VERITAS File System 4.1 (HP OnlineJFS/JFS 4.1) Release Notes

HP-UX 11i v2

Sixth Edition

Manufacturing Part Number: 5991-2762

December 2005

Printed in the United States

Legal Notices

Trademark Notices

Contents

Preface

The *VERITAS File System 4.1 (HP OnlineJFS/JFS 4.1) Release Notes* provides information on the September 2005 release of VERITAS File System 4.1.

Publication History

The manual publication date and part number indicate its current edition. The publication date will change when a new edition is released. The manual part number will change when extensive changes are made.

To ensure that you receive the new editions, you should subscribe to the appropriate product support service. See your HP sales representative for details.

- First Edition: December 2002, 5187-1878, HP-UX 11i Version 1(B.11.11)
  CD-ROM (Software Pack 11i December 2002) and Web (**http://docs.hp.com**)
- Second Edition: August 2003, 5971-4774, HP-UX 11i Version 2 (B.11.23)
  Web (**http://docs.hp.com**)
- Third Edition: December 2003, B3929-90015, HP-UX 11i Version 2 (B.11.23)
  Web (**http://docs.hp.com**)
- Fourth Edition: May 2005,5991-1139, HP-UX 11i Version 2 (B.11.23)
- Fifth Edition: September 2005, 5991-1832, HP-UX 11i Version 2 (B.11.23)
- Sixth Edition: December 2005, 5991-2762, HP-UX 11i Version 2 (B.11.23)

Technical Support

For license information contact:

- Software License Manager:

  `http://licensing.hp.com/welcome.slm`

- HP Licensing Services contact:

  `http://licensing.hp.com/licenseAdmins.slm`

- (Europe)

  — Phone: +353.(0)91.75.40.06

Email: codeword_europe@hp.com

- (U.S. and Canada)
  - Phone: +1 650.960.5111
  - Email: hplicense.na@hp.com
- (Asia Pacific)

  Phone: 0120.42.1231 or 0426-48-9310 (Inside Japan)

  +81.426.48.9312 (Outside Japan)

For latest information on available patches visit:

- http://itrc.hp.com

For technical support visit:

- http://welcome.hp.com/country/us/en/support.html

Typographic Conventions

Table 1 describes the typographic conventions used in this document.

Table 1 Typographic Conventions

Typeface	Usage	Examples
monospace	Computer output, files, directories, software elements such as command options, function names, and parameters	Read tunables from the /etc/vx/tunefstab file. See the *ls* (1) manpage for more information.
italic	New terms, book titles, emphasis, variables replaced with a name or value	See the *VERITAS File System 4.1 Release Notes*for details.
%	C shell prompt	Not applicable
$	Bourne/Korn shell prompt	Not applicable
#	Superuser prompt (all shells)	Not applicable
\	Continued input on the following line; you do not type this character	# **mount -F vxfs \** **/h/filesys**

Table 1 Typographic Conventions (Continued)

Typeface	Usage	Examples
[ ]	In command synopsis, brackets indicates an optional argument.	`ls [ -a ]`
|	In command synopsis, a vertical bar separates mutually exclusive arguments.	`mount [ suid | nosuid ]`
blue text	An active hypertext link	In PDF and HTML files, click on links to move to the specified location.

Related Documentation

For more information about VERITAS 4.1 products refer to the following documents located in the `/usr/share/doc` directory:

- *VERITAS File System 4.1 Release Notes*
- *VERITAS File System 4.1 Administrator's Guide*
- *VERITAS Volume Manager 4.1 Hardware Notes*
- *VERITAS Volume Manager 4.1 Release Notes*
- *VERITAS Volume Manager 4.1 Troubleshooting Guide*
- *VERITAS Volume Manager 4.1 Migration Guide*
- *VERITAS Volume Manager 4.1 Administrator's Guide*
- *VERITAS Enterprise Administrator (VEA 500 Series) Getting Started*
- *VERITAS Storage Foundation 4.1 Cross-Platform Data Sharing Administrator's Guide*
- *VERITAS Flashsnap Point-In-Time-Copy Solutions Administrator Guide*

HP Encourages Your Comments

HP encourages your comments concerning this document. We are committed to providing documentation that meets your needs.

You can send comments to: `netinfo_feedback@cup.hp.com`

You must include document title, manufacturing part number, and any comment, error found, or suggestion for improvement you have concerning this document. Also, include what we did right so we can incorporate it into other documents.

1 VERITAS File System 4.1 Release Notes

This document provides information on VERITAS File System (HP OnlineJFS/JFS) Release 4.1. The VERITAS File System (HP OnlineJFS/JFS) 4.1 is supported on HP-UX 11i v2 September 2004 or later.

The following topics are discussed in this document:

- "New Features" on page 9
- "End of Product Support" on page 14
- "Unsupported Features" on page 15
- "Required HP-UX Patches" on page 16
- "Documentation" on page 17
- "Online Manual Pages" on page 17
- "Command Summary" on page 18
- "Software Issues in Base-VXFS" on page 20
- "Product Licensing" on page 26

New Features

The VERITAS File System 4.1 has the following new features and changes:

- **Cross-Platform Data Sharing (CDS)**

  Cross-platform data sharing feature allows data to be serially shared among heterogeneous systems that have direct access to the physical devices holding the data. This feature can be used only with VERITAS Volume Manager. For more information on Cross-Platform data sharing refer to the *VERITAS Storage Foundation 4.1 Cross-Platform Data Sharing Administrator's Guide*.

NOTE To use the CDS feature, an extended VxFS license is required which enables CDS.

- **Multi-Volume Support (MVS)**

  The multi-volume support feature allows several volumes to be represented by a single logical object. All I/O to and from an underlying logical volume is directed by way of volume sets. This feature can be used only in conjunction with VERITAS Volume Manager.

  The following commands are associated with MVS:

`fsapadm`	VxFS allocation policy administration utility.
`fsvoladm`	VxFS device administration utility.

  Table 1-1 lists the application programming interface (API) associated with MVS:

Table 1-1 Functions in the MVS Application Programming Interface

`vxfs_ap_assign_ckpt`	`vxfs_ap_assign_file`	`vxfs_ap_assign_fs`	`vxfs_ap_define`
`vxfs_ap_enforce_file`	`vxfs_ap_enumerate`	`vxfs_ap_remove`	`vxfs_ap_query`
`vxfs_ap_query_ckpt`	`vxfs_ap_query_file`	`vxfs_ap_query_fs`	`vxfs_vol_add`
`vxfs_vol_deencapsulate`	`vxfs_vol_encapsulate`	`vxfs_vol_enumerate`	`vxfs_vol_remove`
`vxfs_vol_resize`	`vxfs_vol_stat`		

NOTE Full-VxVM enables the creation of Multi-Volume sets using VxVM. But a VxFS file system cannot be created on this Multi-Volume set, as it would require a separate license for enabling Multi-Volume Support on VxFS.

- **Large File Systems and Large File Support**

  The VxFS 4.1 release is the first release to support Disk Layout Version 6. File system up to 256TB in size can be created on Disk Layout Version 6. However, file system larger than 2 TB can be created only on 64-bit kernel on VERITAS Volume Manager. A 32 TB file system requires a 2 K block size, a 64 TB file system requires a 4 K block size, and a 128

TB or larger file system requires an 8 K block size. File systems over 32 terabytes are called extra large file systems and require a license to create. However, only file system upto 32TB has been qualified and will be supported as of this stage.Also, the OnlineJFS license is required to create file systems over 2 TB and under 32TB. The EBFS license is required to create file systems over 32TB.

An online conversion utility `vxupgrade` is provided to upgrade existing disk layouts to Disk Layout Version 6 on mounted file systems.

NOTE In this release of VxFS, file sizes can grow only up to 2 TB.

LVM, also supports creation of VxFS file system, of size, less than or equal to 2TB.

- **Quality of Storage Service (QoSS)**

  The Quality of Storage Service (QoSS) feature allows mapping of more than one device to a single file system.The QoSS feature is built on the multi-volume support technology introduced in this release.Using QoSS, policies can be configured, that automatically relocate files from one device to another,or files can be relocated using relocation commands. Having multiple devices improves performance for applications that access specific types of files. QoSS is managed using the VERITAS Enterprise Administrator (VEA) GUI, or the `fssweep`, `fsmove` and `fsrpadm` commands. QoSS functionality is a separately licensable feature that is available with the `VRTSfspro` package.

  The following commands are associated with QoSS:

`fsmove`	Relocates files to a destination component volume.
`fssweep`	Traverses the directory structure of one or more file systems.
`fsrpadm`	Administer relocation policies.

NOTE To use the QoSS feature, a separate license key is required.

- **Storage Checkpoint Quotas**

  Usage limits can be set on Storage Checkpoints. The following keywords were added to the `fsckptadm` command to administer Storage Checkpoints quota functionality:

`getquotalimit`	Displays current quota limits, current usage, and flags set.
`setquotalimit`	Sets the hard limit and soft limits for Storage Checkpoint quotas.
`quotaon` and `quotaoff`	Turns Storage Checkpoint quotas on and off.

- **Storage Checkpoint File System Restore**

  Storage Checkpoints can be used by backup and restore applications to restore either individual files or an entire file system. Restoring from Storage Checkpoints can recover data from incorrectly modified files, but typically cannot be used to recover from hardware damage or other file system integrity problems. File system restoration can be done using the `fsckpt_restore(1M)` administrative command.

  There is one new command associated with this feature:

  `fsckpt_restore` VXFS Storage Checkpoint file system restoration utility.

- **File Change Log**

  The VxFS File Change Log (FCL) tracks changes to files and directories in a file system. The File Change Log can be used by applications such as backup products, webcrawlers, search and indexing engines, and replication software that typically scan an entire file system searching for modifications since a previous scan.

  The following command is associated with FCL:

  `fcladm` VxFS File Change Log administration utility.

  The following new tunable parameters are associated with FCL:

  `fcl_keeptime` Specifies the minimum amount of time for which FCL keeps records in the log.

  `fcl_maxalloc` Specifies the maximum amount of space that can be allocated to the FCL.

  `fcl_winterval` Specifies the interval that must elapse before the FCL records subsequent writes to a file.

  See the `vxtunefs(1M)` online manual page for more information on tunable parameters.

  The File Change Log feature includes the following application programming interface (API) functions:

  `vxfs_fcl_sync` Sets a synchronization point in the File Change Log.

  FCL functionality is a separately licensable feature.

- **Reverse Path Name Lookup**

  The reverse path name lookup feature uses the vxlsino command to obtain the full path name of a file or directory corresponding to an inode. Reverse path name lookup can be useful in applications, such as for clients of the VxFS File Change Log feature, in backup and restore utilities, and in replication products.

  The following command is associated with the reverse name lookup:

`vxlsino`	VxFS reverse path name lookup utility.

The reverse path name lookup feature includes the following application programming interface (API) functions:

`vxfs_inotopath`	Returns path names for a given inode number.
`vxfs_inotopath_gen`	Returns path names for a given inode number with a specific generation count.

- **Intent Log Resize**

  The VxFS intent log can be resized using the `log` option of the `fsadm` command. Dynamically changing the intent log size can improve system performance. The maximum intent log size is now 2 gigabytes. For more information on intent log resizing refer to the *VERITAS File System Administrator's Guide* and the `mkfs_vxfs(1M)` and `fsadm_vxfs(1M)` manpages for more information on `intent log` size.

- **Named Data Streams**

  The named data streams feature allows the inode to be associated with multiple data streams. In VxFS 4.1, the original data stream is accessed in the same way as in the previous releases, but new data streams are referenced through a new directory inode associated with the file. The directory inode points to the new inodes, one per stream. The directory containing the named streams is not directly visible to the user.

  VxFS named data stream functionality is available through the following application programming interface (API) functions:

`vxfs_nattr_link`	Links to a named data stream.
`vxfs_nattr_open`	Open a named data stream.
`vxfs_nattr_rename`	Renames a named data stream.
`vxfs_nattr_unlink`	Removes a named data stream.

- **Other Base-VXFS Enhancements**

  — The `fsadm` command allows fragmentation, reporting, and defragmentation operations on an individual file or directory, or on Storage Checkpoints.

  — Disk layout versions prior to version 4 cannot be mounted. Use the `vxfsconvert` command to convert them to a disk layout version that can be mounted.

  — The `vxupgrade` command upgrades only Disk Layout Version 4 and Disk Layout Version 5. Disk layout versions prior to version 4 cannot be mounted.

  — The `largefiles` option is now the default file size option for the `mount` command and `mkfs` command.

— The delaylog option is now the default intent logging mount option. The change of the default mount option from log to delaylog does not increase the risk of data loss, but allows Base-VxFS to cache data to improve performance. For more information on the delaylog option refer to *VERITAS File System 4.1 Administrator's Guide*.

— The default Storage Checkpoint creation mode is now removable.

— Operation of the intent log replay was improved to increase the speed of recovery after a file system failure.

— The histlog function was implemented in the fsdb_vxfs command. The history log records structural changes to the file system to aid in product support.

— Two new tunable parameters, inode_aging_count and inode_aging_size, for use with the Storage Checkpoint API, were added to the vxtunefs command. See the vxtunefs(1M) manual page for more information.

More VxFS functions can be performed from the VERITAS Enterprise Administrator (VEA) GUI. For more information on VEA refer to the *VERITAS Enterprise Administrator (VEA 500 Series) Getting Started guide*.

The vxfsu_get_ioffsets library call was renamed vxfs_get_ioffsets.

- **VxFS Enhancement**

  The vx_ninode and vxfs_bc_bufhwm can be dynamically tuned without rebooting the machine. The changes take effect immediately, although they can be optionally made to take effect only after a reboot. For for more information on VxFS enhancements refer to the *VERITAS File System 4.1 Administrator's Guide*.

End of Product Support

The VxFS 4.1 release does not support the creation or mounting of file systems using the Disk Layout Version 3. However, VxFS 4.1 still provides the capability to upgrade Disk Layout Version 3 using the vxfsconvert utility, but this is the last major release to do so. HP recommends that you upgrade all older disk layouts.

HP recommends that you upgrade file systems using older disk layouts to Disk Layout Version 6. For more information on upgrading see “Compatibility With Previous Versions of VxFS” on page 16.

The QuickLog feature is not available in the VxFS 4.1 release. The multi-volume support feature replaces most of the functionality provided by QuickLog. For more information on migrating QuickLog devices to the multi-volume support function refer to the *VERITAS File System 4.1 Administrator's Guide*.

This is the last release to support shared (cluster) mounts on VxFS Disk Layout Version 5 cluster file systems. Upgrade all current cluster file system disk layouts to Disk Layout Version 6 to enable new VERITAS File System features and to support upgrades in future releases.

Unsupported Features

The following features are not supported in this release:

- VERITAS Quick I/O for Databases
- VERITAS QuickLog
- VERITAS FlashSnap
- Storage Checkpoints
- Cluster File Systems
- Moving VERITAS Volume Manager Volumes or VERITAS File Systems from HP-UX 11i Version 1 to HP-UX 11i Version 2 systems has not yet been tested and is not supported.

Compatibility With Previous Versions of VxFS

VxFS 4.1 file systems employ Disk Layout Version 6. With the VxFS 4.1 release, you can no longer create or mount Disk Layout Version 3 file systems. VxFS 4.1 is the last major release to support Disk Layout Version 4 and Disk Layout Version 5. HP recommends upgrading any previously installed VxFS file system to the Disk Layout Version 6 available with VxFS 4.1.

After installing Base-VxFS 4.1, you can upgrade the disk layout on mounted file systems using the `vxupgrade` command (see the `vxupgrade(1M)` manual page for details), or `vxfsconvert` (see the `vxfsconvert(1M)` manual page) to upgrade the disk layout of unmounted file systems post installation.

CAUTION VxFS file systems must be cleanly unmounted before upgrading to the VERITAS File System 4.1 release from any previous release.

Required HP-UX Patches

For a list of the required HP-UX patches, refer to the *VERITAS 4.1 Installation Guide*.

NOTE If LVM is going to be used with Disk Layout Version 6 and VxFS 4.1, the LVM patch `PHCO_33308` needs to be installed. This patch provides the correct `LIBC` relink with LVM

If LVM is used the user needs to install the following patches as a safeguard for potential data corruption.

- **LVM**

  The default disk layout of VxFS 4.1 is version 6. When reducing a volume, LVM checks on the volume to prevent part of the existing filesystem from getting lost. Likewise, when an LVM physical volume is created, LVM checks to make sure that it is not clobbering an existing filesystem. LVM does not recognize a VxFS 4.1 filesystem with disk layout version 6. To make LVM work with Disk Layout Version 6, PHCO_33308 needs to be installed.

- **swapon**

  `PHCO_33238` must be installed on the system for `swapon` command to enable VxFS 4.1 filesystem for paging.

WARNING **System Administration Manager (SAM) does not recognize a VxFS 4.1 file system with Disk Layout Version 6.**

Documentation

The *VERITAS File System 4.1 Administrator's Guide* and the *VERITAS File System 4.1 Release Notes* are available under the `/usr/share/docs` directory. These documents are provided as PDF files. To view or print PDF documents, you need the Adobe Acrobat Reader.

Online Manual Pages

Table 1-2 lists the VERITAS File System 4.1 manpages.

Table 1-2 **VERITAS File System 4.1 Manpages**

Section 1M	**Description**
`extendfs_vxfs`	Extend the size of VxFS file system.
`getext`	Gets extent attributes for a VxFS file system.
`setext`	Sets extent attributes on a file in a VxFS file system.
`df_vxfs`	Reports the number of free disk blocks and inodes for a VxFS file system.
`ff_vxfs`	Lists file names and inode information for a VxFS file system.
`fsadm_vxfs`	Resize or defragment a VxFS file system.
`fscat_vxfs`	Cats a VxFS file system.
`fsck_vxfs`	Checks and repairs a VxFS file system.
`fsdb_vxfs`	VxFS file system debugger.
`fstyp_vxfs`	Returns the type of file system on a specified disk partition.
`mkfs_vxfs`	Constructs a VxFS file system.
`mount_vxfs`	Mounts a VxFS file system.

Table 1-2 VERITAS File System 4.1 Manpages (Continued)

ncheck_vxfs	Generates path names from inode numbers for a VxFS file system.
vxdump	Incremental file system dump.
vxfsconvert	Converts an unmounted file system to VxFS or upgrades a VxFS disk layout version.
vxfsstat	Displays file system statistics.
vxrestore	Restores a file system incrementally.
vxtunefs	Tunes a VxFS file system.
vxupgrade	Upgrades the disk layout of a mounted VxFS file system.
Section 4	**Description**
fs_vxfs	Format of a VxFS file system volume.
inode_vxfs	Format of a VxFS file system inode.
tunefstab	VxFS file system tuning parameters table.
Section 7	**Description**
vxfsio	VxFS file system control functions.
vxlsino	VxFS reverse path name lookup utility.
vxumount	Unmount a VxFS File System

Command Summary

Table 1-3 lists the VxFS commands.

Table 1-3 VERITAS File System 4.1 Commands

Command	Description
df	Reports the number of free disk blocks and inodes for a VxFS file system.
extendfs	Extend a VxFS file system size.

Table 1-3 VERITAS File System 4.1 Commands (Continued)

Command	Description
`ff`	Lists file names and inode information for a VxFS file system.
`fsadm`	Resize or defragment a VxFS file system.
`fscat`	Cats a VxFS file system.
`fsck`	Checks and repairs a VxFS file system.
`fsdb`	VxFS file system debugger.
`fstyp`	Returns the type of file system on a specified disk partition.
`getext`	Gets extent attributes for a VxFS file system.
`mkfs`	Constructs a VxFS file system.
`mount`	Mounts a VxFS file system.
`ncheck`	Generates path names from inode numbers for a VxFS file system.
`newfs`	Creates a new VxFS file system.
`setext`	Sets extent attributes on a file in a VxFS file system.
`vxdump`	Incremental file system dump.
`vxedquota`	Edits user quotas for a VxFS file system.
`vxfsconvert`	Converts an unmounted file system to VxFS or upgrades a VxFS disk layout version.
`vxfsstat`	Displays file system statistics.
`vxlsino`	VxFS reverse path name lookup utility.
`vxrestore`	Restores a file system incrementally.
`vxtunefs`	Tunes a VxFS file system.
`vxupgrade`	Upgrades the disk layout of a mounted VxFS file system.
`vxumount`	Unmount a VxFS File System

Software Issues in Base-VXFS

- **Large File Support**

  VxFS 4.1 only supports files up to 2 TB on 64-bit kernels.

- **Mapping a Large Offset of a File**

  Calling the `mmap(2)` function on a large offset of a file might require a large amount of swap space. If `mmap(2)` is called on a 1 TB offset in `MAP_SHARED` mode, then approximately 256 MB of virtual memory pages could be instantiated. This means that VM requires space to hold 256 MB of data structures. Each of these structures are 16 bytes, thus approximately 4 GB space is required. Since these data structures are able to be paged, more than 4 GB of swap space is reserved when `mmap(2)` is called.

- **Setting max_thread_proc Tunable**

  Setting `max_thread_proc` to a low value may result in the system being in a hung state. If the value of this tunable is less than 1100, the value will be automatically set to 1100 when VxFS is installed. This value must be maintained at or above 1100.

- **DMAPI `dm_get_dirattrs` Can Skip Directory Entries**

  `vx_hsm_get_dirattr()` reads a directory's dirents 8 KB at a time, if the passed in user buffer is not large enough to hold the combined entries' corresponding file stats information (up to ~114kb). Then the next call to `vx_hsm_get_dirattr()` will continue from the wrong offset within the directory, resulting in some directory entries being skipped. HP keeps track of this defect with CR JAGaf75359.

NOTE NoteThere is no error indication. The expectation is to continue from the last dirent information that could fit into the user's buffer. However, the next directory read will begin 8 KB further down the directory.

  As a workaround this issue can be avoided by passing a sufficiently large user buffer.

- **File System With Quotas Cannot Be Mounted After Doing CDS Conversion**

  If a file system is using quotas, it cannot be mounted with quotas turned on after converting the byte order of the file system with the fscdsconv(1M) utility. Similarly, quotas cannot be turned on, if the converted file system is first mounted with quotas turned off. See the `fscdsconv(1M)` manual page for the workaround.

- **CDS Validate Error Running in ja_JP.UTF-8-Encoded Locale**

  The fscdstask validate command returns an error when files on the specified mount point have names with ja_JP.eucJP or ja_JP.PCK encodings, but the locale has been changed to jp_JP.UTF-8. The error is as follows:

```
xargs: Input file is corrupt. : Incorrect byte order
```

Files should be created with the same locale encoding as the file system on which they reside.

- **Large Files Should Be Mounted Only on Systems With Sufficient Memory**

  When a file system is mounted, VxFS keeps certain data structures in the kernel. As the size of the file system increases, the amount of data structures stored by VxFS also increases. The file system typically keeps approximately 128 bytes per allocation unit (32,768 file system blocks). This translates to a usage of 512K per 1 TB for an 8K block size file system (4 MB per 1 TB for a 1K block size file system). Therefore, large file systems must be mounted only on systems that have sufficient memory. The memory requirements for mounting large file systems are shown in the tables below.

Table 1-4 Memory Usage for a File System With a 1K Block Size

File System Size	128GB	1TB	8TB	64TB	256TB
Memory Usage	1MB	4MB	32MB	N/A	N/A

Table 1-5 Memory Usage for a File System With a 2K Block Size

File System Size	128GB	1TB	8TB	64TB	256TB
Memory Usage	512K	2MB	16MB	128MB	N/A

Table 1-6 Memory Usage for a File System With a 4K Block Size

File System Size	128GB	1TB	8TB	64TB	256TB
Memory Usage	256K	1MB	8MB	64MB	N/A

Table 1-7 Memory Usage for a File System With an 8K Block Size

File System Size	128GB	1TB	8TB	64TB	256TB
Memory Usage	128K	512K	4MB	32MB	128MB

While performing a full fsck, the system keeps certain data structures in the core for validating the space usage and inode usage. The space needed depends on the number of inodes and the number of blocks in the file system. The fsck command needs approximately 16 MB per 1 TB for an 8K block size file system (128 MB per 1 TB for a 1K block size file system) and 32 MB per million inodes. Sufficient memory and swap space should be configured on the system before running a full fsck on a large file-enabled system.

A replay fsck does not need a significant amount of memory and does not have these issues.

- **Some Disk Quota Operations Do Not Function on NFS**

  When VxFS file systems are exported via NFS, quotas on the file system apply to users when accessing the file system from NFS clients. However, the HP-UX quota commands on the NFS client cannot be used to edit quotas. The VxFS quota commands can be used on the server to edit quotas.

- **Files and Directories**

  To maximize VxFS performance, do not exceed 100,000 files in the same directory. Use multiple directories instead.

- **Cross-Platform Data Sharing (CDS)**

  — External Quota File

  CDS does not support converting the external quota file. The external quota file would need to be manually migrated as follows:

  On source:

  Prior to unmounting the file system, remove the quotas file. Run the fscdsconv command after unmounting the file system.

  On target:

  Mount the file system without quotas. Manually edit the quotas and the quotas.grp file to enter the limits.

  — Handling ACLs with CDS

  Because not all target platforms support ACLs, converting a file system with ACLs from the source to a target on which ACLs are not supported results in ACLs not being enabled on the target. If the file system is converted back to a target on which ACLs are supported, permission checks are enforced again.

- **Quick I/O Files Cannot Be Sparse Files**

If you try to convert a sparse file to a Quick I/O file, the Oracle instance can fail if Oracle tries to write into an unallocated block. Specifically, datafiles used by the Oracle8i and Oracle9i temporary tablespace may be sparse files, so do not convert these to Quick I/O files. See the VERITAS Storage Foundation 4.1 for Oracle Database Administrator's Guide for more information.

- **PHCO_33308 Required for LVM Commands to Recognize Disk Layout Version 6**

  When reducing a volume, LVM checks the volume to prevent any part of the existing file system from being thrown away. Likewise, when an LVM physical volume is created, LVM ensures that it is not clobbering an existing file system. However, LVM does not recognize a VxFS 4.1 file system with disk layout Version 6, which is the default disk layout version in VxFS 4.1. Install PHCO_33308 for LVM to work with disk layout Version 6.

- **PHCO_33238 Required for swapon Command to Recognize Disk Layout Version 6**

  The swapon command does not recognize a VxFS file system with disk layout Version 6, and therefore does not produce a warning while using a device that has such a file system. Install PHCO_33238 to enhance the `swapon` command to recognize the VxFS disk layout Version 6.

- **Applications using `statvfsdev(3C)` interface need to relink with new LIBC**

  The applications which use `statvfsdev` (3C) interface on a device name, need to relink with the new `LIBC` library, with the option `=llibdlv_vxfs` to be able to understand Disk Layout Version 6, available with Base-VXFS 4.1.

  If the applications are not relinked, they will not be able to recognize Disk Layout Version 6 file system. However, they would continue to recognize Disk Layout Version 4 and Disk Layout Version 5 file system. The applications that are linked with the new library with the option `-llibdlv_vxfs` will work with VxFS 3.5 on HP-UX 11i version 2 without re-linking. However, the dynamic binaries need the new library on HP-UX 11i version 2 for VxFS 3.5 to work.

- **System Hang Occur on Systems With Less Cache Memory**

  VxFS allocates a fixed amount of memory. The amount of memory allocated can be changed using the tunables `vx_ninode` and `vx_bc_bufhwm`. After long hours of operation, low-memory systems, slow down or hang due to memory pressure. To solve this problem, lower the values of `vx_ninode` and `vx_bc_bufhwm` to limit VxFS memory consumption.

NOTE VxFS consumes a fixed percentage of memory for the default values of tunables such as `vx_ninode` and `vx_bc_bufhwm`. VxFS consumes nearly 10% of total physical memory for the default value of `vx_ninode`. If the system has 512 MB of memory, VxFS will require up to 52 MB to store the VxFS inode cache. Therefore it is necessary to tune the tunables according to the type of load on the system.

- **Tuning Down the Inode Cache**

  VxFS file systems allocate and release inodes based on the file system load. Generally, larger inode caches help the file systems to perform better in case of a file server or web server load. The global tunable, `vx_ninode`, determines the maximum possible size of the Base-VXFS inode cache. If the default value of `vx_ninode` is set to zero, Base-VXFS automatically tunes the size of the inode cache at boot time, based on the size of the physical memory on the system. On systems with a RAM size of less than or equal to 1 GB per CPU, you can manually tune down `vx_ninode` to a value not less than that of `nfile` (`nfile` is an HP-UX tunable that represents the maximum number of file descriptors). The `vxfsstat` command can be used to check the current `vx_ninode` usage.

- **Tuning Down the Buffer Cache**

  VxFS 4.1 implements a private buffer cache to use exclusively for metadata. The allocations to this buffer cache vary during system usage based on file system load and the maximum cache size specified in the global tunable, `vxfs_bc_bufhwm`. If the value of `vxfs_bc_bufhwm` is set to zero, VxFS automatically tunes the maximum size of the metadata buffer cache at boot time based on the system memory size. On systems with a RAM size of 1 GB per CPU or less, you can manually tune down the value of `vxfs_bc_bufhwm` to a minimum of 6MB.

- **Full File System Cannot Be Resized**

  In some circumstances, the `fsadm` command cannot resize a 100% full file system due to lack of space for updating structural information. If a file system is busy or too fragmented, the resize operation fails.

- **Data Integrity Issues with Disks and Disk Arrays with Write-back Caches**

  Disk drives configured to use a write-back cache, or disk arrays configured with a volatile write-back cache, exhibit data integrity problems. The problems occur after a power failure, SCSI bus reset, or other event in which the disk has cached data, but has not yet written it to non-volatile storage. A workaround for this problem is to disable the write-back cache.

- **The `fsadm` Command Cannot Truncate a Directory**

  The `fsadm` command cannot truncate a directory if it has only one extent that is more than two blocks in length, even if all the directory entries are deleted.

- **Inode Limitation on File Systems Without Large File Support**

File system with 8 million inodes can be created only using the largefiles option of mkfs command (the fsadm utility can also be used to set the largefiles flag on the file system). See the mkfs_vxfs(1M) and fsadm_vxfs(1M) manpages for details.

- **Some Fields Not Displayed by the fstyp Command**

  The fstyp -v command displays super-block information. For Disk Layout Version 4 and Disk Layout Version 5 fields such as nau, logstart or logend display zeros. For more information refer to the mkfs_vxfs (1M) and mkfs (1M) manpages.

- **A Change in the Method of Computing CUT Values Cause Misleading Error Messages to Display.**

  In this release, the method for computing the Current Usage Table (CUT) values for a Disk Layout Version 2 file system has changed.

  If a Version 2 file system is mounted on a system running VxFS 4.1, and that file system is subsequently used on an earlier version of VxFS, the following messages may display when performing a full fsck:

  ```
  vxfs fsck: incorrect CUT entry for filset 1, fix? (ynq)
  ```

  ```
  vxfs fsck: incorrect CUT entry for filset 999, fix? (ynq)
  ```

  This is expected and does not indicate file system corruption. Answer y to both questions. There is no need to perform a full fsck when moving such a file system to and from different versions of VxFS unless the file system is dirty, in which case a full fsck is required.

- **Non-standard Command Behavior When Using Access Control Lists (ACLs)**

  The output of the ls -l command on VxFS file systems shows mask /CLASS_OBJ in place of group permissions if ACLs are in use on a file or a directory. The getfacl command can be used to detemine the effective group permissions.

  The chmod command changes mask/CLASS_OBJ instead of the group permissions if ACLs are in use on a file or a directory. GROUP_OBJ is not changed by chmod, and because effective group permissions are determined by GROUP_OBJ and CLASS_OBJ, the default group may not receive the permissions specified by chmod. Because ls -l shows mask only (which is changed by chmod), it only appears that the group permissions are changed as specified in chmod. Use of chmod command is not advisable on files with ACLs. Instead, use the getfacl command to manipulate permissions.

  Refer to the following manpages for ACL-related information aclcheck(3), aclsort(3), chmod(1), getfacl(1), ls(1), setfacl(1), and uname(1).

- **The newfs -R Command Allows Reserving Swap Space Larger Than the Device**

The `newfs -R` command will reserve swap space greater than is available on the underlying device.

Product Licensing

Product keys are available on the License Key certificate. The certificate also includes the number of product licenses purchased. A single key enables product installation on the number and the type of systems for which you purchased the license. A key may enable the operation of more products than specified on the certificate, but you are legally limited to the number of product licenses purchased.

The VERITAS licensing commands are provided in a software package named `VRTSvlic`. You must install the `VRTSvlic` package for the licensing process to work.

VERITAS Licensing Commands

- `vxlicinst` - Licenses a VERITAS product already installed on a system.
- `vxlicrep`- View currently installed licenses.
- `vxlictest`- Retrieves features that are encoded in a license key along with their descriptions.

NOTE The `VRTSvlic` package replaces the functionality of the `VRTSlic` package that was used in earlier releases. The following table lists equivalency between licensing commands in `VRTSlic` and `VRTSvlic`:

Table 1-8 Equivalency Between `VRTSlic` and `VRTSvlic` Commands

VRTSlic command	VRTSvlic command	Description
`vxlicense -c`	`vxlicinst`	Installs a license.
`vxlicense -p`	`vxlicrep`	Reports installed licenses.
`vxlicense -t`	`vxlictest`	Tests a license.

NOTE For more information on new licensing commands in the `VRTSvlic` package, refer to the `vxlicinst(1), vxlicrep(1)` and `vxlictest(1)` manpages

Licensed Features

The following is a list of VERITAS File System features that require a license in addition to a VxFS license. These extended VxFS features are licensed only with VERITAS Storage Foundation products and are not available individually. See “Technical Support” on page 5 for contact information.

Table 1-9 VxFS Licensed Features

VxFS Extended Feature	License Key Feature Name
File Change Log	File Change Log
Cross-Platform Data Sharing	Cross-Platform Data Sharing
Extra Large File Systems	Extra-Big File Systems
Multi-Volume File Systems	Multi-Volume Support
Quality of Storage Service (QoSS)	Quality of Storage Service
Storage Checkpoints	VXCKPT
Quick I/O for Databases	VXFDD
Cluster File System	VXCFS