# VERITAS® File System
## Release Notes
### Release 3.2.6 - Binary

*READ THIS BEFORE INSTALLING THE SOFTWARE!*

# Introduction

This guide provides information on the VERITAS File System™ (VxFS®)
Release 3.2.6. This release is intended for Solaris 2.5.1 and Solaris 2.6 operating
systems. References in this document to VxFS 3.2 regarding new features,
compatibility, and software limitations apply to VxFS 3.2.6.

**Note:** Read this entire document before installing the VERITAS File System.

Packages on the installation media include the VERITAS File System software,
File System documentation, and the optional VERITAS Quick I/O™ for
Databases and VERITAS Accelerator *for NFS*. Topics include:

- Getting Help
- New Features
- Year 2000 Compliance
- Changes in VxFS Release 3.2.5
- Using VERITAS Quick I/O or VERITAS Accelerator for NFS
- End of Product Support
- Compatibility With Previous Versions of VxFS
- Installing VxFS
- Upgrading VxFS
- Documentation
- Known Software Limitations and Problems in VxFS Release 3.2.6
- Software Problems Fixed in This Release

- Using VxFS in FirstWatch and other HA Environments

- Using the Accelerator With FirstWatch

# Getting Help

For license information or information about VERITAS service packages, contact VERITAS Customer Support.

U.S. and Canadian Customers: 1-800-342-0652

International Customers: +1 (650) 335-8555

Fax: (650) 335-8428

Electronic mail: support@veritas.com

For additional information about VERITAS and VERITAS products, visit the website at:

```
www.veritas.com
```

## Licensing and Support From Sun Microsystems

When you buy the VERITAS File System through Sun Microsystems, you must also purchase a license kit from Sun for each package. For support and licensing information, refer directly to the license kits, *not* the contact information provided above and in the VERITAS File System documentation.

## New Features

VxFS Release 3.2 has the following new features:

### ▼ Access Control Lists (ACLs)

VxFS Release 3.2 supports POSIX access control lists (ACLs) to enhance system security. ACLs let system administrators set file access permissions for specified users and groups.

### ▼ Large File Support

In previous VxFS releases, files were limited to 2 GB. VxFS 3.2 has improved the file system inode layout which allows efficient handling of files up to two terabytes in size. VxFS supports large files only on Solaris 2.6, as Solaris 2.5.1 does not support large files.

### ▼ Online Manual Pages

Online manual pages are now available for the following VERITAS-specific commands:

- `cp_vxfs`(1)
- `cpio_vxfs`(1)
- `ls_vxfs`(1)
- `mv_vxfs`(1)

## Year 2000 Compliance

VERITAS File System release 3.2.6 was thoroughly tested to ensure year 2000 compliance. Visit the VERITAS web site at www.veritas.com for the complete Year 2000 Certification Statement.

## Changes in VxFS Release 3.2.5

For year 2000 compliance, changes were made to vxdump and vxrestore functionality. There were two modifications to vxdump.

- The vxdump -T *date* command can only do incremental backups—level-zero dumps are no longer allowed.

- If you incorrectly entered the date in the vxdump -T *date* command, it returned an error, but proceeded with the dump. It now aborts the dump and returns a more complete error and usage message. The specified date must contain both the 2-digit year and the 2-digit century.

## Using VERITAS Quick I/O or VERITAS Accelerator *for NFS*

The VERITAS File System has two optionally licensable add-on products: VERITAS Quick I/O™ for Databases and the VERITAS Accelerator *for NFS.* These products are described in the *VERITAS File System System Administrator's Guide*, however, they cannot be purchased through Sun Microsystems.

For information on these products, contact VERITAS Software by phone at 1-800-258-UNIX or 1-650-335-8000, or send email to vx-sales@veritas.com.

## End of Product Support

This is the last major release of the VERITAS File System that will support the volcopy utility. Scripts and applications using volcopy should be phased out before the next VxFS major release.

The VERITAS Quick I/O for Databases feature has two renamed commands:

- vxmkcdev was changed to qiomkfile

- vxfddstat was changed to qiostat

The next major release, VxFS 3.3, will be the last to support the original command names.

## Compatibility With Previous Versions of VxFS

**Note:**  VERITAS strongly recommends updgrading any previously installed VxFS 3.2.*x* or older File Systems to VxFS 3.2.6.

If you are using VERITAS Quick I/O for Databases and you are upgrading to VxFS 3.2.6, you must also upgrade Quick I/O to Version 1.1.4. Contact VERITAS Customer Support (see "Getting Help") for more information.

VxFS 3.2 implements a new disk layout called Version 4. To ensure the best performance, upgrade any Version 1 and Version 2 disk layouts to Version 4. You can do the upgrade on line using the vxupgrade command (see the vxupgrade(1M) manual page for details).

VxFS 3.2 supports Version 1 and 2 disk layouts, but the contents of intent logs created on previous versions of VxFS cannot be used by VxFS 3.2. This is not a problem for 1.*x* or 2.*x* file systems created by VxFS 3.2, however, the *first* time you mount an older 1.*x* or 2.*x* file system on VxFS 3.2 *and* a file system check is required, you must run an fsck -o full to repair it (see the fsck_vxfs(1M) manual page for details).

## Installing VxFS

See the *VERITAS File System Installation Guide* for complete instructions on how to install VxFS using the pkgadd command.

The VERITAS CD-ROM purchased from VERITAS contains the following file system packages:

- VRTSvxfs—VxFS software and online manual pages

- VRTSfsdoc—VxFS Documentation

- VRTSfdd—VERITAS Quick I/O for Databases

- VRTSvxld—VERITAS Accelerator *for NFS*

**Note:** VxFS is a licensed product; you must obtain a license key before installing VxFS. To obtain a license key, complete a License Key Request Form and fax it to VERITAS Customer Support (see the section entitled "Getting Help" for contact information). See the section on obtaining a license key in the *VERITAS File System Installation Guide* for additional information.

## Upgrading VxFS

See the *VERITAS File System Installation Guide* for instructions on how to upgrade VxFS.

**Note:** License keys valid for VxFS 2.3.*x* are also valid for VERITAS 3.2.6 File Systems.

## Documentation

The following documents accompany this VxFS release:

- *VERITAS File System Installation Guide*

- *VERITAS File System Quick Start Guide*

- *VERITAS File System System Administrator's Guide*

**Note:** These guides are provided as PostScript and PDF files.

The `VRTSvxfs` package contains manual pages for VxFS commands and utilities.

### Displaying Documentation Online

The VERITAS File System guides are provided on the CD-ROM under the `pkgs/VRTSfsdoc/root/opt/VRTSfsdoc` directory. When doing a `pkgadd` for `VRTSfsdoc`, the program gives you the choice of installing the PostScript, PDF, or both formats. See the *VERITAS File System Installation Guide* for `VRTSfsdoc` package installation information.

## PostScript Format

You can use the Solaris Image Tool (`/usr/openwin/bin/imagetool`) or another PostScript viewer to display the following VxFS guides in their PostScript format:

- *VERITAS File System Installation Guide.*

  After installing the `VRTSfsdoc` package, you can access this guide in the directory `/opt/VRTSfsdoc/install`.

- *VERITAS File System Quick Start Guide.*

  After installing the `VRTSfsdoc` package, you can access this guide in the directory `/opt/VRTSfsdoc/quick_start`.

- *VERITAS File System System Administrator's Guide.*

  After installing the `VRTSfsdoc` package, you can access this guide in the directory `/opt/VRTSfsdoc/sys_admin`.

For ease of reference, the name of each chapter in a guide having multiple chapters consists of the chapter number, followed by a descriptive name of the chapter.

## PDF Format

The online documentation described above is also provided in Adobe Portable Document Format (PDF). To view or print PDF documents, you must use the Adobe Acrobat Reader. You can use Acrobat Reader as a stand-alone application, or as a plug-in to your web browser.

## Printing **PostScript** Documentation

To print the PostScript versions, you need access to a PostScript printer or print facilities that allow you to print PostScript documents.

You can print the PostScript documentation in either of the following ways:

- Use the print option in your PostScript viewer to print one or more pages.

- Print entire chapters using the `lp` command and your PostScript printer.

A list of all files in sequence is in the `files_to_print` file in each of the `/opt/VRTSfsdoc/*` directories; you can use this file to print an entire guide. For example, you can print the System Administrator's guide by going to the directory `/opt/VRTSfsdoc/sys_admin` and entering the following command:

```
lp -d printer_name `cat files_to_print`
```

## Documentation Notes

The *VERITAS File System Installation Guide, VERITAS File System System Administrator's Guide,* and online manual pages were updated for the 3.2 release.

## Manual Pages

This release includes the following online man pages as part of the VRTSvxfs package.

### Section 1

- cp_vxfs
- cpio_vxfs
- getext
- ls_vxfs
- mv_vxfs
- setext

### Section 1M

- df_vxfs
- ff_vxfs
- fsadm_vxfs
- fscat_vxfs
- fsck_vxfs
- fsdb_vxfs
- fstyp_vxfs
- labelit_vxfs
- mkfs_vxfs
- mount_vxfs
- ncheck_vxfs
- volcopy_vxfs
- vxdump
- vxedquota
- vxquot

- vxquota
- vxquotaoff
- vxquotaon
- vxrepquota
- vxrestore
- vxtunefs
- vxupgrade

## Section 4

- fs_vxfs
- inode_vxfs
- tunefstab

## Section 7

- vxfsio

# Known Software Limitations and Problems in VxFS Release 3.2.6

▼ **Panics on Solaris 2.6 Due to High Kernel Stack Consumption**

Although stack consumption was reduced in VERITAS File System release 3.2.2 (see "Software Problems Fixed in VxFS 3.2.2," incident number 21605), to avoid possible panics, it's a good idea to increase the stack size for the `svc_run()` threads from the default of 8K (one page on sun4u systems) to 16K. To do this, add the following line to `/etc/system` and reboot:

```
set rpcmod:svc_run_stksize=0x4000
```

The `svc_run_stksize` tuneable is not available for Solaris releases prior to 2.6.

▼ `vxrestore` **Cannot Restore** `VSOCK` **vnode Type Objects**

VxFS 3.2.2 added support for `VSOCK` vnode type for Solaris 2.6, however, `vxrestore` does not restore `VSOCK` vnode type objects.

▼ **Doing a** `pkgadd` **After a** `pkgrm` **May Cause a System Crash**

On Solaris operating systems, loading kernel modules with similar symbol tables can cause a system crash. It is advisable that while upgrading packages you reboot the system after a `pkgrm` and before doing a `pkgadd` of `VRTSvxfs` or `VRTSfdd`.

▼ **Under Some Conditions,** `fsadm` **Cannot Truncate a Directory**

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

▼ **The** `qio_cache` **Feature is Not Supported**

The `qio_cache` parameter of `vxtunefs` to enable or disable file system caching is not supported this release. Use of this feature is not recommended, but it is allowed by the file system.

## ▼ A Change in the Method of Computing CUT Values May Cause Misleading Error Messages to Display.

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

If a Version 2 file system is mounted on a system running VxFS 3.2, and that file system is subsequently used on an earlier version of VxFS, then the following messages may display when performing a full `fsck`:

```
vxfs fsck: incorrect CUT entry for filest 1, fix? (ynq)
vxfs fsck: incorrect CUT entry for filest 999, fix? (ynq)
```

This is expected and does not indicate file system corruption. Answer `y` to the 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.

## ▼ Inode Limitation on File Systems Without Large File Support

To be able to create more than 8 million inodes, the file system must be created using the `largefiles` option of `mkfs` (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) man pages for details.

## ▼ Some Fields Not Displayed by `fstyp` Command

The `fstyp` `-v` option shows the super-block. For the Version 4 file system, some information is no longer in the super-block, so fields such as `nau`, `logstart`, or `logend` will display zeros. `nau` can be computed using the following formula:

```
(size + aulen - 1) / aulen
```

The above fields are displayed by `fstyp` `-v`. You can use `mkfs` `-m` *raw_device_file* to display many fields that are not part of super-block. See the `mkfs_vxfs`(1M) and `mkfs`(1M) man pages for more information.

# Software Problems Fixed in This Release

This section lists problems fixed since VxFS Release 3.2.5. The VERITAS incident numbers are in brackets.

[26728]     System panics occurred when mounting a file system read-only while its NFS exported read-write and being write accessed by NFS clients.

[26934]     In some configurations, VxFS exhibited poor performance while doing multi-threaded sequential reads.

[26913]     A problem in the Direct I/O code generated spurious errors and invoked an `fsck` on file systems with large files enabled. Discovered during internal testing; never reported in the field.

[26699]     A full `fsck` did not rebuild the damaged current usage table (CUT) correctly, which could lead to usable inodes being marked bad.

[26616]     The Direct I/O code miscalculated the I/O size on files with large extents (over $2^{32}$ bytes). Discovered during internal testing; never reported in the field.

[25972]     The `vxdump` utility was not reading date information from `/etc/dumpdates` correctly when doing an incremental dump on a snapshot file system.

[25911]     The `vxdump` and `vxrestore` utilities failed to connect to a remote non-UNIX host.

[25782]     There was a memory allocation problem when calls were made using the `VX_MEMNOWAIT` flag.

[25671]     Solaris generated error messages incorrectly identifying VxFS as the cause of system panics. Added code to ignore requests to sync during a system panic.

[25661]     Read errors occurred because the `fsadm -D` command could not process directories with extents larger than 64 K.

[25631]     The `fsadm -d` command was not sorting directory entries efficiently.

[25620]     ACLs were not correctly masking user and group permissions.

[25516]    The `vxdump` command dumped core when it encountered an unusual entry in the `vfstab` file, such as a missing mount point or special fields.

[25332]    Internal offset checking for large files was not working correctly over NFS.

[25254]    Resizing a file system to a multiple of 32 K blocks would fail because the allocation unit number was calculated incorrectly.

[25253]    Could not turn on quotas or mount with quotas after upgrading a VxFS file system from the Version 2 to the Version 4 disk layout.

[25177]    Panics occurred in `vx_vget()` during NFS operations on an unmounted file system.

[24955]    Quick I/O for Databases was not working on 64-bit Solaris systems.

[24947]    A VxFS file system would not mount with the Accelerator (`mount vxldlog=`) if the Accelerator device was not available.

[24589]    Errors occurred when users tried to create files when quotas were enabled but there was no user quota record in the quotas file. The user and group ID processing code was modified.

[24098]    When using Quick I/O or direct I/O, panics occurred in `vx_snap_read_blk()` doing a read from a snapshot. Fixed by doing the I/O against the snapshot device.

[23860]    A race condition occurred when the `vx_fset_inull_list()` routine was called during an unmount. Discovered during internal testing; never reported in the field.

[21707]    Sequential 8 K reads affected performance on VxFS 3.2.1.

[21282]    The `df -o s` command sometimes generated errors on large file systems when invoked on a block device.

## Software Problems Fixed in VxFS 3.2.5

[24887]    When `vxdump -u` was run on VxFS snapshot, it recorded an incorrect device name, using the primary file system device name instead of the snapshot.

[24827]    (Accelerator *for NFS* Version 1.0.4) Accelerator driver was overwriting the VxLD superblock with map code.

[24778]    The `vxdump -f` command could not handle `user@host` syntax when writing to a remote device.

[24769]    The `vxrestore` command failed to restore file systems created by `vxdump -T`. Changed code to allow only incremental backups. See "Changes in VxFS Release 3.2.5" on page 4 for more information.

[24725]    The `vxrepquota` utility displayed incorrect usage/limit information when the block limits were defined with values in the gigabyte range.

[24691]    (Accelerator *for NFS* Version 1.0.4) Accelerator device failed to mount, returning the error message:

```
Cannot ioctl (CFG_ATTACH) '/dev/xxx' errno=Device busy
```

See incident number 21773 for related information.

[24674]    Could not check (`fsck`) or mount a file system that had more than 8 million inodes. Changed `mount` and `fsck` code to support larger inode lists.

[24621]    A deadlock occurred in the kernel between transaction processing code (`putpage`) and locking code (`bmap`).

[24323]    Pages corrupted by the page locking code (`page_unlock`) was causing system panics.

[24249]    The `vxdump -T` command now fails and displays an error message if the date is formatted incorrectly. See "Changes in VxFS Release 3.2.5" on page 4 for more information.

[24085]    System panics occurred due to incorrect handling of ioctls; code did not distinguish between UFS vnodes and VxFS vnodes

[24072]    System would hang while executing an `ls -R` command.

[24014]    System hangs occurred due to incorrect allocation of indirect address extents.

[23848]    (Accelerator *for NFS* Version 1.0.4) Only four file system records would be reported by `vxld_stat`.

[23759]    (Accelerator *for NFS* Version 1.0.4) Accelerator occasionally panicked because private state information was written by a lower level driver to the wrong field (`b_private`).

[23579]    The HSM application would sometimes hang due to a delay in the signal processing code.

[23416]    The current usage table (CUT) was not properly maintained on version 2 disk layouts.

[23364]    A `vxdump` procedure aborted thirty minutes after asking for a new tape and no tape was supplied. The limit was removed so that the tape request now waits indefinitely.

[23318]    The `mount` and `fsck` commands now verify the file system size before successfully completing.

[23312]    If a named pipe or a stream file was attached to a VxFS file, the file could not be renamed and would generate an EBUSY error.

[22491]    System would hang due to a race condition between processes waiting on a system lock (`vx_rwsleep_rec_lock`).

[21773]    (Accelerator *for NFS* Version 1.0.4) Accelerator failed to create a log volume larger than two gigabytes. Program now limits the size of volumes created with `vxld_mklog` to two gigabytes and prints the error message:

```
vxld_mklog: Warning: Accelerator volume '/dev/xxx' is
too big (Using the first 2Gb)
```

[18791]    (Accelerator *for NFS* Version 1.0.4) Could not recover an Accelerator device after I/O errors in an associated Accelerator volume without doing a reboot. Accelerator now allows a volume marked BAD to be removed.

[18394]    The `vxdump` command had a default tape size limitation of 13.2 gigabytes. Limit was increased to 1.3 terabytes.

## Software Problems Fixed in VxFS 3.2.4

[23833]   Panics would occur if there was an I/O error at the same time VxFS was reading the replica `ilist` inode off disk.

[23749]   Growing a VxFS 3.2.3 file system with a Version 4 layout failed because the allocation unit (AU) summary or `EMAP` inodes were truncated to their original size.

[23744]   The `vxfs_ninode` parameter (specifying the number of in-core VxFS inodes) could be auto-tuned to a value smaller than `ncsize`, causing system hangs. `ncsize` is the number of directory name lookup cache (DNLC) entries.

[23613]   There were potential problems creating snapshots because flushes to disk were initiated before all transactions completed and the system frozen.

[23605]   System panics occurred when the kernel code implementing file system reorganization (`fsadm`(1M)) incorrectly set exclusion zones. Observed on Version 1 and 2 file systems. Never reported in the field.

[23603]   There was a possibility of a VxFS panic due do disk I/O errors in the flushing code path.

[23516]   (Quick I/O for Databases 1.1.2) Quick I/O (`VRTSqio` package) sometimes returned files instead of file devices.

[23440]   A deadlock in the buffer cache code caused by a locking hierarchy violation occurred under heavy system loads.

[23178]   System panics or reports of invalid directories occurred when the kernel code implementing file system reorganization (`fsadm`(1M)) failed to purge directory block buffers from the buffer cache after they were consolidated. Never reported in the field.

[23175]   (Re-opened from Release 3.2.3) The `vxfs_ninode` parameter tuning curves (specifying the number of in-core inodes) lead to system hangs or panics on Sun4d systems (such as SPARCserver 1000 or SPARCserver 2000) with more than one gigabyte of main memory. This typically occurred under significant file system loads (such as doing backups) that accessed many files.

          Two new kernel messages were added to VxFS:

```
NOTE: msgcnt x vxfs: mesg 068: ncsize parameter is
greater than 80% of the vxfs_ninode parameter;
increasing the value of vxfs:vxfs_ninode

WARNING: msgcnt x vxfs: mesg 069: memory usage
specified by the vxfs:vxfs_ninode and
vxfs:vx_bc_maxbufhwm parameters exceeds available
memory; the system may hang under heavy load
```

Message 068 displays when the value auto-tuned for the
vxfs_ninode parameter is less than 125% of the ncsize
parameter. This message occurs only if one of the system tuneable
parameters—ncsize, vxfs_ninode, maxusers, or
max_nprocs—is set manually in the file /etc/system.

To avoid system hangs, set vxfs_ninode to at least 125% of the
value of ncsize. The best way to do this is to adjust ncsize down,
rather than adjusting vxfs_ninode up.

Message 069 displays when the value of the system tuneable
parameters—vxfs_ninode and vx_bc_maxbufhwm—add up to a
value that is more than 66% of the kernel virtual address space or
more than 50% of the physical system memory. VxFS inodes require
approximately one kilobyte each, so both values can be treated as if
they are in units of one kilobyte. This message occurs only if system
tuneables were altered manually; it cannot occur because of auto-
tuning.

[23135]     After restoring a file system from a backup, the ls(1) command
            occasionally blocked for a short time on the restored directory while
            waiting for delayed extending writes to flush. Flushing information
            to disk no longer locks out ls.

[23126]     System would hang when the kernel code implementing file system
            reorganization (fsadm(1M)) entered a loop, consuming 100% of one
            processor's CPU time in the kernel. Observed only on Version 1 and
            2 file systems, but also possible on Version 4 file systems.

[22972]     System panics or memory corruption occurred when the kernel code
            implementing file system reorganization (fsadm(1M)) freed a data
            structure using the incorrect size. Never reported in the field.

[22665]     (Quick I/O for Databases 1.1.1) VxFS panics occurred when doing a
            backup with NetBackup.

[22269]     While running for several hours on very fragmented files on a
            Version 1 or Version 2 file system, the kernel code implementing file
            system reorganization (`fsadm`(1M)) overestimated the amount of
            transaction space required and started spinning in the kernel. Never
            reported in the field.

[21568]     System panicked due to a race condition that occurred when the
            `vx_inull_list()` routine was called during an unmount.

[21180]     The kernel code implementing file system reorganization
            (`fsadm`(1M)) copied data from old extents to new extents past the
            end of a file. This did not corrupt data, but reorganization attempts
            failed unnecessarily, and spurious data I/O error warnings (Message
            38) occurred during writes:

```
WARNING: msgcnt x: vxfs: mesg 038: vx_dataioerr -
file system file data write error
```

## Software Problems Fixed in VxFS 3.2.3

[23220]     Panics occurred when executing some DMAPI IOCTLs.

[23175]     Sun4d systems (such as SPARCserver 1000 or SPARCserver 2000)
            with more than 512 megabytes of memory would sometimes hang
            while accessing file systems with a large number of inodes due to
            kernel virtual memory exhaustion.

[23040]     In some rare circumstances, a race condition caused two processes to
            infinitely loop. This was most likely to occur if `vxfs_ninode` was
            insufficiently tuned by the user.

[22931]     Panics occurred when a VxFS file was used as a swap file and the
            extents allocated to the file were not a multiple of `pagesize`.

[22980]     VxFS did not reclaim additional space allocated to files until the
            inodes were inactivated. VxFS now reclaims space from inodes that
            are idle for 10 minutes and which have aggressively allocated
            additional space.

[22798]     Unmounting a snapshot file system left the underlying device open,
            so the snapshot volume could not be removed without opening and
            closing it again.

[22576]     Panics occurred when a VxFS file was used as a swap file.

[22474]     There was a deadlock in `readdir` code on very heavily used file systems.

[19710]     Affected Version 1 and 2 file systems, or Version 4 file systems upgraded from Version 1 or 2. File systems with more than a million inodes were taking too much time to mount. Code was modified to optimize mount times.

## Software Problems Fixed in VxFS 3.2.2

[22279]     DMAPI subsystem did not properly support largefiles configuration. Contact your HSM application vendor for details.

[22106]     `vxdump` and `ufsdump` produced a mutually incompatible record format in `/etc/dumpdates`.

[21923]     VxFS was experiencing performance problems over NFS. VxFS timing and locking conflicts were resolved to improve efficiency. Customers running NFS servers on `vxfs` will see significant performance gains.

[21849]     Added support for VSOCK vnode type for Solaris 2.6.

[21812]     In some cases VxFS 3.2.1.1 would return an error on write faults for user mappings. This would cause utilities such as `ld` to dump core.

[21605]     Panics were occurring in disk driver code because of higher stack consumption in VxFS. VxFS 3.2.2 has considerably reduced the stack size to prevent these panics.

## Software Problems Fixed in VxFS 3.2.1.1

[21427]     In systems with more than 5 gigabytes of memory, VxFS would panic while loading.

[21202]     In Version 1 file systems, the free inode count in the superblock was incorrect when the file system was unmounted by the reboot command prior to reboot.

[21147]     A race between threads doing file and record locking, and threads doing a snapshot file system mount or unmount, caused the file system to hang.

## Using VxFS in FirstWatch and other HA Environments

The VERITAS file system can be used in FirstWatch® and other High
Availability environments, much as a UFS file system can. One difference is
that since the vxfs driver is a loadable driver and the UFS driver is not, the
vxfs driver is not guaranteed to occupy the same position in each system's
virtual file system switch (vfssw) table.

To ensure reliable failover of a VxFS file system between hosts, the following
line should be added in the same position to each host's /etc/system file:

```
forceload: fs/vxfs
```

**Note:**  When using VxFS in a High Availability environment, make sure that
all systems in the cluster are running the same version of VxFS. Systems
running different versions of VxFS cannot failover.

## Using the Accelerator With FirstWatch

See the *VERITAS File System Installation Guide* for information on installing and
using the VERITAS Accelerator *for NFS* with FirstWatch.