `-f` options to specify tar format and tape drive.

DO look at the `salvage` utility log file or the files on the tar tape to be sure you have gotten all the files you need. **DO NOT** proceed until you have collected all the information you can. Writing over the corrupt data on the disk is an irreversible process. Once you have overwritten old data, there is no way to rerun the `salvage` command to retrieve it.

If you have run the `salvage` utility and have been unable to recover a large number of files, and if you have not reused the volume from the original domain, you can run the `salvage` utility with the `-S` option. This process is very slow because the utility has to scan the entire volume looking for metadata pages. Running the `salvage` command in this fashion can present a security problem. It might be possible to recover data from older, deleted AdvFS domains while attempting to recover data from the current domains.

See also:
For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
Reference Pages: `salvage(8)`

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Recovering File Data from a Corrupted Domain"
Reference Pages: `salvage(8)` Top of Page

Creating a domain for recovery

The easiest way to recover domain content is to create a new domain (see Creating a domain in Setup:Domains) and restore the files to it from backup. You can then remove the original domain.

If you do not have adequate space for another domain or if you do not have a spare volume on which to house the domain temporarily or if you do not have AdvFS Utilities:

1. Save your existing filesets to tape either through backup (see Upkeep:Backup) or with the `salvage` utility.
2. Create a new domain on unused storage with the `mkfdmn` command. Create a mount-point directory for the new domain with the `mkdir` command.
3. Create filesets with the `mkfset` command.
4. Mount the new filesets.

DO NOT perform recovery steps that overwrite your old domain or use the `mkfdmn` command with your existing domain name unless you have created a backup.

If you need to add a volume and have lost access to your license registration, for your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Restoring a Multivolume usr Domain"

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Restoring a Multivolume usr Domain" Top of Page

Repopulating the domain

From backup and the output of the `salvage` command, restore the filesets that the `verify` and `fixfdmn` commands could not fix.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `vdump(8)` and `vrestore(8)`

Top of Page

Advanced File System Quick Reference

Troubleshooting: Avoiding Problems

Some problems can be avoided simply by being aware of scenarios that can cause file system corruption and avoiding them.

Upgrading domains to overcome disk file structure incompatibility

By default, domains created on Tru64 UNIX Version 5.0 Operating System software and later have a new on-disk format that is incompatible with earlier versions (see Creating a domain in Setup:Domains). The newer operating system recognizes the older disk structure, but the older does not recognize the newer.

If you perform an update installation (not a full installation) to the Version 5 operating system, all domains transferred in the installation process (including `root`, `usr`, and `var`) remain Domain Version Number (DVN) 3. All domains created after the update are DVN4.

To update existing domains to the new format, see Upgrading a domain to DVN4 in Setup:Domains.

If you try to mount a fileset belonging to a DVN4 domain when you are running a version of the operating system earlier than Version 5.0, you get an error message. To access a fileset in a DVN4 domain from an older operating system, NFS mount the fileset from a Version 5.0 system or upgrade your operating system to Version 5.

DO NOT run Version 4.x operating system software utilities on a machine running Version 5 operating system software. **DO NOT** run Version 5 operating system software utilities on a machine running a Version 4 operating system. They are not necessarily compatible and might cause corruption.

Top of Page

Version dependency

If a system crashes, AdvFS performs recovery at reboot. Domains that had mounted filesets at the time of the crash are recovered when the filesets are remounted. This recovery keeps the AdvFS metadata consistent and makes use of the AdvFS transaction log.

Since different versions of the operating system use different transaction log structures, it is important that you recover your filesets on the version of the operating system that was running at the time of the crash. If you do not, you risk corrupting the domain metadata and/or panicking the domain.

Therefore, if you want to reboot using a different version of the operating system, first cleanly unmount all AdvFS filesets. If the domain or the system panicked, it is best to reboot using the original version of the operating system.

Top of Page

Backing up to a block device

DO NOT back up to block 0 (that is, partition `a` or `c`) unless you have first cleared the disk label with the `disklabel`

`-z` command.

If you back up to a block device that has a disk label in block 0, you do not get an error message on `vdump` or `rvdump`, but the disk label is not overwritten and the saveset is invalid.

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `disklabel(8)`, `vdump(8)`, and `vrestore(8)`

Top of Page

Reusing volumes labeled AdvFS

All volumes (disks, disk partitions, LSM volumes, and so on) are labeled either "unused" or with the file system for which they were last used. You can only add a volume labeled "unused" to your domain. Relabel a volume "unused" with the `disklabel` command.

If the volume you want to add is part of an existing domain (an `/etc/fdmns` directory entry exists), the easiest way to return the volume label to unused status is to remove the volume with the `rmvol` command or to remove the domain with the `rmfdmn` command, which relabels all volumes that were in the domain as "unused."

If the volume you want to add is not part of an existing domain but is giving you a warning message because it is labeled, reset the disk label.

Top of Page

Advanced File System Quick Reference

Terminology Overlap

There are a number of system activities where the same word is used or the word is part of a phrase whose meaning depends upon its context.

Clone

- An AdvFS fileset clone is a read-only snapshot of fileset data at the time the clone was created. It is updated as the fileset is updated. AdvFS fileset cloning is analogous to the HSG80 controller 'ADD SNAPSHOT_UNITS' command.
- An HSG80 controller clone is a complete physical copy of the system so there is no need to track changes.
- A system clone is a complete copy of an operating system used to generate the system on more than one installation.

Volume

- An AdvFS volume is anything that behaves like a UNIX block device. This can be a disk, a disk partition, or logical storage.
- An LSM volume is an object that represents a hierarchy of plexes, subdisks, and LSM disks in a disk group. Applications and file systems make read and write requests to the LSM volume.

Domain

- An AdvFS domain is a named pool of storage that contains one or more volumes.
- In networking terminology, a domain is a collection of related systems on a subnetwork of an intranet or the Internet.
- In Windows NT, a domain is a collection of computers defined by the administrator of a Windows NT Server network that share a common directory database.

Top of Page

Advanced File System Quick Reference

Calling technical support

File a report with your software support organization if you experience:

- Domain panic
- Data corruption
- System panic

Describe what the machine was doing when the problem occurred and what might have led to it. Also report the following:

- Operating system version
- Dump information
- Metadata (see Saving metadata in Troubleshooting:Recovery Procedures)
- Output from the `sys_check` utility
- System error logs
- Error messages

See also:
For your version of the operating system, see:

Tru64 UNIX publications page
Reference Pages: `sys_check(8)`

Top of Page

Advanced File System Quick Reference

Additional resources

Compaq Tru64 UNIX Best Practices documentation provides you with a recommended method to handle a task, rather than presenting all of the options available. Check the System Management section.

Additional resources can be accessed through the tru64.org page:

- Tru64-UNIX-Managers List, a quick-turnaround troubleshooting discussion by administrators of Alpha systems running Tru64 UNIX, and the list archives.
- Encompass, a Compaq user group
- Links to documentation and Tru64 UNIX FAQs

Some of these are resources created by users, administrators, and developers working with the system, rather than by Compaq Computer Corporation employees. They do not represent an official corporate statement.

Top of Page

Advanced File System Quick Reference

Glossary

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

AdvFS Utilities	Utilities that allow you to create and manipulate multivolume domains. It is a layered product available by license.
asynchronous I/O	An nonblocking I/O scheme where data is written to the cache and may return control to the calling application before the data is written to disk.
atomic-write data logging	Guarantees that either all data in a write system call (up to 8 kilobytes) is written to the disk, or that none of the data is written to the disk.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

balance	To even the distribution of files between volumes of a domain.
bitfile	A set of pages that AdvFS views as one entity. Reserved files and user files are bitfiles.
block	A 512-byte unit of disk storage. Sixteen blocks comprise a page.
BMT	Bitfile Metadata Table. An array of 8-kilobyte pages, each with a header and an array of mcells. Located on each volume, the BMT contains all metadata for all files that have storage on the volume.
buffer cache	The area of memory that contains the blocks of data read from and/or waiting to be written to disk.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

clone fileset	A read-only copy of a fileset that is created to capture fileset data at a particular time. The contents of the clone fileset can be backed up while the original fileset remains available to users.
contiguous	Storage that is physically adjacent on a disk volume.
copy-on-write	The process by which original information is saved in a clone fileset when data in the original file is changed.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

defragment	To make each file and free space in a domain more contiguous.
direct I/O	An I/O scheme that synchronously reads and writes data from a file without copying it into a cache.
dirty data	Data that has been written by the application, but the file system has cached it in memory so it has not yet been written to disk.
domain	In AdvFS, named pool of storage that contains one or more volumes.
domain ID	A set of numbers that identify the domain to the system.
domain panic	A condition that prevents further access to the domain when corruption in the domain is detected. AdvFS allows the filesets in the domain to be unmounted after a domain panic.
DVN	Domain Version Number. A number in the on-disk metadata that specifies file structure. Domains created under Version 5.0 and later operating system software are DVN4, while domains created under earlier operating system software are DVN3.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

/etc/fdmns directory	A directory that contains the domain definitions.
/etc/fstab file	A file that identifies file systems that are to be mounted at system reboot.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

file fragment	Created when a file uses only part of the last page of file storage allocated or has a total size of less than 8 kilobytes.
fileset	A hierarchy of directory and files. A fileset represents a mountable portion of the directory hierarchy of the AdvFS file system.
fileset quota	A quota that limits the amount of disk storage that a fileset can consume or the number of files a fileset can contain.
frag file	A special file that stores the last partial page of a file in a fileset. Using frag files reduces the amount of wasted disk space.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

grace period	The period of time a quota's soft limit can be exceeded as long as the hard limit is not exceeded.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

hard limit	The quota limit for disk block usage or number of files that cannot be exceeded.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Logical Storage Manager (LSM)	Logical Storage Manager is a storage management system that provides data redundancy and volume-level striping. It is a layered product available by license.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

mcell	Metadata cells that contain records of file structure.
metadata	File structure information such as file attributes, extent maps, and fileset attributes.
migrate	To move files from one volume to another within a domain.
mirror	(v.) To maintain identical copies of data in different storage areas. (n.) One of the copies that is maintained.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

page	An allocation of 8 kilobytes of contiguous disk space (16 blocks).

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

quota file	A file that stores quota limits and keeps track of number of files, disk block usage, and grace period per user ID or per group ID. User and group quota files cannot be deleted, but need not be used. Fileset quota information is stored within the fileset metadata.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

RAID	Redundant array of independent disks. A technology that provides high disk I/O performance and data availability through mirroring, striping, and parity calculation. The Tru64 UNIX operating system provides RAID functionality by using disks and software (LSM). Hardware-based RAID functionality is provided by intelligent controllers, caches, disks, and software.
root tag file	A directory that defines the location of all filesets in a domain. Each file domain has one.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

saveset	A collection of blocks created by the `vdump` utility to save AdvFS backup information.
SBM	Storage Bitmap. Keeps track of allocated disk space in a volume.
soft limit	The quota value beyond which disk block usage or number of files is allowed only during the grace period.
sparse file	A file whose pages do not all have allocated disk space.
StorageWorks	A Compaq licensed product that includes AdvFS Utilities and Logical Storage Manager.
stripe	To distribute blocks of a file across multiple storage areas. AdvFS stripes individual files. LSM and hardware RAID stripe all files.
synchronous I/O	An I/O scheme where data is written both to the cache and to the disk before the write request returns to the calling application.
SysMan	A command-line wrapper that allows the system administrator to perform system management tasks from a user interface.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

tag	A unique identifier for an AdvFS file within a fileset.
transaction log	The log that records changes to metadata before the changes are written to disk. These changes are written to disk at regular intervals.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

volume	For AdvFS, anything that behaves like a UNIX block device. This can be a disk, disk partition, or logical storage.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

write-ahead logging	The process by which the modifications to the file-structure information are completely written to a transaction log before the actual changes are written to disk.

Top of page

Advanced File System Quick Reference

Frequently Asked Questions

1. Why do du, df, showfsets, and ls give different values?

For a comparison of the commands that look at disk space, see Checking fileset and domain disk space in Upkeep:Domain Upkeep.

Top of Page

2. How do I find the volumes in a domain?

Go to `/etc/fdmns/`*`domain_name`* and use the `ls` command. For example:

```
# cd /etc/fdmns/usr_domain
# ls
dsk8d
```

Top of Page

3. How do I convert from AdvFS to UFS?

An appendix in the AdvFS documentation details conversions to and from AdvFS. For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Converting File Systems"

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Converting File Systems" Top of Page

4. Where can I find AdvFS file system limits?

For operating systems prior to Version 5.1A, see the Tru64 UNIX publications page, Related Topics, Software Product Description.

For Version 5.1A, see the documentation page.

Top of Page

5. What is AdvFS Utilities?

AdvFS Utilities is a layered product available by license. StorageWorks contains AdvFS Utilities and Logical Storage Manager (LSM). The utilities allow you to create and manipulate multivolume domains. Contact your vendor for more information.

Top of Page

6. Why is my quota file so large? I haven't set quotas.

AdvFS keeps user and group quota information in two files: `quota.user` and `quota.group`. These files cannot be deleted. Quota files are sparse files, so their boundaries may make them appear quite large when viewed with the `ls` command. However, the `du` command shows the actual space that is used by the file.

For your version of the operating system, see:

Version 4.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Guide to File System Administration
"Quotas"

Version 5.x Tru64 UNIX publications page
System and Network Management Documentation Bookshelf:
AdvFS Administration
"Quotas" Top of Page

7. How do I find which volume my file is on?

The `showfdmn` displays all the volumes in the domain. Use the `showfile` command to determine the location of a particular file. Use the `migrate` command to move a file from one volume to another.

Top of Page

8. Help! I used `mkfdmn` (or `addvol`) and ruined the domain I was trying to recover

See Troubleshooting:Recover Procedures.

Top of Page

9. I'm not familiar with Compaq. How do commands compare?

You can look at an unauthorized command comparison on in A Sysadmin's Universal Translator (ROSETTA STONE)

Top of Page

10. How do I put more than one dump (saveset) on a tape?

Use the `vdump` command and specify a no-rewind tape such as `/dev/ntape/tape0`. Then save the next saveset after

the previous. For certain SCSI/CAM drives, you can use the `mt -f /dev/ntape/tape-id fsf seod` command to skip to the end of the last saveset.

To restore from a tape containing multiple savesets, rewind the tape then use the `mt fsf` *n* command to skip to the *n*th saveset. You can also use `vrestore -x` command and specify the files you want to restore.

Top of Page

11. I've converted to a Version 5 operating system and defragmenting is much slower.

For Version 5 and later operating systems, the `defragment` utility creates a thread per volume. This makes defragmenting many small volumes faster than defragmenting a large one. However, multiple threads can severely affect I/O performance. You can emulate the single-thread performance of earlier operating systems with `defragment -N 1`.

Top of Page