# TruCluster Server

## Cluster Installation

Part Number: AA-RHGXD-TE

**June 2001**

**Product Version:** TruCluster Server Version 5.1A

**Operating System and Version:** Tru64 UNIX Version 5.1A

This manual describes how to install the TruCluster Server software on the Tru64 UNIX operating system.

**Compaq Computer Corporation**
**Houston, Texas**

# Contents

## 3 Install and Configure the Tru64 UNIX Operating System



## 4 Create a Single-Member Cluster



## 5 Add Members



## 6 Reinstalling Cluster Members

## Tables

# About This Manual

This manual describes how to:

- Prepare for cluster installation.
- Install TruCluster™ Server Version 5.1A software on the Compaq Tru64™ UNIX Version 5.1A operating system and create a new cluster.
- Add members to a cluster.
- Perform a rolling upgrade from TruCluster Server Version 5.1 to TruCluster Server Version 5.1A.
- Upgrade a TruCluster Production Server cluster or Available Server configuration to a TruCluster Server Version 5.1A cluster. Upgrade a TruCluster Memory Channel Software cluster to TruCluster Server Version 5.1A.

## Audience

This manual is for system administrators who install, configure, and administer the TruCluster Server product. The instructions in this manual assume that you are an experienced UNIX administrator who can configure and maintain hardware, operating systems, and networks.

## New and Changed Features

The following changes have been made to this manual since the Version 5.1 release:

- *Chapter 1* contains two new sections: *Section 1.1* describes new or changed product features that affect installation and upgrades, and *Section 1.3* provides an overview of the commands used to create and upgrade clusters.
- *Chapter 2* includes some LAN IP address information. (The new *Cluster LAN Interconnect* manual contains the information needed to configure the hardware and install TruCluster Server on a cluster that has a LAN interconnect.)

  All restrictions about using the Logical Storage Manager (LSM) for the clusterwide root (/) and member swap file systems have been removed.

  *Table 2–2* has increased the minimum requirements for member boot disk root and swap size from 128 MB to 256 MB.

*Section 2.4.5* now recommends that you have at least 50 MB free space on each member's boot partition when allocating disk space for rolling upgrades.

*Section 2.6* now includes a recommendation for setting the value of the `boot_osflags` console variable to `A` before booting a new member added with the `clu_add_member` command.

- *Section 3.5.2* mentions that the `clu_add_member` command now provides information about the new member's boot disk to help you locate the correct disk from the new member's console.
- *Chapter 5* is updated to include the disk location hints and the setting of the `boot_osflags` console variable to `A` before booting the new member.
- Because the supported upgrade path from TruCluster Server Version 5.0 is now a full installation, the chapter "Upgrading from TruCluster Server Version 5.0" has been removed; the description of installation configuration files is now in *Section 6.3*.
- *Chapter 7* describes how to use the rolling upgrade mechanism to install a New Hardware Delivery (NHD) kit on a Version 5.1A cluster. The chapter also includes tables that list the supported upgrade paths from Version 5.1 to Version 5.1A, and tables that list the differences between the Version 5.1 and Version 5.1A `clu_upgrade` commands. In addition, the chapter is reorganized to put the upgrade procedure at the beginning of the chapter.
- *Chapter 8* includes information about TruCluster Server Version 5.1A support for LAN hardware as the cluster interconnect and server-only support for UNIX File System (UFS) read/write support. It also includes a new section that describes an upgrade option for TruCluster Memory Channel Software clusters that have little or no shared storage.

## Organization

This manual is organized as follows:

| | |
|---|---|
| *Chapter 1* | Describes the types of installations and the contents of the TruCluster Server kit. |
| *Chapter 2* | Describes what you need to read and do before beginning the installation. |
| *Chapter 3* | Explains how to install and configure Tru64 UNIX on the system that will become the first cluster member. |
| *Chapter 4* | Provides instructions for running the `clu_create` command to create a single-member cluster. |
| *Chapter 5* | Provides instructions for running the `clu_add_member` command to add a member to the cluster. |

| | |
|---|---|
| *Chapter 6* | Provides instructions for reinstalling the first cluster member, and for reinstalling additional cluster members. |
| *Chapter 7* | Explains how to perform a rolling upgrade. |
| *Chapter 8* | Explains how to upgrade from TruCluster Software Version 1.5 or Version 1.6. |
| *Appendix A* | Contains checklists to use when gathering the information that you need to create a cluster. |
| *Appendix B* | Summarizes changes to system files that are a result of TruCluster Server software installation. |
| *Appendix C* | Contains sample installation logs for `clu_create`, `clu_add_member`, and `clu_upgrade`. |
| *Appendix D* | Contains examples of boot messages following the successful creation of a single-member cluster and the addition of a member to the cluster. |
| *Appendix E* | Provides manual storage information gathering and configuration procedures for those upgrading from TruCluster Version 1.5 or Version 1.6 products who are not using the migration scripts that are described in *Chapter 8*. |

## Related Documents

Consult the following TruCluster Server manuals for assistance in understanding, configuring, installing, and administering clusters:

- TruCluster Server *Software Product Description* (SPD) — The authoritative description of the TruCluster Server Version 5.1A product. You can find the latest version of the SPD at the following URL:

  **`http://www.tru64unix.compaq.com/docs/pub_page/spds.html`**

- *Cluster Technical Overview* — Introduces the TruCluster Server product and provides descriptions of the major subsystems.
- *Cluster Release Notes* — Provides a brief introduction to new features in TruCluster Server and describes known problems and workarounds.
- *Cluster Hardware Configuration* — Describes how to set up the systems that will become cluster members, and how to configure cluster shared storage.
- *Cluster LAN Interconnect* — Describes how to use LAN hardware as the cluster interconnect.
- *Cluster Administration* — Describes cluster-specific administration tasks.

- *Cluster Highly Available Applications* — Describes how to deploy existing applications in a TruCluster Server cluster and how to write cluster-aware applications.

The TruCluster Server documentation is available on the World Wide Web at the following URL:

`http://www.tru64unix.compaq.com/docs/pub_page/cluster_list.html`

In addition, have available the following manuals from the Tru64 UNIX Version 5.1A documentation set:

- *Installation Guide*
- *Installation Guide — Advanced Topics*
- *Release Notes*
- *System Administration*
- *Network Administration: Connections*
- *Network Administration: Services*
- *Security*

The Tru64 UNIX documentation is available on the World Wide Web at the following URL:

`http://www.tru64unix.compaq.com/docs/`

### Icons on Tru64 UNIX Printed Manuals

The printed version of the Tru64 UNIX documentation uses letter icons on the spines of the manuals to help specific audiences quickly find the manuals that meet their needs. (You can order the printed documentation from Compaq.) The following list describes this convention:

| | |
|---|---|
| G | Manuals for general users |
| S | Manuals for system and network administrators |
| P | Manuals for programmers |
| R | Manuals for reference page users |

Some manuals in the documentation help meet the needs of several audiences. For example, the information in some system manuals is also used by programmers. Keep this in mind when searching for information on specific topics.

The *Documentation Overview* provides information on all of the manuals in the Tru64 UNIX documentation set.

## Online Documentation

The TruCluster Server documentation set is available on the Tru64 UNIX Documentation CD-ROM and on the Tru64 UNIX Associated Products Volume 2 CD-ROM in a format that is readable with a Web browser (HTML format) or with the Adobe Acrobat Reader (PDF format). In addition, the reference pages are available in HTML format.

If you are working on an Alpha™ system, the operating system provides a Netscape browser for viewing the HTML documentation. If you want to view the documentation on a Windows PC or Macintosh, you will need a browser that supports Javascript 1.1. We recommend using Version 4.0 or higher of Netscape Navigator, Netscape Communicator, or Microsoft Internet Explorer.

Most cross-references are hot links that you can follow from book to book, from book to reference page, from reference page to book, and from reference page to reference page. The book or reference page to which you are referring opens in a separate window so that you can gather the information that you need and then easily return to the book or reference page that you were originally reading.

To read the HTML documentation using the Netscape browser:

1. Log in as the superuser.
2. Create a mount point for the cluster documentation:

   ```
   # mkdir /usr/share/doclib/online/clusters
   ```

3. Insert the CD-ROM into the CD-ROM drive and mount it.

   On a Tru64 UNIX Version 5.0 or later system:

   ```
   # mount -o rrip /dev/disk/cdrom<n>c \
     /usr/share/doclib/online/clusters
   ```

   On a Tru64 UNIX system prior to Version 5.0:

   ```
   # mount -r -t cdfs -o rrip /dev/rz<n>c \
     /usr/share/doclib/online/clusters
   ```

   In these commands, *n* is the unit number of your CD-ROM device.

4. After the CD-ROM is mounted, users can access the documentation with Netscape.

   From a terminal emulator window, start Netscape with the following command:

   ```
   % /usr/bin/X11/netscape&
   ```

   From the Common Desktop Environment (CDE) front panel, click on: Application Manager⇒Desktop_apps⇒Netscape

To view the PDF files, you must install Version 3.0 or higher of the Adobe Acrobat Reader. A suitable version of the Acrobat Reader for Tru64 UNIX, Windows PCs, Macintosh, and other platforms is included on the Tru64 UNIX Documentation CD-ROM. You can also obtain the latest version directly from the Adobe Systems Inc. Web site:

**http://www.adobe.com**

With the Acrobat Reader you can scroll through books, print selected sections or the entire book, and copy sections to the clipboard.

You can access the documentation in the TruCluster product area on the Associated Products Volume 2 CD-ROM on a Tru64 UNIX system, as well as on a variety of operating systems that use the ISO 9660 Level 1 CD-ROM standard.

To help you locate specific information in the TruCluster Server documentation, the Associated Products CD-ROM contains a copy of the AltaVista Search CD-ROM software with a complete index of the TruCluster Server HTML documents. (The Tru64 UNIX documentation CD-ROM includes the Master Index for the Tru64 UNIX manuals and the TruCluster Server manuals.) The AltaVista software runs on an *x*86-based PC with Windows 95, Windows 98, Windows 2000, or Windows NT Version 4.0.

**Note**

When documentation is accessed from the documentation CD-ROM or the Web, links to referenced documents will take you to those documents. These links to documents outside the TruCluster Server documentation set do not work when the TruCluster Server documentation is accessed from the Associated Products CD-ROM.

## Reader's Comments

Compaq welcomes any comments and suggestions you have on this and other Tru64 UNIX manuals.

You can send your comments in the following ways:

- Fax: 603-884-0120 Attn: UBPG Publications, ZKO3-3/Y32
- Internet electronic mail: `readers_comment@zk3.dec.com`

  A Reader's Comment form is located on your system in the following location:

  `/usr/doc/readers_comment.txt`

Please include the following information along with your comments:

- The full title of the manual and the order number. (The order number appears on the title page of printed and PDF versions of a manual.)
- The section numbers and page numbers of the information on which you are commenting.
- The version of Tru64 UNIX that you are using.
- If known, the type of processor that is running the Tru64 UNIX software.

The Tru64 UNIX Publications group cannot respond to system problems or technical support inquiries. Please address technical questions to your local system vendor or to the appropriate Compaq technical support office. Information provided with the software media explains how to send problem reports to Compaq.

## Conventions

This manual uses the following typographical conventions:

| | |
|---|---|
| `#` | A number sign represents the superuser prompt. |
| `% cat` | Boldface type in interactive examples indicates typed user input. |
| *file* | Italic (slanted) type indicates variable values, placeholders, and function argument names. |
| . . . | A vertical ellipsis indicates that a portion of an example that would normally be present is not shown. |
| `cat(1)` | A cross-reference to a reference page includes the appropriate section number in parentheses. For example, `cat(1)` indicates that you can find information on the `cat` command in Section 1 of the reference pages. |

# 1
# Introduction

This manual describes how to install and configure the TruCluster Server product.

Before you begin the installation, read this manual, paying close attention to the chapters that deal with your type of installation. Becoming familiar with the general sequence of installation steps can save time and prevent later problems.

**Important Notes**

As an alternative to Memory Channel, you can use local area network (LAN) hardware for the cluster interconnect. The new *Cluster LAN Interconnect* manual contains the information needed to configure the hardware and install TruCluster Server on a cluster that has a LAN interconnect. If you are using LAN hardware for the cluster interconnect, see that manual.

The procedures in this manual assume that each system's hardware and firmware are installed and configured as described in the *Cluster Hardware Configuration* manual. Do not begin the software installation until the hardware and firmware are installed and configured, and all storage and cluster interconnect hubs or switches are turned on.

Before you install TruCluster Server, read the *Cluster Release Notes*. If you are not familiar with the TruCluster Server architecture and terminology, read the *Cluster Technical Overview* manual.

This chapter provides the following information:

- New or changed installation features for Version 5.1A (Section 1.1)
- Installation types (Section 1.2)
- Installation commands (Section 1.3)
- Full installation overview (Section 1.4)
- License requirements (Section 1.5)
- TruCluster Server subsets (Section 1.6)

- Tru64 UNIX subsets (Section 1.7)
- General considerations (Section 1.8)

## 1.1 New or Changed Installation Features for Version 5.1A

The following list describes new or changed features for TruCluster Server Version 5.1A that affect installation or upgrade procedures:

- The minimum memory requirement for a system used in a cluster is computed by adding 64 MB to the current base operating system minimum requirement. (See Section 1.8 for more information.)
- You can configure local area network (LAN) hardware for use as the cluster interconnect. LAN hardware provides an alternative to using Memory Channel hardware. For TruCluster Server Version 5.1A, the *Cluster LAN Interconnect* manual contains the information needed to install TruCluster Server on a cluster that has a LAN interconnect. If you plan to use LAN hardware for the cluster interconnect, see that manual.
- Logical Storage Manager (LSM) support is extended to the root (`/`) and swap file systems. You can use LSM to mirror the clusterwide root (`/`), `/usr`, and `/var` file systems.
- The upgrade path from TruCluster Server Version 5.0 is a full installation of TruCluster Server Version 5.1A.
- You can use the rolling upgrade procedure to install a New Hardware Delivery (NHD) kit. (See Chapter 7 for more information.)
- A cluster member can mount a UNIX File System (UFS) file system read/write (for local use only). The file system is not accessible by other cluster members; it cannot be exported; there is no failover. This feature provides an additional data conversion path when migrating UFS file systems to Advanced File System (AdvFS) domains as part of an upgrade from TruCluster Software Version 1.5 or Version 1.6. (See Chapter 8 for more information.)
- There is an upgrade path for TruCluster Memory Channel Software clusters. (See Section 8.9 for more information.)

See the *Cluster Technical Overview* manual for the full list of new and changed features for TruCluster Server Version 5.1A.

## 1.2 Installation Types

There are two types of TruCluster Server installations:

**full installation** Perform a full installation to create a new TruCluster Server cluster, or when you cannot (or choose not to) upgrade an existing cluster to TruCluster Server

Version 5.1A. However, a full installation on an existing cluster will result in that cluster being unavailable during installation and application configuration.

**upgrade installation**

An upgrade installation preserves as much of the current configuration as possible. The type and version of your current product determine your upgrade path. Table 1–1 lists TruCluster products and their associated upgrade paths to TruCluster Server Version 5.1A.

**Table 1–1: Upgrade Paths to TruCluster Server Version 5.1A**

| Current Product | Recommended Procedure | Comments |
|---|---|---|
| TruCluster Server Version 5.1 | Rolling upgrade. | See Chapter 7. Use the procedure in that chapter to perform a rolling upgrade of the base operating system and cluster software. |
| TruCluster Server Version 5.0A | Rolling upgrade to Version 5.1 followed by a rolling upgrade to Version 5.1A. | Another option is to perform a full installation of Tru64 UNIX Version 5.1A and TruCluster Server Version 5.1A. (You can save the `.membern.cfg` configuration files for use when re-creating the cluster. See Section 6.3 for more information on configuration files.) |
| TruCluster Server Version 5.0 | Full installation of Tru64 UNIX Version 5.1A and TruCluster Server Version 5.1A | |
| TruCluster Production Server Software or TruCluster Available Server Software Version 1.5 or Version 1.6 | Full installation of Tru64 UNIX Version 5.1A and TruCluster Server Version 5.1A. | See Chapter 8. This chapter provides three upgrade options.[a] |

**Table 1–1: Upgrade Paths to TruCluster Server Version 5.1A (cont.)**

| Current Product | Recommended Procedure | Comments |
|---|---|---|
| Earlier versions of TruCluster Production Server Software or TruCluster Available Server Software | Full installation of Tru64 UNIX Version 5.1A and TruCluster Server Version 5.1A. | See Chapter 1 through Chapter 5. Another option is to upgrade to a Version 1.5 or 1.6 product and then use one of the upgrade options in Chapter 8. |
| TruCluster Memory Channel Software | Full installationof Tru64 UNIX Version 5.1A and TruCluster Server Version 5.1A. | See Chapter 8. Section 8.9 discusses upgrading this type of cluster to TruCluster Server Version 5.1A. |

[a] If you have an existing production-level cluster, an alternative to either a full installation or an upgrade is to create and configure a separate cluster, test all applications, and migrate data from the old cluster to the new cluster. Chapter 8 discusses this option.

## 1.3 Installation Commands

This section introduces the commands that create a cluster, add members to a cluster, remove members from a cluster, and perform a rolling upgrade:

`clu_create`

You run the `clu_create` command on a Tru64 UNIX system to create the first member of a cluster. See `clu_create`(8) and Chapter 4 for information on when and how to use the `clu_create` command.

`clu_add_member`

You run the `clu_add_member` command on a current member of a cluster to add a member to the cluster. See `clu_add_member`(8) and Chapter 5 for information on when and how to use the `clu_add_member` command.

`clu_delete_member`

You run the `clu_delete_member` command on a current member of a cluster to remove another member from the cluster. See `clu_delete_member`(8) and *Cluster Administration* for information on when and how to use the `clu_delete_member` command.

`clu_upgrade`

You run the `clu_upgrade` command on specific members of the current

cluster to perform a rolling upgrade. See `clu_upgrade`(8) and Chapter 7 for information on when and how to use the `clu_upgrade` command.

## 1.4 Full Installation Overview

The following list outlines the major steps needed to form a new TruCluster Server Version 5.1A cluster:

1. Using the information in the *Cluster Hardware Configuration* manual, configure system and storage hardware and firmware.
2. Selecting AdvFS file systems, install Tru64 UNIX on one or more disks. The disks are either private disks on the system that will become the first cluster member, or disks on a shared bus that the system can access.

   **Note**

   You install Tru64 UNIX on only one system. You do not have to install Tru64 UNIX on each system that will become a cluster member.

3. Configure the Tru64 UNIX system, including network and time services. Load and configure the applications you plan to use in the cluster.
4. Load the TruCluster Server license and subsets.
5. Run the `clu_create` command to create a boot disk for the first cluster member and to create and populate the clusterwide[1] root (`/`), `/usr`, and `/var` AdvFS file systems.
6. Halt the Tru64 UNIX system and boot the disk containing the first member's cluster boot partition. As the system boots, it forms a single-member cluster and mounts the clusterwide root (`/`), `/usr`, and `/var` file systems.
7. Log in as `root` and run the `clu_add_member` command to add members to the cluster. Boot each new member before adding the next.

## 1.5 License Requirements

There are no clusterwide licenses. You must install licenses on each cluster member.

[1] The term **clusterwide** refers to those file systems that must be accessible to all cluster members regardless of which members are up, and therefore should reside on storage that is directly accessible by all cluster members.

For the first member, after installing and licensing Tru64 UNIX, load and register a TruCluster Server license (TCS-UA) to create a single-member cluster. You can load the TruCluster Server subsets without registering the TruCluster Server license, but you cannot create a cluster until you load and register a TruCluster Server license.

Each time you add an additional member to the cluster, you must register a TruCluster Server license. After you boot the new member, you must also register a Tru64 UNIX license (because the new member will use the Tru64 UNIX clusterwide root (`/`), `/usr`, and `/var` file systems). In addition, you must register any additional required application licenses on that member.

**Note**

You can boot a system that does not have a TruCluster Server license. The system joins the cluster and boots to multiuser mode, but only `root` can log in (with a maximum of two users). The cluster application availability (CAA) daemon, `caad`, is not started. The system displays a license error message reminding you to load the license. This policy enforces license checks while making it possible to boot, license, and repair a system during an emergency.

## 1.6 TruCluster Server Subsets

Table 1–2 lists the TruCluster Server subsets.

**Table 1–2: TruCluster Server Subset Contents**

| Subset Name | Description | Contents |
|---|---|---|
| `TCRBASE520` | TruCluster Base Components | Installation checks and all mandatory cluster components. |
| `TCRMAN520` | TruCluster Reference Pages | Reference pages and examples. |
| `TCRMIGRATE520` | TruCluster Migration Components | Migration utilities for use when upgrading from a Version 1.5 or Version 1.6 TruCluster Production Server Software or TruCluster Available Server Software. |

Table 1–3 lists the approximate disk space requirements, in megabytes (MB), for each TruCluster Server Version 5.1A subset in the root (`/`), `/usr`, and `/var` file systems on the Tru64 UNIX system. (Table 2–2 provides the minimum and recommended size requirements for the disks needed to create a cluster.)

**Table 1–3: TruCluster Server Subset Sizes**

| Subset | Root (/) File System (MB) | /usr File System (MB) | /var File System (MB) | Total (MB) |
|---|---|---|---|---|
| TCRBASE520 | 0.4 | 38.0 | 6.5 | 44.9 |
| TCRMAN520 | 0 | 0.8 | 0 | 0.8 |
| TCRMIGRATE520 | 0 | 1.0 | 0 | 1.0 |
| **Total (MB)** | **0.4** | **39.8** | **6.5** | **46.7** |

## 1.7 Tru64 UNIX Subsets

When you install Tru64 UNIX, you must select AdvFS as the file system type for the root (/), /usr, and /var file systems.

Regardless of the types of systems in the cluster, we strongly recommend that, unless prohibited by site policy, you load all subsets when installing the Tru64 UNIX system. You might add different systems at a later date, or you might install an application that has a dependency on a subset you did not install.

**Note**

If your cluster will contain different types of systems, make sure to load the optional Tru64 UNIX subsets needed to support different hardware configurations. For example, because keyboards and graphics cards require specific subsets to work properly, load all keyboard and font subsets.

You can add additional software subsets later, but, because you are dealing with only one system, it is easier to install them before you create a cluster. The *Cluster Administration* manual explains how to install subsets in an existing cluster.

## 1.8 General Considerations

Note the following general installation considerations:

- If you build your own kernels, be aware that in a cluster /vmunix is a context-dependent symbolic link (CDSL):

  ```
  /vmunix -> cluster/members/{memb}/boot_partition/vmunix
  ```

  Treat a CDSL as you do any other symbolic link: remember that copying a file follows the link, but moving a file replaces the link. If you move (instead of copy) a kernel to /vmunix, you replace the symbolic link with the actual file.

The *Cluster Technical Overview* describes CDSLs; *Cluster Administration* provides information on using and repairing CDSLs.

- The base operation system sets a minimum requirement for the amount of memory required to install Tru64 UNIX. In a cluster, each member must have at least 64 MB more than this minimum requirement. For example, if the base operating system requires 128 MB of memory, each system used in a cluster must have at least 192 MB of memory.
- TruCluster Server supports the UNIX File System (UFS) as a read-only file system clusterwide. That is, a member can mount a UFS file system and other members of the cluster have read-only access to it. However, a member can mount a UFS file system read/write, but only that member has read/write access to the file system (local use only). No other cluster members can access that file system. There is no failover should that member go down.

# 2

# Preparation

This chapter describes what to read and what to do before installing the TruCluster Server software. Table 2–1 summarizes the preparation guidelines, and references sources of information that provide more detail.

**Table 2–1: Preparation Tasks**

| Task | See |
|---|---|
| Make copies of the information checklists. | Appendix A |
| Make sure you have the manuals and the product authorization keys (PAKs) you will need when installing Tru64 UNIX and TruCluster Server software. | Section 2.1 |
| Read the TruCluster Server *Cluster Release Notes*. | *Cluster Release Notes* |
| For all systems that will be in the cluster, check the hardware and firmware for installation readiness. (New versions of Tru64 UNIX and TruCluster Server usually require new versions of the AlphaServer™ SRM firmware.) | TruCluster Server *Software Product Description* (SPD) and the *Cluster Hardware Configuration* manual. |
| Decide which member IDs to assign. | Section 2.2 |
| Obtain required IP names and addresses. | Section 2.3 |
| Decide which disks and partitions to use to install Tru64 UNIX and TruCluster Server. Provide additional space for future rolling upgrades. | Section 2.4 |
| Decide how many votes to assign each potential cluster member and, if configuring, the quorum disk. | Section 2.5 and the *Cluster Administration* manual |
| Set console variables. | Section 2.6 |

## 2.1 Documentation and PAK Requirements

Before beginning the installation, have the following Version 5.1A documents available:

- *Cluster Hardware Configuration*
- *Cluster LAN Interconnect*
- *Cluster Release Notes*
- *Cluster Technical Overview*

- *Cluster Highly Available Applications*
- *Cluster Administration*
- Tru64 UNIX *Installation Guide*
- Tru64 UNIX *Installation Guide — Advanced Topics*
- Tru64 UNIX *Release Notes*
- Tru64 UNIX *System Administration*
- Tru64 UNIX *Network Administration: Connections*
- Tru64 UNIX *Network Administration: Services*
- Tru64 UNIX *Security* (if you are using Enhanced Security)
- For each member, a Tru64 UNIX product authorization key (PAK)
- For each member, a TruCluster Server PAK

## 2.2 Member IDs

Each cluster member has a unique member ID, whose value is an integer from 1 through 63 inclusive. The cluster software uses member IDs to identify each member of the cluster. When adding a member to the cluster, the installation programs offer the next available member ID as the default value. During the installation, you can accept the offered default value or enter any unused member ID.

For example, consider a two-node cluster consisting of members `pepicelli` and `polishham`. If, when installing the cluster software, the default member IDs are accepted, `pepicelli` is assigned member ID 1 and `polishham` is assigned member ID 2:

```
pepicelli      member ID = 1
polishham      member ID = 2
```

When a member is added to the cluster, its member ID is written to its `sysconfigtab` file as the value of the `memberid` variable in the `generic` subsystem.

## 2.3 IP Names and Addresses

You will need the following IP name and address information when creating a cluster and adding members:

- A cluster name and IP address, which are assigned to the default cluster alias (Section 2.3.1).
- For each cluster member, a host name and IP names and addresses for each external network interface (Section 2.3.2).

- For each cluster member, an IP name and address for that member's virtual cluster interconnect (Section 2.3.3).

**Notes**

A Memory Channel interconnect requires one IP name and address per member. The installation programs provide default IP names and addresses.

A LAN interconnect requires two IP names and addresses per member. The installation programs provide default IP names and addresses.

All cluster members must be configured to use either Memory Channel or LAN hardware. You cannot mix interconnect types within a cluster.

The Tru64 UNIX *Network Administration: Connections* manual provides guidelines on allocating IP addresses. You can also refer to RFC 1918, in which the Internet Assigned Numbers Authority (IANA) reserves the following blocks of IP address space for use by private internets:

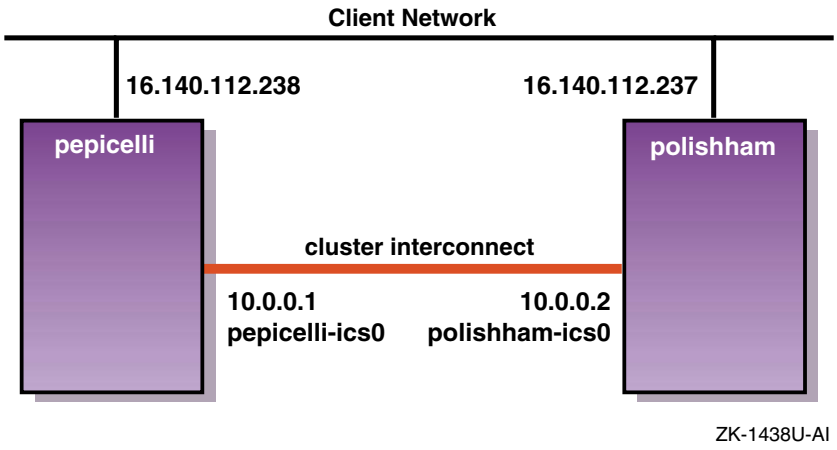
```
10.0.0.0      -   10.255.255.255
172.16.0.0    -   172.31.255.255
192.168.0.0   -   192.168.255.255
```

When the installation software offers default addresses for cluster interconnect networks, these default addresses are in class C networks (subnet mask 255.255.255.0).

For attributes that use host-name-like strings, such as member host names, the cluster name, and the name (or names) associated with each member's cluster interconnect, the installation programs require that you follow the standard host-name conventions defined in RFC 952, and amended in RFC 1123: use only the letters A-Z or a-z, numbers 0–9, '.' (period), and '-' (hyphen) as characters. The first character of the host name must be a letter or (per RFC 1123) a digit.

The examples in the following sections show the names and addresses used to create a sample cluster named `deli`. This cluster has two members, `pepicelli` and `polishham`, and uses Memory Channel hardware for its cluster interconnect. Figure 2–1 shows the network topology for cluster `deli`.

**Figure 2–1: Cluster Network Interfaces**




## 2.3.1 Cluster Name and IP Address

Each cluster has a cluster name, which is not the host name of one of the cluster members, but a host name that you assign to the entire cluster. This name is the host name associated with the default cluster alias, which is an IP address assigned to the entire cluster. It provides a method for external clients to request services from the cluster rather than from individual cluster members.

The following example shows the fully qualified cluster name and default cluster alias IP address for cluster `deli`:

```
16.140.112.209   deli.zk3.dec.com   deli   # default cluster alias IP address
                                           # cluster host name
                                           # (default cluster alias name)
```

The default cluster alias IP address must be a valid address to which clients can route. Several Internet services in the cluster use this address as the source address when connections originate from the cluster. If the address is not one to which clients can respond, the services will not work.

A cluster alias address should not be a broadcast address or a multicast address, nor should it reside in the subnet used by the cluster interconnect. In addition, although cluster members can use and advertise IPv6 addresses, IPv6 addresses are not supported by the cluster alias subsystem. Therefore, you cannot assign IPv6 addresses to cluster aliases.

You can assign a cluster alias an IP address that resides in one of the private address spaces defined in RFC 1918:

```
10.0.0.0      -   10.255.255.255   (10/8 prefix)
172.16.0.0    -   172.31.255.255   (172.16/12 prefix)
192.168.0.0   -   192.168.255.255  (192.168/16 prefix)
```

However, by default, the cluster alias daemon will not advertise an alias address that resides in these private address spaces. If you assign one of these addresses to the default cluster alias, do the following after creating a single-member cluster:

1. Set the value of the CLUAMGR_ROUTE_ARGS variable to resvok in the /etc/rc.config.common file:

   ```
   # rcmgr -c set CLUAMGR_ROUTE_ARGS resvok
   ```

2. Use the cluamgr command to notify the cluster alias daemon, aliasd, that it can advertise reserved addresses:

   ```
   # cluamgr -r resvok,start
   ```

After CLUAMGR_ROUTE_ARGS is set to resvok in /etc/rc.config.common, each cluster member automatically runs the cluamgr -r resvok command at boot time.

### 2.3.2 Member Host Names and IP Addresses

Each cluster member has a host name. By convention, this host name is usually associated with an IP address assigned to one of the system's network interfaces. Excluding the cluster interconnect interfaces, which are covered in Section 2.3.3, you will need the following names and IP addresses available for use by cluster members:

- For each member, a host name.

  You supply a host name when installing Tru64 UNIX on the system that will become the first cluster member, and when running clu_add_member to add members to the cluster.

- For each external network interface on each member, the IP name and IP address you will use when configuring that interface.

  You configure a network interface when installing Tru64 UNIX on the system that will become the first cluster member. (You can also configure other external network interfaces on that system before running clu_create. However, do not configure the cluster interconnect interface; clu_create will configure that interface.)

  The clu_add_member command configures only the cluster interconnect interface. During the first boot of a new member, you are given an opportunity to configure additional network interfaces. We recommend

that you configure at least one interface, whose IP name is the host name of the system.

For example, for the `deli` cluster members, each system has one external network interface. We use each member's host name as the IP name associated with that member's external interface's IP address:

```
16.140.112.238  pepicelli  # First  member's tu0 IP name and address.
                           # This is the Tru64 UNIX system.
                           # The host name is pepicelli.
16.140.112.237  polishham  # Second member's tu0 IP name and address.
                           # The host name is polishham.
```

### 2.3.3 IP Names and Addresses for Each Member's Cluster Interconnect

A cluster must have a dedicated **cluster interconnect**, which is a separate and distinct communications channel on which only cluster members reside. All cluster members must be connected to the cluster interconnect.

- For clusters that use a Memory Channel cluster interconnect, you need an IP name and an IP address for the virtual cluster interconnect interface on each member system. (Although a cluster member can have redundant physical interfaces for failover purposes, these interfaces share a single virtual network address.) Only one `IFCONFIG` and `NETDEV` entry represent a member's Memory Channel cluster interconnect in its `/etc/rc.config` file, regardless of the number of physical interfaces present in the system.

  By default, the installation programs offer an IP name set to the short form of the member's host name followed by `-ics0`, and IP addresses on the 10.0.0 subnet with the host portion of the address set to the member ID. For example, `pepicelli-ics0` and `10.0.0.1`.

  **Note**

  In previous releases, where Memory Channel was the only supported cluster interconnect device, `-mc0` was the identifier appended to the short form of a member's host name.

- For clusters that use a LAN cluster interconnect, you need two IP names and IP addresses to represent the cluster interconnect interface on each member system. The default location for the first address (`-ics0`) is on the 10.0.0 subnet. The default location for the second IP address (`-icstcp0`) is on the 10.1.0 subnet. See the *Cluster LAN Interconnect* manual for more information.

Cluster interconnect IP addresses must reside in a Class C network (because the installation programs use Class C network masks for these addresses).

However, because each cluster's interconnect IP addresses are private to that cluster, you can conserve address space by using the same subnet for several, or all, TruCluster Server clusters at your site. For example, each TruCluster Server cluster can use 10.0.0 as its cluster interconnect subnet; each cluster can have a member whose `-ics0` IP address is 10.0.0.1.

**Note**

Cluster interconnect IP addresses cannot end with either `.0` or `.255`. Addresses of this type are considered broadcast addresses. A system with this type of cluster interconnect IP address cannot join a cluster.

The following example shows the cluster interconnect IP names and addresses for two members of the `deli` cluster, `pepicelli` and `polishham`. This cluster uses a Memory Channel cluster interconnect.

```
10.0.0.1 pepicelli-ics0  # first member's  interconnect IP name and address
10.0.0.2 polishham-ics0  # second member's interconnect IP name and address
```

## 2.4 Disks

This section provides the following information:

- Disks needed for installation (Section 2.4.1)
- LSM considerations (Section 2.4.2)
- Minimum disk layout for a two-node cluster (Section 2.4.3)
- Disk space recommendations (Section 2.4.4)
- How to allocate additional disk space in preparation for a rolling upgrade (Section 2.4.5)

**Notes**

Because it is important that the cluster's root (`/`), `/usr`, and `/var` file systems are highly available, we recommend that you mirror these file systems. You can use either Redundant Array of Independent Disk (RAID) controllers or LSM (software RAID) to mirror these file systems. Because the Logical Storage Manager (LSM) is not supported for member boot partitions or the quorum disk, use RAID controllers (for example, an HSG80) to mirror those file systems.

If you plan to use Fibre Channel storagesets for any of the disks required for the base operating system or the cluster, read the Fibre Channel chapter in the *Cluster Hardware Configuration* manual.

Configure all storage hardware before installing the Tru64 UNIX operating system.

### 2.4.1 Disks Needed for Installation

You need to allocate disks, either physical or logical, for the following uses:

- One or more disks to hold the Tru64 UNIX operating system. The disks are either private disks on the system that will become the first cluster member, or disks on a shared bus that the system can access.
- One or more disks on a shared bus to hold the clusterwide AdvFS file systems: root (`/`), `/usr`, and, optionally, `/var`.

  Using separate disks provides better performance (more I/O in parallel). Separate disks also provide more load-balancing options. The *Cluster Administration* manual discusses balancing disk I/O.
- One boot disk per member on a shared bus. We recommend that member boot disks be on a shared bus for three reasons:

  1. When adding a member, the current member must have access to the new member's boot disk.
  2. If boot disks are on private buses and a member cannot boot, you cannot mount that disk from a running member and attempt to diagnose the problem.
  3. Some cluster administrative commands, like `clu_quorum`, might need access to a member's boot disk when the member is down.
- Depending on the number of members in the cluster, one disk on a shared bus to act as the quorum disk.

The examples in this section show each bootable disk from the point of view of a system console device (`DK`), and from the point of view of the operating system device special file (`dsk`). If you know the console device name for a bootable disk, put that information in the checklists in Appendix A as you go through this section. However, until you install Tru64 UNIX, you cannot know which special file is associated with each physical device. After you install Tru64 UNIX, use the information in Section 3.5 to map console device names to device special file names.

**Note**

If your storage configuration contains RAID array controllers, you can use RAID storage for the cluster-related disks. For example, you do not have to allocate a single physical disk for each cluster member's boot disk.

However, HSZ70 and HSZ80 controllers do not support partitioned storage sets or partitioned single-disk units in multiple-bus failover dual-redundant configurations. If you want to use dual SCSI buses with an HSZ70 or HSZ80 configured for multiple-bus failover, you must delete your partitions before configuring the controller for multiple-bus failover. With multiple-bus failover, you cannot use logical disks for boot partitions. You must use an entire physical disk for a member boot partition.

---

The following sections provide more information about the disks needed to install a cluster.

#### 2.4.1.1 Tru64 UNIX Disk (Private or Shared)

The base operating system is installed using AdvFS file systems on one or more physical or logical disks. These disks are either private disks on the system that will become the first cluster member, or disks on a shared bus that the system can access. The following example assumes that you install the base operating system on `DKA0` and that the operating system maps `DKA0` to `dsk0`:

```
DKA0        dsk0a       root_domain#root
            dsk0g       usr_domain#usr
            dsk0h       var_domain#var
```

Because the base operating system is still available after you create a cluster, in an emergency you have the option of booting the base operating system and attempting to diagnose and fix problems.

**Caution**

Because the base operating system has access to data on any shared buses, do not boot the base operating system while the cluster is running. The base operating system has no knowledge of the barriers and locks that the cluster uses to control multisystem access to shared data. You risk data corruption if the Tru64 UNIX system accesses a disk used by the cluster. Boot the base operating system only after shutting down the entire cluster.

---

**Restrictions:** A disk used for the Tru64 UNIX operating system cannot be used as a clusterwide disk, a member boot disk, or the quorum disk.

#### 2.4.1.2 Clusterwide root (/), /usr, and /var Disks (Shared)

When you create a cluster, the installation scripts copy the Tru64 UNIX root (/), /usr, and /var file systems from the Tru64 UNIX disks to the disks you specify for the clusterwide file systems.

The physical or logical disks used for the clusterwide file systems should be on a shared bus so that cluster members have access to the disks regardless of whether an individual cluster member is up or down. (If a file system that must be accessible to all cluster members is on a member's private bus, and if that member crashes, the remaining cluster members cannot access that file system.)

During the installation, you supply the names of the disk partitions that will contain the clusterwide root (/), /usr, and /var file systems. Each AdvFS file system must be in a separate partition; however, the partitions do not have to be on the same disk. Table 2–2 lists the minimum recommended sizes for all cluster file systems.

For example:

```
dsk1b       cluster_root#root
dsk2c       cluster_usr#usr
dsk3c       cluster_var#var
```

**Note**

In a cluster, /var cannot be in /usr. If you install Tru64 UNIX with /var in /usr, clu_create will create separate partitions for them.

The cluster_root#root file system is often put on a b partition for these reasons:

- Size: The b partition is usually larger than a.
- Nonbootable: No cluster member should boot the cluster_root#root file system. (Each member boots from a minimal root (/) file system on the a partition on its boot disk, not from the clusterwide root (/) file system.)

If, after you create and begin to use the cluster, you decide that you need extra space, you can add volumes to cluster_root#root, cluster_usr#usr, or cluster_var#var.

**Note**

The physical devices containing all volumes in AdvFS domains and all storage in LSM volumes for the file systems that all

cluster members must access should be on a bus or buses shared by all members.

---

**Restrictions:** If any partition on a disk is used by a clusterwide root (/), /usr, or /var file system, the disk cannot be used as a member boot disk or as the quorum disk.

#### 2.4.1.3 Member Boot Disks (Shared)

Each member has a boot disk, which can either be a physical or a logical disk. This disk should be on a bus shared by other cluster members. This disk must be on a bus that is accessible to the system that is adding the member to the cluster.

For example:

```
DKC400     dsk10     first  member's boot disk  [pepicelli]
DKC600     dsk12     second member's boot disk  [polishham]
```

By default, the installation scripts reformat each member's boot disk to contain three partitions:

- An a partition for that member's bootable root file system
- A b partition for swap
- An h partition (fstype of cnx) for cluster status information

There are no usr or var file systems on a member's boot disk.

**Note**

To avoid booting the wrong disk for a new member, you must know both the /dev/disk/dsk special file name for a member's boot disk and the DK device name for that disk from the new member's console.

When adding a member, the clu_add_member command displays information about a member boot disk. This information includes, if known, the disk's serial number (worldwide ID), manufacturer, model number, and physical location (bus/target/logical unit number (LUN)). Remember that the bus/target/LUN information is based on the system where you run clu_add_member. Depending on system and storage configuration, this information may or may not provide the same view of storage as the new member's console.

Use the information provided by clu_add_member and the information in Section 3.5 to map the special file name for a

member boot disk to the physical device. Put this information in the Member Attributes table in Appendix A (Table A–3).

---

A member disk can contain more than the three required partitions. However, those partitions should not contain data that you want to preserve. One use for a spare partition on a member boot disk might be to provide extra `/tmp` space for that member.

If a disk meets the following requirements, the installation scripts do not relabel the disk:

- The `a` partition meets the minimum size requirements in Table 2–2 and has an offset of `0`.
- The `b` partition meets the minimum size requirements in Table 2–2, the partition has an offset greater than the size of partition `a`, and the file system type is `swap`.
- The `h` partition is exactly 1 MB in size, has an offset greater than the size of `b` plus the offset of `b`, the end of the partition is contiguous with the end of the disk, and the file system type is `cnx`.

**Restrictions:** A member boot disk cannot contain any of the clusterwide root (`/`), `/usr`, or `/var` file systems. A member boot disk cannot be used as the quorum disk.

#### 2.4.1.4 Quorum Disk (Shared)

A quorum disk is a physical or logical disk whose `h` partition contains cluster status and quorum information. For example:

```
dsk7        quorum disk
```

A quorum disk helps to prevent cluster partitions, and also increases the availability of clusters. For a two-node cluster, we recommend that you configure a quorum disk, and give the quorum disk 1 vote. If each member and the quorum disk have 1 vote, the cluster can maintain quorum and continue operating as long as 2 votes are present. Therefore, the cluster can service client requests when both members are up, or when one member is up and the quorum disk is available. However, without a quorum disk, if one member fails, the cluster loses quorum and cannot service client requests.

Before deciding whether or not to configure a quorum disk, read Section 2.5 in this manual and the quorum disk information in the *Cluster Administration* manual. You can configure a quorum disk after creating a cluster.

**Note**

Because the quorum disk is accessed by cluster members during cluster transitions, no I/O barriers are placed on the quorum disk. For this reason, do not use any remaining partitions on this disk for data. If you specify a quorum disk during the installation or later with the `clu_quorum` command, do not specify a disk that contains data you want to preserve. A disk is relabeled when it becomes a quorum disk; any existing raw data or file systems are no longer accessible.

**Restrictions:** A cluster can have only one quorum disk. A quorum disk should be on a shared bus to which all cluster members are directly connected. If it is not, members that do not have a direct connection to the quorum disk may lose quorum before members that do have a direct connection to it. A member boot disk cannot be used as the quorum disk. A disk containing a clusterwide root (`/`), `/usr`, or `/var` file system cannot be used as the quorum disk.

### 2.4.2 LSM Considerations

If you plan to use LSM to mirror cluster file systems, be aware that mirroring file systems such as root (`/`) (`root_domain`), `/usr` (`usr_domain`), or `/var` (`var_domain`) on the Tru64 UNIX system has no influence on the cluster's `cluster_root`, `cluster_usr`, or `cluster_var` domains. For example, if you mirror the `root_domain` on the Tru64 UNIX system and then run `clu_create` to create a cluster, the `cluster_root` is not automatically mirrored. Therefore, if you do not plan to use LSM to mirror file systems on the Tru64 UNIX system but you do want to use LSM to mirror file systems on the TruCluster Server cluster, you can set up LSM on the Tru64 UNIX system but defer the mirroring of file systems until you boot the first member of the cluster.

After you create a single-member cluster with `clu_create`, you can use the `volmigrate` command to move the `cluster_root` domain to an LSM volume, and optionally mirror that volume. If you plan to use the `volmigrate` command to move `cluster_root`, be aware that `volmigrate` moves the data from the disk partition you specify to `clu_create` to new target storage that you must specify when you run `volmigrate`. (The disk partition you specify to `clu_create` is then no longer used for `cluster_root` and is available for other uses.) For this reason, you should think of the disk partition you specify to `clu_create` as temporary storage for the `cluster_root` domain, and plan for the additional storage required by `volmigrate`.

**Note**

You must use `volmigrate` for the `cluster_root` domain. You can use either `volmigrate` or `volencap` for the `cluster_usr` or `cluster_var` domains. When planning storage, be aware that `volmigrate` requires additional storage; `volencap` encapsulates the current partitions.

See the *Cluster Administration* manual for more information using LSM in a cluster. You might also want to read the `volmigrate`(8) and `volencap`(8) reference pages for information on using these commands in a cluster.

### 2.4.3 Minimum Disk Layout for a Two-Node Cluster

For a two-node cluster, the minimum disk layout uses at least four disks (although five is recommended):

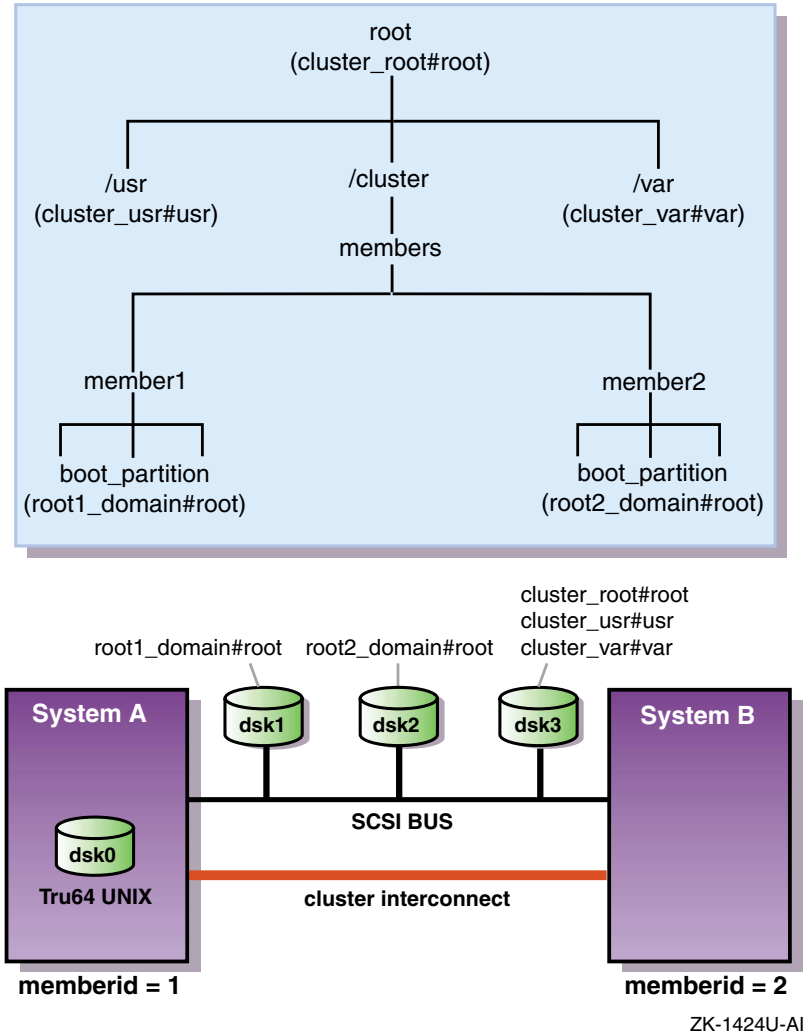
- One disk (private or shared) for the system that will become the first cluster member. This disk will hold the Tru64 UNIX Version 5.1A operating system.
- One disk on a shared bus for the clusterwide root (`/`), `/usr`, and `/var` file systems.
- Two disks on a shared bus for member boot partitions.
- Recommended: One disk on a shared bus for the quorum disk.

This minimum disk layout is not a no-single-point-of-failure (NSPOF) configuration. Although it puts all member boot disks on a shared bus, it does not mirror the clusterwide root (`/`), `/usr`, and `/var` file systems. We recommend that you mirror these critical file systems.

The *Cluster Hardware Configuration* manual describes NSPOF hardware configurations.

Figure 2–2 shows the relationship between the file systems these disks contain and the resulting cluster directory structure. The quorum disk is not shown because it does not contain a file system.

**Figure 2–2: Clusterwide Files and Member Boot Partitions**




## 2.4.4 Disk Space Recommendations

Table 2–2 provides the minimum recommended size requirements for disk partitions. These size requirements should be adequate for a cluster containing up to four members. Section 2.4.5 provides information on the disk space required to perform a rolling upgrade. Read that section as part of your disk space planning.

**Table 2–2: Disk Space Recommendations**

| File System (Type) | Partition | Recommended Minimum Size | Comment |
|---|---|---|---|
| Cluster root (/) (AdvFS) | b | 200 MB | The minimum partition size requirement is the larger of 125 MB or (1.125 x currently_used_root_size). For example, if the Tru64 UNIX root (/) file system currently uses 158 MB of its partition, the minimum partition size for the clusterwide root is 177.75 MB (1.125 x 158 = 177.75). The absolute minimum partition size requirement is 125 MB.<br>If you have the space, consider making the cluster root partition three times the size of the Tru64 UNIX root partition.<br>TruCluster Server supports adding volumes to the root file system. In a cluster, you can use the AdvFS addvol command to add volumes to cluster_root#root. (Note that the AdvFS Utilities require a separate license.) This lets you expand the root file system as the cluster grows. See the *Cluster Administration* manual for more information on adding volumes to AdvFS file domains. |
| Cluster /usr (AdvFS) | g | 1000 MB | The minimum partition size requirement is the larger of 675 MB or (1.125 x currently_used_/usr_size). The absolute minimum partition size is 675 MB. |
| Cluster /var (AdvFS) | h | 1000 MB | The minimum partition size requirement is the larger of 360 MB or (1.125 x currently_used_/var_size). The absolute minimum partition size is 360 MB. |
| Member boot disk: root (AdvFS) | a | 256 MB | The only required file systems on member boot disks are root, swap, and a 1 MB partition used for cluster status. For disks that contain no file systems, the installation procedure uses the default a partition size. For disks that contain file systems, the installation procedure calculates whether the sizes are acceptable. If the required partitions are usable, the installation procedure prompts whether you want to use the existing partition sizes. |

**Table 2–2: Disk Space Recommendations (cont.)**

| File System (Type) | Partition | Recommended Minimum Size | Comment |
|---|---|---|---|
| Member boot disk: `swap` (swap) | `b` | depends on system | The remainder of the member boot disk after the `a` and `h` partitions are allocated. The minimum size requirement is 256 MB.<br>To reduce traffic on shared storage buses, you can reconfigure swap after creating the cluster to put each member's primary swap partition on a private disk. See the *Cluster Administration* manual for information on reconfiguring swap space for a cluster member. (When configuring swap space for systems with large amounts of memory, make sure that the swap space is adequate. See the Tru64 UNIX *Installation Guide — Advanced Topics* manual for information on planning swap space.) |
| Member boot disk: CNX cluster status | `h` | exactly 1 MB | The `h` partition (`fstype cnx`) stores cluster state information. The installation procedure creates this partition on each member's boot disk. |
| Quorum disk | `h` | exactly 1 MB | You can use a small disk as the quorum disk. Because all cluster members must be able to retrieve the information from the quorum disk, no I/O barriers are placed on this disk. Therefore, do not put file systems on this disk. |

### 2.4.5 Allocating Additional Disk Space in Preparation for a Rolling Upgrade

A rolling upgrade, described in Chapter 7, has the following disk space requirements:

- At least 50 percent free space in root (`/`), `cluster_root#root`.
- At least 50 percent free space in `/usr`, `cluster_usr#usr`.
- At least 50 percent free space in `/var`, `cluster_var#var`, plus, if you are updating the operating system, an additional 425 MB to hold the subsets for the new version of the base operating system.
- If there is a separate `i18n` domain for the Worldwide Language Support (WLS) subsets, at least 50 percent free space in that domain.
- No tagged files are placed on member boot partitions. However, programs might need free space when moving kernels to boot partitions. We recommend that you reserve at least 50 MB free space on each member's boot partition.

**Note**

Starting with TruCluster Server Version 5.1A, the minimum disk size for a member root domain is 256 MB. The minimum disk size in earlier versions was 128 MB.

You have two options:

- Increase the size of the root (`/`), `/usr`, `/var`, and, if using a separate domain for the WLS subsets, `i18n` domains when creating the cluster.
- Use AdvFS management utilities like `addvol` or `dtadvfs` to add volumes to the existing domains before starting a rolling upgrade. (Note that the AdvFS Utilities require a separate license.)

**Note**

You cannot use the `addvol` command to add volumes to a member's root domain (the `a` partition on the member's boot disk). Instead, you must delete the member from the cluster, use `diskconfig` or SysMan to configure the disk appropriately, and then add the member back into the cluster.

## 2.5 Quorum: Member Votes and Expected Votes

Each system that has the potential to join a cluster has 0 or 1 votes. The number of votes assigned to a system is stored in its member-specific `/etc/sysconfigtab` file as the value of `cluster_node_votes`. Each system also has a value called expected votes stored in its `/etc/sysconfigtab` file as the value of `cluster_expected_votes`. In general, the value of `cluster_expected_votes` is the sum of the `cluster_node_votes` values for all potential members plus, if a quorum disk is configured, the votes assigned to the quorum disk.

The subject of member votes, quorum, and the forming and maintaining of a viable cluster is a complex topic that cannot be covered in a few paragraphs. The *Cluster Administration* manual describes how quorum is calculated and how cluster membership is maintained. Read that quorum information before deciding how many votes to assign to each potential cluster member during installation.

The following list describes the default node vote choices offered during installation by `clu_create` and `clu_add_member`:

**Note**

If you are not sure what to do for node votes during the installation, accept the default choices to ensure that the systems can form a cluster. You can always modify votes later.

- The first member always gets 1 node vote.

  Only members with votes can form a cluster; non-voting members can only join a cluster. Therefore, the first member must have a vote in order to form a single-member cluster.

- The default for the quorum disk is 1 vote. Legal values are 0 and 1. If you configure a quorum disk, give it 1 vote. (Do this unless you are creating a single-member cluster and do not plan to add another member for a while. A single-member cluster with a quorum disk can lose quorum if the quorum disk has a vote and the disk fails.) If your cluster will have two members, configure a quorum disk and give it 1 vote. Add the second member as soon as possible.

- The default node vote values offered for additional members are derived from the expected votes setting on the member where you run `clu_add_member`. If the value of expected votes is 1, the default offered by `clu_add_member` is 0. If the value of expected votes is greater than 1, the default offered by `clu_add_member` is 1. Legal values are 0 and 1.

  If you plan to create a two-member cluster with a quorum disk, give 1 vote to each member and 1 vote to the quorum disk.

In the sample `deli` cluster, described in Section 2.3, the first member (`pepicelli`) was automatically assigned 1 vote, a quorum disk was configured (1 vote), and the second member (`polishham`) was assigned 1 vote during installation. Therefore, the value of `cluster_expected_votes` is 3, and 2 votes are required for quorum.

The default vote values offered by `clu_create` and `clu_add_member` are biased towards the optimal availability of a two-member cluster (two voting members and a quorum disk). To foster availability in other cluster configurations, take into account the suggestions in Table 2–3 and Table 2–4. We strongly recommend the use of a quorum disk to increase availability for clusters with an even number of members (2, 4, 6, etc.). This is especially true for small clusters. Nevertheless, to help you in those situations in which a quorum disk is not an option, Table 2–4 lists the optimal vote assignments for clusters in which a quorum disk is not configured.

**Table 2–3: Recommended Vote Assignments for a Cluster with a Quorum Disk**

| Number of Members | Quorum Disk Votes | Node Vote Assignments (nodes in increasing member ID order, separated by commas) |
|---|---|---|
| 1 | 0 | 1 |
| 2 | 1 | 1,1 |
| 3 | 0 | 1,1,1 |
| 4 | 1 | 1,1,1,1 |
| 5 | 0 | 1,1,1,1,1 |
| 6 | 1[a] | 1,1,1,1,1,1 |

[a] A voting quorum disk increases the availability of a cluster with an even number of members. However, when a cluster has more than four members, the increase in availability resulting from a voting quorum disk is relatively small.

**Table 2–4: Recommended Vote Assignments for a Cluster Without a Quorum Disk**

| Number of Members | Node Vote Assignments (nodes in increasing member ID order, separated by commas) |
|---|---|
| 1 | 1 |
| 2 | 1,0 |
| 3 through 8 | 1, ..., 1 |

## 2.6 Console Variables

The following list contains console variables whose settings are important for systems in a cluster. The list indicates whether you have to set the variable, or whether `clu_create` or `clu_add_member` sets the variable.

`boot_osflags`

**`set boot_osflags A`**

If `boot_osflags` is set to `A`, the system automatically boots to multiuser mode. Tru64 UNIX automatically sets this variable when you install the system that will become the first member of the cluster. When adding members, set this variable before booting the member for the first time.

`boot_reset`

**`set boot_reset on`**

If the value boot_reset is off, only a warm boot is performed on a system halt or boot command; if the value is on, a full reset (cold boot) is performed. Setting the value to on brings the system to a known state at each boot.

The clu_create command automatically sets boot_reset to on for the first member during cluster creation. The clu_add_member command creates a one-time-only script that sets boot_reset to on during the first boot of the new member.

You must manually set this variable for the AlphaServer 8200 and 8400 systems.

bootdef_dev

**set bootdef_dev *boot_disk***

The bootdef_dev variable specifies an ordered list of devices (separated by commas with no spaces) from which the system tries to boot if the boot command is entered without parameters, or if an AUTO_ACTION boot is in progress.

The clu_create command automatically sets the boot device for the first member during cluster creation. The clu_add_member command creates a one-time-only script that sets the boot device during the first boot of the new member.

You must manually set this variable for the AlphaServer 8200 and 8400 systems.

If your hardware configuration includes HS controllers that are connected to dual SCSI or Fibre Channel buses and configured for multiple-bus failover, clu_create and clu_add_member can set only one bus path. You must halt the system and manually set both values at the console before booting. For example:

```
>>> set bootdef_dev device1,device2
```

Separate each device with a comma; do not use any spaces. If the console is unable to boot from the first device, it will try the next device, and so on. If all paths to the device are not specified, you might see the console error message not connected when attempting to boot the member.

**Note**

If a system is using a Fibre Channel disk as its cluster member boot disk, neither clu_create nor clu_add_member can set the bootdef_dev variable unless you used either the wwidmgr set or wwidmgr quickset console command to configure device and port path

information. Even so, clu_create and clu_add_member will set only one path. See the Fibre Channel chapter in the *Cluster Hardware Configuration* manual for a procedure to set multiple paths for the bootdef_dev variable when a member boot disk is accessed via Fibre Channel.

---

boot_dev

**set boot_dev *boot_disk***

The boot_dev variable determines which disk is used for a reboot.

The clu_create command automatically creates and sets this variable for the first member during cluster creation. The clu_add_member command creates a one-time-only script that creates and sets this variable during the first boot of a new member.

You must manually set this variable for the AlphaServer 8200 and 8400 systems.

bus_probe_algorithm

**set bus_probe_algorithm new**

For systems that support the bus_probe_algorithm console variable, you must set this variable to new on the first system before installing Tru64 UNIX, and on each additional member before booting it for the first time. Setting the value of bus_probe_algorithm to new ensures that peripheral component interconnect (PCI) devices are consistently probed on all member systems.

The following AlphaServer systems support the bus_probe_algorithm console variable: 800, 1000, 1000A, 2000, 2100, and 2100A. Newer systems do not use this variable.

To check and set the bus_probe_algorithm console variable, follow these steps:

1. Shut the system down to console mode and power cycle the system. This clears any transient console variables.
2. Enter the show command:

   ```
   >>> show bus_probe_algorithm
   ```

   If the variable is supported, the console will print its name and its current value (either old or new).
3. If the firmware supports the bus_probe_algorithm console variable and its current value is old, set the value to new:

   ```
   >>> set bus_probe_algorithm new
   ```

## 2.7 Sample Cluster Configuration Checklists

Table 2–5, Table 2–6, and Table 2–7 use the checklists in Appendix A to summarize the information used for the sample `deli` cluster configuration.

**Table 2–5: Sample Tru64 UNIX System Attributes**

| Attribute | Value |
|---|---|
| Host name | `pepicelli.zk3.dec.com` |
| Host name IP address | `16.140.112.238` |
| Tru64 UNIX root partition (for example, `dsk0a`) | `dsk0a` |
| Tru64 UNIX root device from console (for example, `DKA0`) | `DKA0` |
| Tru64 UNIX `/usr` partition (for example, `dsk0g`) | `dsk0g` |
| Tru64 UNIX `/var` partition (for example, `dsk0h`) | `dsk0h` |

**Table 2–6: Sample Cluster Attributes**

| Attribute | Value |
|---|---|
| Cluster name (fully qualified) | `deli.zk3.dec.com` |
| Default cluster alias IP address | `16.140.112.209` |
| Clusterwide root partition (for example, `dsk1b`) | `dsk1b` |
| Clusterwide root (`/`) disk device serial number (for example, WWID) | `DEC RZ1CF-CF (C) DEC 50022303` |
| Clusterwide `/usr` partition (for example, `dsk2c`) | `dsk2c` |
| Clusterwide `/usr` disk device serial number (for example, WWID) | `DEC RZ1CB-CS (C) DECQD2202330Y3WJL` |
| Clusterwide `/var` partition (for example, `dsk3c`) | `dsk3c` |
| Clusterwide `/var` disk device serial number (for example, WWID) | `DEC RZ1CF-CF (C) DEC 50021480` |
| If using a quorum disk, the disk device (for example, `dsk7`) | `dsk7` |
| If using a quorum disk, quorum disk device serial number (for example, WWID) | `DEC RZ28L-AS (C) DEC-JED716250N6TF6` |
| If using a quorum disk, the number of votes assigned to the quorum disk | `1` |

**Table 2–7: Sample Member Attributes**

| Attribute | 1st Member | 2nd Member | 3rd Member |
|---|---|---|---|
| Member host name (fully qualified) | `pepi-celli.zk3.dec.com` | `polish-ham.zk3.dec.com` | |
| Host name IP address | `16.140.112.238` | `16.140.112.237` | |
| Member ID (`memberid`: 1-63) | `1` | `2` | |

**Table 2–7: Sample Member Attributes (cont.)**

| Attribute | 1st Member | 2nd Member | 3rd Member |
|---|---|---|---|
| Number of votes assigned to this member | 1 | 1 | |
| Boot disk (for example, dsk10) | dsk10 | dsk12 | |
| Boot device from console (for example, DKC400) | DKC400 | DKC600 | |
| Boot device serial number (for example, WWID) | DEC RZ1CF-CF (C) DEC 50066053 | DEC RZ1CF-CF (C) DEC 50066104 | |
| Boot device physical location (bus/target/LUN) | bus-2-targ-11-lun-0 | bus-2-targ-13-lun-0 | |
| Virtual cluster interconnect IP name | pepicelli-ics0 | polishham-ics0 | |
| Virtual cluster interconnect IP address | 10.0.0.1 | 10.0.0.2 | |
| Physical cluster interconnect device name | Memory Channel | | |
| Physical cluster interconnect IP address (LAN only) | n/a | n/a | |
| Additional network interface IP name | n/a | n/a | |
| Additional network interface IP address | n/a | n/a | |

# 3

# Install and Configure the Tru64 UNIX Operating System

When creating a cluster, you do not need to install the Tru64 UNIX operating system on all the systems that will become cluster members. You install the Tru64 UNIX operating system only on the system that will become the first cluster member. This system will, in essence, be cloned to create the first cluster member.

You need one Tru64 UNIX system in order to run the `clu_create` command to configure that system as the first cluster member. After you boot this first cluster member, you run `clu_add_member` to create boot disks for additional members. Those members do not need individual copies of the base operating system (although they do need individual Tru64 UNIX licenses).

If your cluster will contain different types of systems, load the optional Tru64 UNIX subsets needed to support different hardware configurations. For example, because keyboards and graphics cards require specific subsets in order to work properly, load all keyboard and font subsets. You can install them later, but, if you have enough disk space on the Tru64 UNIX system, we strongly recommend that, unless prohibited by site policy, you load all subsets when installing the Tru64 UNIX operating system.

Configure the Tru64 UNIX system fully before creating a cluster. This lets you verify your configuration before creating a cluster.

In addition to configuring the base operating system, load and configure the layered products and applications that you want available to the cluster.

If you plan to use Fibre Channel storagesets for the base operating system disks, read the Fibre Channel chapter in the *Cluster Hardware Configuration* manual.

Table 3–1 lists the installation tasks in order and references sources of necessary information.

**Notes**

If you are performing a rolling upgrade of a cluster, go to Chapter 7 and follow the directions in that chapter.

If you are upgrading to TruCluster Server Version 5.1A from TruCluster Production Server Software or Available Server Software Version 1.5 or Version 1.6, go to Chapter 8 and follow the directions in that chapter.

If you are upgrading to TruCluster Server Version 5.1A from TruCluster Memory Channel Software, go to Section 8.9 and follow the directions in that section.

---

**Table 3–1: Installing Tru64 UNIX**

| Task | See |
|---|---|
| Make sure that all storage is properly installed and configured (for example, if you are using one or more HS controllers, make sure that RAID sets and units are configured). | *Cluster Hardware Configuration* manual |
| Examine console variables. | Section 2.6 |
| Update SRM firmware. | Section 3.1 |
| Install the Tru64 UNIX operating system. | Section 3.2 and the Tru64 UNIX *Installation Guide* |
| Configure basic services. | Section 3.3 and the Tru64 UNIX *Network Administration: Services* manual |
| Configure enhanced security (optional). | Section 3.4 and the Tru64 UNIX *Security* manual |
| Configure the disks needed for cluster installation. | Section 2.4, Section 3.5, and the Tru64 UNIX *System Administration* manual |

## 3.1 Update SRM Firmware

New versions of Tru64 UNIX and TruCluster Server usually require new versions of the AlphaServer SRM firmware. Firmware updates are located on the Alpha Systems Firmware CD-ROM, which is included in the base operating system Software Distribution Kit. To determine whether you need to update firmware, see the TruCluster Server *Software Product Description* (SPD) and the firmware release notes for each type of system in the cluster. Update firmware as needed before installing software. You can find the latest version of the TruCluster Server SPD at the following URL:

**http://www.tru64unix.compaq.com/docs/pub_page/spds.html**

## 3.2 Install the Tru64 UNIX Operating System

**Note**

The cluster installation copies the Tru64 UNIX root (/), /usr, and /var file systems to create the clusterwide root (/), /usr, and /var file systems. Therefore, we recommend that you fully configure the Tru64 UNIX system before creating a cluster.

Before performing the installation procedures described in the Tru64 UNIX *Installation Guide*, read the following list and incorporate these tasks into the installation:

1. Make sure that all storage devices are turned on.
2. Make sure that any hubs or switches used by the cluster interconnect are turned on.
3. Install the Tru64 UNIX operating system on one or more disks. The disks are either private disks on the system that will become the first cluster member, or disks on a shared bus that the system can access.

**Note**

We recommend that you load all subsets when installing the Tru64 UNIX operating system.

4. Use Advanced File System (AdvFS) file systems.
5. If installing Worldwide Language Support (WLS) in a separate file system, we recommend that the disk containing this file system is on a bus shared by all cluster members. See the Tru64 UNIX *Installation Guide — Advanced Topics* manual for information on installing WLS.
6. If you plan to use the Logical Storage Manager (LSM) to mirror file systems on the cluster, configure LSM on the Tru64 UNIX system. Put the root disk group (rootdg) on a disk on a shared bus. (Also see Section 2.4.2 in this manual and read the chapter that describes how to configure LSM for use in a cluster in the TruCluster Server *Cluster Administration* manual.)
7. If you are installing a patch kit as part of the base operating system installation, load the TruCluster Server subsets before installing the patch kit. If the TruCluster Server kit is not loaded before the patch operation, patches for TruCluster Server software will not be loaded.

The sequence of events when patching the initial installation of Tru64 UNIX are as follows:

a. Install and configure the Tru64 UNIX operating system.

b. Use the `setld` command to install the TruCluster Server kit.

c. Patch the system.

d. Use the `clu_create` command to create the single-member cluster.

8. If you add new hardware (for example, additional network adapters) after you install or update the Tru64 UNIX operating system, remember to boot `/genvmunix` and build a customized kernel. Otherwise, the system's kernel configuration file will not contain these hardware options, and the kernel you build during TruCluster Server installation will not recognize the new hardware. The Tru64 UNIX *System Administration* manual provides information on configuring kernels.

9. This step applies only to systems connected to Asynchronous Transport Mode (ATM) networks. To configure support for ATM LAN Emulation (LANE), select the necessary options from the list displayed by `doconfig`. In the following partial list of `doconfig` options, the options required for LANE support are marked with an asterisk (*):

```
  IP Switching over ATM (ATMIFMP)
* LAN Emulation over ATM (LANE)
  Classical IP over ATM (ATMIP)
* ATM UNI 3.0/3.1 Signalling for SVCs
```

## 3.3 Configure Basic Services

Using the information in the Tru64 UNIX *Network Administration: Connections* manual, run the `netconfig` utility or SysMan Menu and configure the system's standard network interfaces.

**Note**

Do not configure the interfaces for the cluster interconnect at this time; you will configure those interfaces when creating a cluster.

Configure the following basic services:

- A routing daemon: `gated` is required for cluster alias (Section 3.3.1).
- A time service: The Network Time Protocol (NTP) is recommended (Section 3.3.2).
- A name server, for example, the Berkeley Internet Name Domain (BIND) server (Section 3.3.3).

- If the cluster will be a file server, configure the Network File System (NFS) (Section 3.3.4).
- If the cluster will be a Network Information Service (NIS) master, slave, or client, configure NIS (Section 3.3.5).
- If the cluster will be a Dynamic Host Configuration Protocol (DHCP) server, configure DHCP (Section 3.3.6).
- If the cluster will be a mail server, configure mail (Section 3.3.7).
- If the cluster will be a print server, configure printing (Section 3.3.8).

Read the following sections and incorporate that information when configuring services.

**Notes**

If you choose not to configure one or more of these services before you create a cluster, see the *Cluster Administration* manual for information on how to configure these services in a running cluster. However, it is easier to configure the services before creating a cluster.

Section 3.3.9 summarizes the preferred network configuration for the Tru64 UNIX system before beginning the TruCluster Server installation.

### 3.3.1 Routing Daemon

The cluster alias software is designed to work with the `gated` routing daemon. Each cluster member's alias daemon, `aliasd`, creates a `/etc/gated.conf.membern` file for that member.

Do not use the `ogated` or `routed` daemons.

**Note**

The following routing information is provided only for experienced network administrators.

The `aliasd` daemon supports only the Routing Information Protocol (RIP). The daemon creates a modified version of `gated.conf` (`/etc/gated.conf.membern`) for each member using `rip` routes.

If you have a customized routing environment, or need to use a routing protocol such as Open Shortest Path First (OSPF) with the cluster alias, you can start alias routing on each cluster member with the `nogated`

option, which directs `aliasd` to not restart `gated` with the modified version of `gated.conf`. (After you create each cluster member, you are responsible for correctly merging the cluster alias information from the `/etc/gated.conf.membern` file into that member's `gated.conf` configuration file.)

To start alias routing with the `nogated` option, set `CLUAMGR_ROUTE_ARGS` to `nogated` in the Tru64 UNIX `/etc/rc.config.common` or `/etc/rc.config` file. When a cluster member boots with `CLUAMGR_ROUTE_ARGS=nogated`, the alias daemon starts (RIP packets used), but within a few seconds the `nogated` option takes effect. (After you install the TruCluster Server software, read the `nogated` section in `cluamgr`(8).)

In the following example, the first `rcmgr` command sets the variable in `/etc/rc.config.common`, the second in `/etc/rc.config`:

```
# rcmgr -c set CLUAMGR_ROUTE_ARGS nogated
# rcmgr set CLUAMGR_ROUTE_ARGS nogated
```

**Note**

In a cluster, you normally use `/etc/rc.config.common` when setting a variable for use by all cluster members, and a member-specific `rc.config` file when setting a variable for use by one member. However, when a variable is set in the `/etc/rc.config` file on the Tru64 UNIX system before creating a cluster, each cluster member will have that variable set in its member-specific `rc.config` file.

### 3.3.2 Time Server

Running a distributed time service provides clusterwide consistency for time stamps used by the file systems and applications. We recommend that you configure a distributed time service such as the Network Time Protocol (NTP) daemon (`xntpd`). NTP provides highly accurate synchronization and tracks the reliability of time sources. For information on NTP, see the Tru64 UNIX *Network Administration: Services* manual and `ntp_intro`(7).

If system times are not synchronized, any checks that rely on accurate time stamps will fail. If your site does not use NTP, make sure that whatever time service you use meets the granularity specifications defined in RFC 1035 *Network Time Protocol (Version 3) Specification, Implementation and Analysis*.

Because the system times of cluster members should not vary by more than a few seconds, we do not recommend using the `timed` daemon to set the time.

Generally, there is some system in your environment that is considered most informed as far as time is concerned. This system may be getting time from some other system that is considered a reliable time source. If there are no time servers on your network and you plan to use the cluster as a time server, see the Tru64 UNIX *Network Administration: Services* manual and `ntp_manual_setup`(7). Configure the Tru64 UNIX system as a time server before creating the cluster.

If you want the cluster to act as a reliable time source, make the time service a highly available service after creating the cluster. See the *Cluster Highly Available Applications* manual for information on setting up highly available services.

During cluster creation, if the Tru64 UNIX system is using NTP, the first cluster member inherits the NTP setup of the Tru64 UNIX system. When you add members, each member becomes a peer of the other members.

### 3.3.3 Name Server

A cluster can act as a name server, or client, or both. When the base operating system is configured as a BIND server, the BIND daemon, `named`, is automatically configured as a single-instance highly available service in the cluster.

When configuring a name server (for example, BIND) on the Tru64 UNIX system, make sure that the `hosts` entry in `/etc/svc.conf` has the `local` service listed before the `bind` or `yp` services. For example:

```
hosts=local,bind,yp
```

### 3.3.4 NFS

Because a TruCluster Server cluster can provide highly reliable Network File System (NFS) services to clients, we recommend that you configure the base operating system as an NFS server before creating a cluster. You can also configure the system as an NFS client.

When the base operating system is configured as an NFS server, the NFS `lockd` and `statd` daemons are configured as a single-instance highly available service in the cluster. The cluster is thus a highly available NFS server.

### 3.3.5 NIS

If the cluster will be a Network Information Service (NIS) master, slave, or client, configure NIS before creating a cluster. See the Tru64 UNIX *Network Administration: Services* manual for information on configuring NIS.

All cluster members must be in the same NIS domain. If the Tru64 UNIX operating system is configured as a master server, all cluster members will be configured as NIS masters. If the Tru64 UNIX operating system is configured as a slave server or a client, all cluster members will be configured as slaves or clients.

### 3.3.6 DHCP

If the base operating system is configured as a Dynamic Host Configuration Protocol (DHCP) server, the DHCP daemon, `joind`, is configured as a single-instance, highly available service in the cluster.

**Caution**

Do not configure the system as a DHCP client. A member of a cluster cannot be a DHCP client, because the IP names and addresses associated with each member's cluster interconnect must be absolutely stable in order to form a cluster.

### 3.3.7 Mail Server

If the cluster will be a mail server, configure mail before creating a cluster. See the Tru64 UNIX *System Administration* manual and the TruCluster Server *Cluster Administration* manual for information on configuring mail.

### 3.3.8 Print Server

If the cluster will be a print server, configure printing before creating a cluster. See the Tru64 UNIX *System Administration* manual and the TruCluster Server *Cluster Administration* manual for information on configuring printers.

### 3.3.9 Network Services Summary

Table 3–2 summarizes the preferred network configuration for the Tru64 UNIX system before beginning the TruCluster Server installation. The table also lists what configuration information is passed to the first cluster member by `clu_create`, and to additional cluster members by `clu_add_member`.

**Table 3–2: Network Services Summary**

| Service/Daemon | Recommendation | Comment |
|---|---|---|
| Routing Daemon | `gated` | Do not use `ogated` or `routed`. |
| Time Server | NTP | The first cluster member inherits the configuration of the base operating system. Additional members are automatically configured as peers. |
| Name Server | Optional (BIND and/or NIS) | All cluster members must be in the same domain (DNS and NIS). In `svc.conf`, set `hosts=local,bind,yp`. If the base operating system is configured as a name server, that name server will be configured as a highly available service in the cluster. If the cluster is a BIND server, the cluster alias name will be the name of the BIND server in the clusterwide `/etc/resolv.conf` file. |
| NFS | Optional (set up as a server on the base operating system; can also configure as a client) | All cluster members will run client versions of the `lockd` and `statd` daemons. One cluster member at a time will run the server versions of the `lockd` and `statd` daemons, which will be configured as a highly available service. |
| NIS | Optional | All cluster members must be in the same domain. If the base operating system is configured as a master server, NIS will be configured as a highly available service in the cluster. If the base operating system is configured as a slave server or a client, all cluster members will be configured as slaves or clients. |
| DHCP | Optional (server only) | If configured on the base operating system, DHCP will be configured as a highly available service in the cluster. |
| Mail | Optional | Configure before creating cluster. |
| Print Server | Optional | Configure before creating cluster. |

## 3.4 Configure Enhanced Security (Optional)

To configure enhanced security in a cluster, do the following:

- When installing Tru64 UNIX Version 5.1A, select the enhanced security subsets (those whose names are in the form of `OSFC2SEC`*xxx* and `OSFXC2SEC`*xxx*).
- Before loading the TruCluster Server license and subsets and creating the cluster, fully configure enhanced security on the Tru64 UNIX system.

It is easier to configure enhanced security before creating a cluster than to configure individual cluster members after creating a cluster. Moreover, because some components, such as Enhanced (C2) Security, depend upon a kernel rebuild or shared library substitution, if you configure them after cluster creation you need to reboot the entire cluster.

See the Tru64 UNIX *Security* manual for information on how to select and configure security options.

## 3.5 Configure the Disks Needed for Cluster Installation

After you install and boot the Tru64 UNIX operating system, determine the sizes and locations of the disks you will use when installing the cluster. The location of member boot disks is especially important because you must boot each member from its boot disk. (A member booted from the wrong boot disk will most likely panic during the boot. Depending on the type of panic, this might affect other cluster members.)

### 3.5.1 Partition Sizes

Verify that the partitions on the disks you plan to use for the cluster file systems and member boot disks meet the minimum size requirements in Table 2–2. Also, use the information in Section 2.4.5 to provide space for future rolling upgrades.

The disk or disks that you plan to use for the clusterwide root (`/`), `/usr`, and `/var` file systems must have disk labels before you run the `clu_create` command. The reason is that `clu_create` prompts for partition information, so the partitions must exist.

Use the `disklabel` or `diskconfig` commands to label and configure disks. You can also use the SysMan Station to view and modify storage configurations.

**Note**

The disks used for member boot disks and the quorum disk do not have to be labeled before use.

The following example shows the output from the `disklabel` command for a disk whose `b`, `g`, and `h` partitions will hold the clusterwide root (`/`), `/usr`, and `/var` file systems in a two-node cluster (cylinder information is not shown):

```
# disklabel -r dsk6
# /dev/rdisk/dsk6c:
type: SCSI
disk: RZ1DF-CB
label:
flags: dynamic_geometry
```

```
bytes/sector: 512
sectors/track: 168
tracks/cylinder: 20
sectors/cylinder: 3360
cylinders: 5273
sectors/unit: 17773524
rpm: 7200
interleave: 1
trackskew: 28
cylinderskew: 72
headswitch: 0           # milliseconds
track-to-track seek: 0  # milliseconds
drivedata: 0

8 partitions:

#            size       offset    fstype  fsize  bsize
  a:       786432            0    unused      0      0
  b:      1552131       786432    unused      0      0
  c:     17773524            0    unused      0      0
  d:      5400220      1572864    unused      0      0
  e:      5400220      6973084    unused      0      0
  f:      5400220     12373304    unused      0      0
  g:      7595651      2338563    unused      0      0
  h:      7839310      9934214    unused      0      0
```

### 3.5.2 Determining Disk Locations

You can use the SysMan Station and the `hwmgr` command to help map device special file names to physical locations. You do not want to boot the wrong disk for a cluster member, so it is important to map a device special file to the correct device for the console that is booting the disk.

**Note**

If you plan to use Fibre Channel storagesets for any disks required for the cluster, read the Fibre Channel chapter in the *Cluster Hardware Configuration* manual.

The following general-purpose `hwmgr` command displays the device name, physical location, and worldwide ID (WWID) for each disk known by the Tru64 UNIX system:

```
# hwmgr -get attr -a dev_base_name \
  -a phys_location -a name -category disk|more
```

Locating the first member's cluster boot disk is not a problem because you will run the `clu_create` command on the system that you are configuring as the first cluster member. The `clu_create` command sets the console `bootdef_dev` variable to the correct boot disk. However, you run the `clu_add_member` on one system to configure a boot disk for another system. The other system does not have an operating system; the only information you can get about its storage configuration is from its console. You need to

map the dsk name that you will specify to clu_add_member to the DK name that you will use to boot this disk from the new member's console.

When adding a member, the clu_add_member command displays information about the new member's boot disk. This information includes, if known, the disk's manufacturer, model number, physical location (bus/target/logical unit number (LUN)), and serial number (WWID). The clu_add_member command uses the following hwmgr command to gather information:

```
# hwmgr -get attr -a dev_base_name=new_member_boot_disk \
-a serial_number -a manufacturer -a model -a phys_location
```

The clu_add_member command reformats the output of the hwmgr command and displays disk location information in the following format:

```
Manufacturer: DEC
Model: HSG80
Target: 6
Lun: 141
Serial Number: SCSI-WWID:01000010:6000-1fe1-0000-0d60-0009-8090-0610-0
```

If you are using Fibre Channel storagesets, you can use the wwidmgr -show wwid command to locate the disk from the console of the new member. (The *Cluster Hardware Configuration* manual provides several examples that show how to use the wwidmgr command.)

If you cannot obtain the serial number of a member boot disk, mapping a device special file name to a console device name with a high degree of confidence depends on whether or not your storage configuration is symmetrical:

- Symmetrical: The systems in the cluster are identical and are configured in exactly the same fashion. You know that bus 1, bus 2, bus 3, ... bus n are the same on all systems; from the consoles, each DKB bus references the same physical bus throughout the cluster, and so on for DKC, DKD, ... DKZ. If this is the case, you can use the hwmgr command or the sysman hw_devices command to display the bus/target/logical unit number (LUN) associated with the dsk special file. You can then use this bus/target/LUN to derive the console device name for the disk by following the procedure in Section 3.5.2.1.
- Nonsymmetrical: The console bus name (for example, DKB) on one system is not the same physical bus as another system's DKB. You cannot guarantee that the bus Tru64 UNIX sees as bus 1 is referenced as bus DKB from all consoles. This may be the case if your cluster contains different types of systems with different numbers of SCSI adapters, or if you did not take care when inserting adapters and cabling buses to create a symmetrical storage configuration (at least for the bus or buses for the member boot disks). Use the procedure in Section 3.5.2.2.

#### 3.5.2.1 Symmetrical Storage Configuration

The following procedure describes how to map a `/dev/disk/dsk*` special file name to the name of a disk displayed by a `show device` command at the new member's console. This procedure will work only when the system on which you run the `hwmgr` command is the same type of system as the one that will later attempt to boot the disk — the same model systems, and the same type and number of storage adapters, located in the same slots on both machines, and connected to the same shared storage. Both consoles have the same view of the bus that contains the disks you will later configure as cluster member boot disks. If you are not sure that the two systems meet these requirements, see the nonsymmetrical procedure in Section 3.5.2.2.

1. On the Tru64 UNIX system, use the `hwmgr` command or the SysMan Station to map the `dsk` filename to the bus/target/LUN of the physical device. (The device special file names for shared storage will be the same for the Tru64 UNIX system and for the cluster.) For example, to find the bus/target/LUN for `dsk13`, enter:

   ```
   # hwmgr -view devices | grep dsk13
   56: /dev/disk/dsk10c  DEC  RZ1CF-CF (C) DEC bus-1-targ-11-lun-0c
   ```

   The disk is at bus 1, target 11, and LUN 0.

2. At the console of the system that will boot the disk, enter the `show device` command:

   ```
   >>> show device
        .
        .
        .
   dkb100.1.0.12.0         DKB100              RZ28M  1104
        .
        .
        .
   ```

   From the console's point of view, the class driver designator for a SCSI disk is `DK`. Bus numbering depends on firmware probe order. The first bus discovered is A, the second bus as B, and so on. In the same manner, the first target is 0, the second as 100, the third as 200, and so on. The `hwmgr` output of `bus-1-targ-1-lun-0` usually translates to `DKB100`.

   **Note**

   If you are using Fibre Channel storagesets, You can also use the console `wwidmgr -show wwid` command to display the list of user-defined IDs (UDIDs) and their associated worldwide IDs (WWIDs).

3. Write the `dsk` name, the `DK` name, and the WWID in the Member Attributes table in Table A–3.

#### 3.5.2.2 Nonsymmetrical Storage Configuration

If your storage configuration is not symmetrical, no command can unambiguously map one system's operating system view of storage (`dsk`) to another system's console view of storage (`DK`).

You can decide whether to map from the console to the disk to the operating system or vice versa. The following procedure starts from the operating system and works back to the console of the other system:

1. Use the `hwmgr` command to flash the activity light on the disk you want to use as a boot disk. For example, to flash the light on the disk accessed through special file `/dev/disk/dsk13`, enter:

   ```
   # hwmgr -flash light -dsf dsk13
   ```

2. Locate the disk by its flashing light and note its position in its storage cabinet.
3. Trace the cable from the cabinet back to the system that will use this disk as its boot disk.

   You now know the location of the disk on its SCSI bus, and the location of the SCSI adapter in the system. If you know the adapter numbering scheme for the system, you can deduce the SCSI bus number for the adapter. (If you drew a storage map when you configured your cluster, that information will be useful. If you have a variety of disks, you can also use the disk's model number and WWID as other pieces of evidence.)

   The following example assumes that the disk you tracked is the second device on the second SCSI adapter for this system.

   At the console of the system that will boot the disk, enter the `show device` command:

   ```
   >>> show device
        .
        .
        .
   dkb100.1.0.12.0         DKB100              RZ28M  1104
        .
        .
        .
   ```

   From the console's point of view, the class driver designator for a SCSI disk is `DK`. Bus numbering depends on firmware probe order. The first bus discovered is A, the second bus is B, and so on. In the same manner, the first target is 0, the second as 100, the third as 200, and so on. The second target on the second SCSI adapter would be `DKB100`. If you know that the disk is an RZ28M, that information also helps to narrow down the choices (unless, of course, all the disks on the bus are RZ28Ms).

**Note**

If you are using Fibre Channel storagesets, you can use the console `wwidmgr -show wwid` command to display a list of user-defined IDs (UDIDs) and their associated worldwide IDs (WWIDs).

4. Write the `dsk` name, the `DK` name, and the WWID in the Member Attributes table in Table A–3.

# 4

# Create a Single-Member Cluster

After installing and configuring the Tru64 UNIX system, follow the directions in this chapter to make this system the first member of the cluster. Table 4–1 lists the tasks in order and references necessary information.

**Note**

If you are using Fibre Channel storagesets for the first member boot disk or the quorum disk, read the Fibre Channel chapter in the *Cluster Hardware Configuration* manual.

**Table 4–1: Create Single-Member Cluster Tasks**

| Task | See |
|---|---|
| Gather the information needed to create a cluster. | Chapter 2 |
| Install and configure Tru64 UNIX. | Chapter 3 |
| Register the TruCluster Server license. | Section 4.1 |
| Load the TruCluster Server subsets. | Section 4.2 |
| Run the `clu_create` command. | Section 4.3 |
| Boot the first member's cluster boot disk. | Section 4.3 and Section 4.4 |
| Make on-disk backup copies of important configuration files | Section 4.5 |
| Perform a full backup of the single-member cluster. | Tru64 UNIX *System Administration* manual |

## 4.1 Register the TruCluster Server Software License

The TruCluster Server kit includes a license Product Authorization Key (PAK). Use this PAK when registering a TruCluster Server license. (Section 1.5 describes the TruCluster Server license.) If you do not have a PAK, call your customer service representative.

For information on installing a license PAK, see the Tru64 UNIX *Software License Management* manual, `lmf`(8), and `lmfsetup`(8).

## 4.2 Load the TruCluster Server Subsets

To load the TruCluster Server kit, follow these steps:

1. Log in as superuser (`root`).
2. Mount the device or directory containing the TruCluster Server Software Version 5.1A kit.
3. Enter the `setld -l` command specifying the directory where the kit is located. For example, if you mount the CD-ROM on `/mnt`:

   ```
   # setld -l /mnt/TruCluster/kit
   ```

   You can choose one of the following subset installation options:

   - All mandatory subsets only
   - All mandatory and selected optional subsets
   - All mandatory and all optional subsets

   We recommend that you choose the "All mandatory and all optional subsets" option.

   After you select an option, the installation procedure verifies that there is sufficient file system space before copying the subsets onto your system.

**Note**

Patch kits include fixes for both the base operating system and for cluster software. If installing a patch kit, patch the system after loading the TruCluster Server subsets but before running `clu_create` to create a single-member cluster.

## 4.3 Run the clu_create Command

The `/usr/sbin/clu_create` command creates the first member of the cluster from the Tru64 UNIX system.

**Note**

The `clu_create` command uses `vdump` and `vrestore` to populate the clusterwide root (`/`), `/usr`, and `/var` file systems. If any of these file systems on the Tru64 UNIX system has a Network File System (NFS) file system mounted on it, and if that file system's NFS server is down, `vdump` will hang when trying to dump the file system. (If you run the `automount` daemon, remember that it mounts and unmounts NFS file systems, and the same potential for hangs exists.)

Before running clu_create, either unmount all NFS file systems or verify that they are accessible. Alternatively, you can reboot the Tru64 UNIX system before running clu_create to clean up any stale mounts. For systems that run the automount daemon, you can disable automounting by running the /sbin/init.d/nfsmount stop command before running clu_create.

---

Run the /usr/sbin/clu_create command. The command prompts for the information needed to create a single-member cluster. Answer the prompts using the information from the checklists in Appendix A. The command also provides online help for each question. To display the relevant help message, enter **help** or a question mark, **?**, at a prompt.

The clu_create command performs the following tasks:

- Sets up the clusterwide root (/), /usr, and /var file systems, and the first member's boot disk.
- Configures a quorum disk (optional).
- Builds a kernel with cluster components.

  If the kernel build succeeds, clu_create copies the new kernel to the first member's boot disk. If the kernel build fails, clu_create displays warning messages but continues creating this first member. (You can boot the cluster genvmunix from the boot disk and attempt to build a kernel on the single-member cluster.)
- Sets boot-related console variables: bootdef_dev and boot_reset; creates and sets boot_dev.

  If clu_create can set the variables and if the kernel build was successful, clu_create offers to reboot the system for you. If you choose not to reboot the system at this time, use the information in Section 4.4 when you are ready to boot the system as a single-member cluster.

  If clu_create cannot set the variables, halt the system, set the variables to the values specified in Section 2.6, and use the information in Section 4.4 to boot the system as a single-member cluster.

**Note**

If the first member's boot disk is accessed through HS controllers that are connected to dual SCSI or Fibre Channel buses and configured for multiple-bus failover, or the system is an AlphaServer 8200 or 8400, halt the system and see Section 2.6 for information on setting the bootdef_dev console variable.

---

The clu_create command writes a log file of the installation to /cluster/admin/clu_create.log. The log file contains all installation prompts, responses, and messages. Examine this log file for errors before booting the system as a single-member cluster. (Section C.1 contains a sample clu_create log file.)

## 4.4 Boot the System as a Single-Member Cluster

If you decided to boot the system yourself, examine the following items in the clu_create log file, /cluster/admin/clu_create.log, before booting the system as a single-member cluster:

1. Verify that the kernel for the new member built properly and was copied to the boot partition for this member. Look for the following kinds of messages in the clu_create.log file:

```
*** PERFORMING KERNEL BUILD ***
 Working....Tue May  8 15:54:11 EDT 2001

The new kernel is /sys/PEPICELLI/vmunix
  Finished running the doconfig program.

  The kernel build was successful and the new kernel
   has been copied to this member's boot disk.
  Restoring kernel build configuration.
```

2. Verify that the boot-related console variables are set to boot the cluster boot disk for the first member, not the Tru64 UNIX boot disk. In the log, look for the following kinds of messages:

```
Updating console variables
  Setting console variable 'bootdef_dev' to dsk10
  Setting console variable 'boot_dev' to dsk10
  Setting console variable 'boot_reset' to ON
  Saving console variables to non-volatile storage
```

3. If the new kernel is in place and the console variables are set correctly, reboot the system as a single-member cluster from its newly created cluster member boot disk. For example:

```
# shutdown -r now
```

4. If the kernel build did not succeed or the console variables could not be set, halt the system:

```
# shutdown -h now
```

   Perform the following procedure:

   a. If clu_create could not set the console variables, set the console variables according to the values specified in Section 2.6. (The remaining steps assume that the console variables are set correctly.)

b. If the kernel build succeeded, boot vmunix from the first member's cluster boot disk (make sure that bootdef_dev is set to the first member's cluster boot disk, not to the Tru64 UNIX disk):

```
>>> boot
```

c. If the kernel build did not succeed or you could not boot vmunix from the first member's boot disk, make sure that bootdef_dev is set to the first member's cluster boot disk (not to the Tru64 UNIX disk) and boot genvmunix:

```
>>> boot -file genvmunix
```

When the system reaches multiuser mode, log in and attempt to build a kernel. If the build succeeds, copy (cp) the new kernel from /sys/*HOSTNAME*/vmunix to /vmunix. (If you move (mv) the kernel to /vmumix, you will overwrite the /vmunix context-dependent symbolic link (CDSL).) Then reboot the system so it will be running on its customized kernel.

```
# doconfig -c HOSTNAME
# cp /sys/HOSTNAME/vmunix /vmunix
# shutdown -r now
```

5. If you assigned a reserved network IP address to the default cluster alias, see Section 2.3.1 for information on advertising that alias address.
6. If you plan to use LSM, see Section 2.4.2.

When you boot this node as a single-member cluster, note that some access-related files, like /etc/ftpusers, are shared by all cluster members while other files, like /etc/securettys, are replaced by CDSLs that point to member-specific files. The reason is that files like ftpusers deal with user accounts (which are clusterwide entities), while files like securettys deal with member-specific information, in this case tty devices.

When first booted as a cluster member, the system runs the clu_check_config command to examine the configuration of several important cluster subsystems. Look at the clu_check_config log files in the /cluster/admin directory to verify that these subsystems are configured properly and operating correctly. If you discover any problems, read clu_check_config(8) so you know what tests the command performs. You can then run the command in verbose mode to display more information about why a subsystem failed the initial test. See the Tru64 UNIX *System Administration* and the TruCluster Server *Cluster Administration* manuals for information on configuring subsystems.

The following group of commands are useful for taking a quick look at the initial configuration of the cluster and some of its major subsystems:

```
# clu_get_info -full | more
# clu_quorum
# cfsmgr | more
# drdmgr 'ls /etc/fdmns/* | grep dsk | sed 's/[a-z]$//' |\
   uniq | sort' | more
# cluamgr -s DEFAULTALIAS
# caa_stat
```

## 4.5 Make On-Disk Backup Copies of Important Configuration Files

Because cluster members rely on the information in the following files, we recommend that, after booting the first member of the cluster, you make on-disk copies of these files in case of inadvertent modification. For member-specific files, the examples assume that the member ID of the first member is `1` (`memberid=1`).

- `/etc/sysconfigtab.cluster`:

  ```
  # cp /etc/sysconfigtab.cluster /etc/sysconfigtab.cluster.sav
  ```

- `/etc/rc.config.common`:

  ```
  # cp /etc/rc.config.common /etc/rc.config.common.sav
  ```

- `/etc/sysconfigtab` - This file is a CDSL whose target is:

  ```
  ../cluster/members/{memb}/boot_partition/etc/sysconfigtab
  ```

  To make a backup copy, change directory to the first member's `boot_partition/etc` directory and make a copy of its `sysconfigtab` file.

  For example:

  ```
  # cd /cluster/members/member1/boot_partition/etc
  # cp sysconfigtab sysconfigtab.sav
  ```

- `/etc/rc.config` - This file is a CDSL whose target is:

  ```
  ../cluster/members/{memb}/etc/rc.config
  ```

  To make a backup copy, change directory to the first member's `etc` directory and make a copy of its `rc.config` file. For example:

  ```
  # cd /cluster/members/member1/etc
  # cp rc.config rc.config.sav
  ```

In addition, we recommend that you perform a full backup of the single-member cluster.

# 5

# Add Members

To add a member to the cluster, run the `clu_add_member` command on any current cluster member. This command creates directories and files for the new member in the clusterwide root (`/`), `/usr`, and `/var` file systems. It also labels and populates a boot disk for the new member. Because the current member must write to the potential member's boot disk, this boot disk must be accessible to the current member; for example, on a shared bus.

**Notes**

You do not need to install Tru64 UNIX on each new member. (However, you will have to register a Tru64 UNIX license on each new member after you boot it.)

When adding members, add one member and boot it into the cluster before running `clu_add_member` to add another member.

If you are using Fibre Channel storagesets for member boot disks, read the Fibre Channel chapter in the TruCluster Server *Cluster Hardware Configuration* manual.

Follow the directions in this chapter to add a member to a cluster. Table 5–1 lists the tasks in order and references necessary information.

**Table 5–1: Add Member Tasks**

| Task | See |
|---|---|
| Gather the information for the new member. | Chapter 2 |
| Set the `boot_osflags`, `boot_reset`, and `bus_probe_algorithm` console variables on the system that will become the new member. | Section 2.6 |
| Update SRM firmware. | Section 5.1 |
| Verify that the Member Attributes checklist in Appendix A contains the `dsk` special file name, the `DK` console device name, and (if known) the WWID for the new member's boot disk. If you do not know the physical location of the new member's boot disk, locate it.[a] | Section 2.4.1, Section 3.5, Appendix A |
| Make sure that the system cannot boot during the installation. | Section 5.2 |

**Table 5–1: Add Member Tasks (cont.)**

| Task | See |
|---|---|
| Run the `clu_add_member` command on a current cluster member. | Section 5.3 |
| From the new member's console, boot the new member into the cluster. | Section 5.4 |
| Make on-disk backup copies of important configuration files | Section 5.5 |

[a] Before exiting, the `clu_add_member` command will display whatever information it has on the disk type, location, and WWID.

**Note**

Before running `clu_add_member`, make sure that the current member is fully configured (applications installed, network interfaces configured, network services configured, TruCluster Server license installed, and so forth).

If you fully configured the Tru64 UNIX system before running `clu_create`, you should be ready to run `clu_add_member`. If you did not fully configure the Tru64 UNIX system before creating a cluster, we recommend that you finish configuring the cluster before running `clu_add_member` because the new member will inherit a large portion of its configuration from the current member.

## 5.1 Update SRM Firmware

New versions of Tru64 UNIX and TruCluster Server usually require new versions of the AlphaServer SRM firmware. Firmware updates are located on the Alpha Systems Firmware CD-ROM, which is included in the base operating system Software Distribution Kit. To determine whether you need to update firmware, see the TruCluster Server *Software Product Description* (SPD) and the firmware release notes for each type of system in the cluster. Update firmware as needed. You can find the latest version of the TruCluster Server SPD at the following URL:

**http://www.tru64unix.compaq.com/docs/pub_page/spds.html**

## 5.2 Prevent the New Member from Booting During Installation

Halt or turn off the system that will become the new member. If halting the system:

1. Set the `auto_action` console variable to `halt`:

```
>>> set auto_action halt
```

2. Set the bootdef_dev console variable to an empty string:

```
>>> set bootdef_dev ""
```

The reason for these precautions is to make sure that the system cannot boot from the disk that clu_add_member will configure as the new member's boot disk.

## 5.3 Run the clu_add_member Command

Run the /usr/sbin/clu_add_member command. The command prompts for the information needed to create a single-member cluster. Answer the prompts using the information from the checklists in Appendix A. The command also provides online help for each question. To display the relevant help message, enter **help** or a question mark, **?**, at a prompt.

The clu_add_member command configures the new member's boot disk, adds and modifies files in the clusterwide file systems, and gives you the option of loading the TruCluster Server license PAK.

**Note**

You can boot a system that does not have a TruCluster Server license. The system joins the cluster and boots to multiuser mode, but only root can log in (with a maximum of two users). The cluster application availability (CAA) daemon, caad, is not started. The system displays a license error message reminding you to load the license. This policy enforces license checks while making it possible to boot and repair a system during an emergency.

Load only the TruCluster Server license at this time. Do not load the Tru64 UNIX license PAK. (You will load the Tru64 UNIX license PAK and any other license PAKs you need after you boot the new member for the first time.)

The clu_add_member command writes a log file of the installation to /cluster/admin/clu_add_member.log. Examine this log file for errors before continuing. (Section C.2 contains a sample clu_add_member log file.)

**Note**

As described in Section 3.5.2, the clu_add_member command attempts to provide some hints about the physical location of a

new member's boot disk. This information is displayed on the screen and written to the clu_add_member.log file.

## 5.4 Boot the New Member

After running clu_add_member, go to the **console** of the newly installed member and perform the following steps:

1. At the console of the new member, set the console variable boot_osflags to A so the system will boot to multiuser mode:

   ```
   >>> set boot_osflags A
   ```

2. At the console of the new member, boot genvmunix from the new member's boot disk:

   ```
   >>> boot -file genvmunix new_member_boot_disk
   ```

   Remember to specify the correct DK device name for the boot disk at the console; do not use the dsk special file name you supplied to clu_add_member. For example, if the console device name for the new member's boot disk is DKC600:

   ```
   >>> boot -file genvmunix DKC600
   ```

   **Caution**

   If you are not sure of the location of the newly installed boot disk, use the information in Section 3.5.2 to determine its physical location.

   In addition to making sure you boot the correct disk, note that you cannot copy a member's boot disk to another disk and then boot that member from the copy. You must boot a member from the disk created by clu_add_member. To change the boot disk for a member, you must reinstall the member following the instructions in Section 6.2.

3. During its first boot, the new member automatically performs the following tasks:

   a. Configures all loaded subsets.

   b. Attempts to build a customized kernel.

      - If the kernel build succeeds, copies the new kernel to the member's boot partition.
      - If the build does not succeed, when the system reaches multi-user mode, you can run doconfig to build a kernel.

Copy (cp) the new kernel from /sys/*HOSTNAME*/vmunix to /vmunix. (If you move (mv) the kernel to /vmumix, you will overwrite the /vmunix CDSL.)

c. Runs a script so you can configure additional network interfaces. (The clu_add_member command configures only the cluster interconnect interface.) We recommend that you configure at least one additional interface: the public network interface that is associated with the new member's host name.

**Note**

If there is no interface associated with the new member's host name and if you use the Common Desktop Environment (CDE), CDE will display an error dialog box when you first log in. Read the dialog box, then click on OK to log out. Select a failsafe session and log in again. Configure a network interface and associate it with the system's host name.

d. Sets the boot_reset and bootdef_dev variables, and creates and sets the boot_dev console variable.

4. The system continues to boot to multi-user mode. When the system finishes booting to multiuser mode, register the Tru64 UNIX license and any other required application licenses. If you did not register the TruCluster Server license while running clu_add_member, register it now.

5. Because this member is still running genvmunix, reboot the system so it is using its custom kernel. This reboot is a required step when adding a member to the cluster.

```
# shutdown -r now
```

**Note**

If the new member's boot disk is accessed through HS controllers that are connected to dual SCSI or Fibre Channel buses and configured for multiple-bus failover, or the system is an AlphaServer 8200 or 8400, halt the system and see Section 2.6 for information on setting the bootdef_dev console variable.

When first booted as a cluster member, the system runs the clu_check_config command to examine the configuration of several important cluster subsystems. Look at the clu_check_config log files

in the /cluster/admin directory to verify that these subsystems are configured properly and operating correctly. If you discover any problems, read clu_check_config(8) so you know what tests the command performs. You can then run the command in verbose mode to display more information about why a subsystem failed the initial test. See the Tru64 UNIX *System Administration* and the TruCluster Server *Cluster Administration* manuals for information on configuring subsystems.

## 5.5 Make On-Disk Backup Copies of Important Configuration Files

Because cluster members rely on the information in the following files, we recommend that, after booting each additional member of the cluster, you make on-disk copies of these files in case of inadvertent modification. For member-specific files, the examples use member 2 (memberid=2). Substitute the correct member ID for your new member when making backup copies of files.

- /etc/sysconfigtab.cluster:

  ```
  # cp /etc/sysconfigtab.cluster /etc/sysconfigtab.cluster.sav
  ```

  Because quorum vote information is updated each time a member is added, make a backup copy of this file after adding each member.

- /etc/sysconfigtab - This file is a CDSL whose target is:

  ```
  ../cluster/members/{memb}/boot_partition/etc/sysconfigtab
  ```

  To make a backup copy, change directory to the new member's boot_partition/etc directory and make a copy of its sysconfigtab file.

  For example:

  ```
  # cd /cluster/members/member2/boot_partition/etc
  # cp sysconfigtab sysconfigtab.sav
  ```

- /etc/rc.config - This file is a CDSL whose target is:

  ```
  ../cluster/members/{memb}/etc/rc.config
  ```

  To make a backup copy, change directory to the new member's etc directory and make a copy of its rc.config file. For example:

  ```
  # cd /cluster/members/member2/etc
  # cp rc.config rc.config.sav
  ```

# 6

# Reinstalling Cluster Members

This chapter explains how to perform the following tasks:

- Re-create a single-member cluster (Section 6.1)
- Reinstall an individual cluster member (Section 6.2)
- Use installation configuration files to automate the re-creation of a cluster or the reinstallation of a single member (Section 6.3)

## 6.1 Re-creating a Single-Member Cluster

This procedure deletes the members of the current cluster, boots the Tru64 UNIX operating system, and re-runs the `clu_create` command to create a new cluster, which may or may not have the same configuration as the current cluster. (The procedure in Section 6.3 uses configuration files to create a cluster with the same configuration as the current cluster.)

To re-create a single-member cluster, follow these steps:

1. Determine which disk is the Tru64 UNIX boot disk:

   a. Look in the `/etc/fdmns` directory to find the special file name for the Tru64 UNIX boot disk (usually in the `root_domain` directory). For example:

   ```
   # ls /etc/fdmns/root_domain
   /dev/disk/dsk0a
   ```

   b. If the disk in on a private bus, use SysMan Station to determine which system has the disk.

   c. Flash the light on the disk. For example:

   ```
   # hwmgr -flash light -dsf /dev/disk/dsk0
   ```

   Section 3.5.2 provides information you can use to map a disk's `dsk` device special file name to its physical location.

2. On a cluster member that has direct access to the Tru64 UNIX disk, delete all other cluster members. (See the *Cluster Administration* manual for information on deleting cluster members. If you want to save the configuration files for these members before you delete them, see Section 6.3.) This system is now the only member of the cluster.

**Caution**

We strongly recommend deleting all cluster members before re-creating a cluster. Otherwise, there is a chance that someone might attempt to boot an old member boot disk into the new cluster.

3. Determine, and remember, the device special file name for this member's cluster boot disk. For example, if this system is member 1, enter:

```
# ls /etc/fdmns/root1_domain
/dev/disk/dsk10a
```

4. Halt this system:

```
# shutdown -h now
```

5. From the system console, boot the Tru64 UNIX operating system to multi-user mode:

```
>>> boot UNIX_disk
```

6. Because this system was not deleted from the cluster, its cluster member boot disk is still capable of being booted by any system that has access to it. To prevent a system from inadvertently booting this disk, we recommend that you zero (clear) the disk label. For example, to zero the disk label for `dsk10`, enter:

```
# disklabel -z dsk10
```

7. Run the `clu_create` command to create a single-member cluster:

```
# /usr/sbin/clu_create
```

See Chapter 4 for information on running the `clu_create` command.

## 6.2 Reinstalling Individual Cluster Members

To reinstall a member, halt that member and perform the following procedure on another member:

1. To remove the member from the cluster, run the `clu_delete_member` command. See the *Cluster Administration* manual for information on removing a member from a cluster.
2. To add the member back into the cluster, run the `clu_add_member` command. See Chapter 5 for information on running the `clu_add_member` command. See `clu_add_member`(8) and Section 6.3 for information on using configuration files.

## 6.3 Using Installation Configuration Files

When the original TruCluster Server cluster was created, `clu_create` and `clu_add_member` wrote configuration files to the `/cluster/admin` directory. The files are named `.member`*n*`.cfg`, where *n* is the member ID of the cluster member. Each time these commands are run successfully, they append the current configuration information to the respective member configuration files. To learn about configuration files and the restrictions on their use, read `clu_create`(8), and `clu_add_member`(8).

The following example shows a configuration file created by `clu_create` for a cluster that has installed the Worldwide Language Support (WLS) subsets and uses LAN hardware for its cluster interconnect (comment lines are wrapped for readability).

```
# clu_create saved configuration values:
# date: Tue May 15 15:47:14 EDT 2001 hostname \
# pepicelli.zk3.dec.com
# Previously saved value in this file have been \
# converted to comment lines
clu_alias_ip=16.140.112.209
clu_boot_dev=dsk10
clu_i18n_dev=dsk14
clu_ics_dev=ics0
clu_ics_host=pepicelli-ics0
clu_ics_ip=10.0.0.1
clu_mem_votes=1
clu_memid=1
clu_name=deli.zk3.dec.com
clu_nr_dev=nr0
clu_phys_devs=ee0,ee1
clu_quorum_dev=dsk7
clu_quorum_votes=1
clu_root_dev=dsk1b
clu_tcp_host=member1-icstcp0
clu_tcp_ip=10.1.0.1
clu_usr_dev=dsk2c
clu_var_dev=dsk3c
```

These installation configuration files contain *variable*=*value* pairs that provide a way to automate the tasks performed by `clu_create` and `clu_add_member`. When you run one of these commands and specify the `-c` option and the name of a configuration file, the command uses the configuration file as input (replacing the need to manually enter information).

If `clu_create` or `clu_add_member` cannot find a required name/value pair when reading a configuration file, the command prompts for the required information and then returns to reading the configuration file. For example, you can delete the first member of the cluster, and then run

`clu_add_member -c` using the configuration file created by `clu_create` (usually `/cluster/admin/.member1.cfg`). The `clu_add_member` command prompts you for any information it needs that is not in the configuration file created by `clu_create`.

**Note**

Configuration files are generated by programs and read by programs. In general, do not manually edit configuration files.

Configuration files make it easy to re-create an existing cluster. However, the information in the configuration files must be accurate; for example, host names, IP addresses, and disk special file names. Because disk devices are named in order of discovery, using configuration files to re-create a cluster implies that you run `clu_create -c` *`member_conf_file`* on the same system it was run on previously and that the storage configuration has not changed. In addition, you must add the members in the same order used when creating the original cluster. For each member, run `clu_add_member -c` *`member_conf_file`* on the same member it was run on previously. (The `date:` comment line in a configuration file contains the date that `clu_create` or `clu_add_member` was run and the name of the host on which it was run.)

If your existing cluster meets the following requirements, you can automate the re-creation of the cluster by saving the current `clu_create` and `clu_add_member` configuration files:

- You plan to use the same disks, host names, and IP addresses for the upgraded cluster.
- The disk storage configuration is the same as when you created the original cluster and added members. This means that the disk names in the `clu_create` and `clu_add_member` configuration files reference the same devices as when the commands were last run. If you have added or removed devices since the cluster was created, the hardware view from the original Tru64 UNIX installation might not match the current configuration. This can be an issue if any of the affected disk names are used for the clusterwide `root (/)`, `/usr`, or `/var` file systems, for member boot disks, or for the quorum disk.

**Note**

The remainder of this section describes how to re-create the entire cluster. You can also use configuration files to re-add a member to a cluster.

The following procedure recreates a TruCluster Server cluster using the configuration files from the current cluster.

1. Perform a full backup of the current cluster.
2. Determine which disk is the Tru64 UNIX boot disk. It is often a private disk on the system that became the first cluster member:

   a. Look in the /etc/fdmns directory to find the special file name for the Tru64 UNIX boot disk (usually in the root_domain directory). For example:

      ```
      # ls /etc/fdmns/root_domain
      /dev/disk/dsk0a
      ```

   b. Use SysMan Station to determine which system has the disk.

   c. Flash the light on the disk. For example:

      ```
      # hwmgr -flash light -dsf /dev/disk/dsk4
      ```

      Section 3.5.2 provides information you can use to map a disk's dsk device special file name to its physical location. You need to know the disk's physical location because you will boot this disk from the system's console.

3. On a cluster member that has access to the Tru64 UNIX operating system disk, mount the disk and save the current configuration files and license PAKs to that disk. For example:

   ```
   # mount root_domain#root /mnt
   # mkdir /mnt/config_files /mnt/licenses
   # cp /cluster/admin/.member*.cfg /mnt/config_files
   # for i in `lmf list | grep -v Product | awk '{print $1}'`
     do
     lmf issue /mnt/licenses/$i.license $i
     done
   ```

   **Note**

   You might also want to save other information such as:

   - Site-specific CAA profiles and action scripts
   - Each member's /etc/rc.config file
   - Each member's cluster alias configuration file (/etc/clu_alias.config)
   - Modifications to system configuration files
   - /etc/fstab
   - A recursive listing (ls -R) of /etc/fdmns/*

In short, anything that has changed since you created the cluster and that you do not want to re-create. If you are not sure what to save, use the `sys_check -all` command to gather system configuration information. (You still have to manually save CAA profiles and scripts, cluster alias configuration files, and member-specific files.)

---

4. Halt the cluster:

   ```
   # shutdown -c now
   ```

5. From the system console, boot the Tru64 UNIX operating system to multi-user mode:

   ```
   >>> boot UNIX_disk
   ```

   **Note**

   This procedure assumes that you have not performed a rolling upgrade of the cluster since you installed the Tru64 UNIX operating system and created the original cluster. If you have performed a rolling upgrade, the version of the base operating system on this disk is not as current as the cluster you shut down. (You can use the `sizer -v` command to display the version of the Tru64 UNIX system.)

   If the Tru64 UNIX system is not at the latest version, do the following:

   a. Take the system to single-user mode.

   b. Delete the TruCluster Server subsets.

   c. Perform an update installation to the latest version of Tru64 UNIX.

   d. Load the latest version of the TruCluster Server subsets.

   e. (Optional) If a patch kit is available for the new version of the base operating system and cluster software, you can patch the Tru64 UNIX system now — before running `clu_create`. This means you will not have to roll the patch kit into the cluster later.

---

6. Register any required saved licenses. (The Tru64 UNIX and the TruCluster Server licenses should already be active.) The following example assumes that you have run `lmf list` and have removed any unneeded `*.license` files from `/licenses`:

```
# for i in /licenses/*.license
  do
  lmf register - < $i
  done
# lmf reset
```

7. Determine which saved configuration file to use with `clu_create`. Then run `clu_create -c`, specifying the name of the configuration file. For example:

```
# cd /config_files
# grep clu_create .*.cfg
.member1.cfg:# clu_create saved configuration values:
# /usr/sbin/clu_create -c /config_files/.member1.cfg
```

8. After booting the first cluster member, use the saved member configuration files to add the remaining members to the cluster.

**Caution**

Add members in the same order and from the same host that they were added in the original cluster. Otherwise, the device names might not be the same as in the original cluster.

Examine each configuration file to determine on which member the original `clu_add_member` command was run. Use the latest `# date` comment in each file to determine the time and the host on which the command was run. The following short script displays the name of a configuration file, the name of the host on which `clu_add_member` was run, and the name of the member that was added:

```
#! /bin/ksh
cd /config_files
for i in 'grep -l unix_host .member*.cfg'
  do
    print '\n' $i
    tail -21 $i | grep -E '^# date|^unix_host'
  done
```

Running the script on the sample three-member cluster displays the following output:

```
.member2.cfg
# date: Tue May 15 17:46:48 EDT 2001 hostname pepicelli.zk3.dec.com
unix_host=polishham.zk3.dec.com

.member3.cfg
# date: Tue May 15 18:09:32 EDT 2001 hostname polishham.zk3.dec.com
unix_host=provolone.zk3.dec.com
```

Using this information, you run `clu_add_member` with the `.member2.cfg` file on the first member of the cluster, `pepicelli`, to add member 2, `polishham`. After booting `polishham`, run

clu_add_member with the .member3.cfg file on polishham to add member 3, provolone.

For example, to add the second member of the cluster, run the following command on pepicelli:

```
# /usr/sbin/clu_add_member -c /config_files/.member2.cfg
```

Remember to boot each new member before adding the next one.

# 7

# Rolling Upgrade

A **rolling upgrade** is a software upgrade of a cluster that is performed while the cluster is in operation. One member at a time is upgraded and returned to operation while the cluster transparently maintains a mixed-version environment for the base operating system, cluster, and Worldwide Language Support (WLS) software. Clients accessing services are not aware that a rolling upgrade is in progress.

A rolling upgrade consists of an ordered series of steps, called **stages**. The commands that control a rolling upgrade enforce this order.

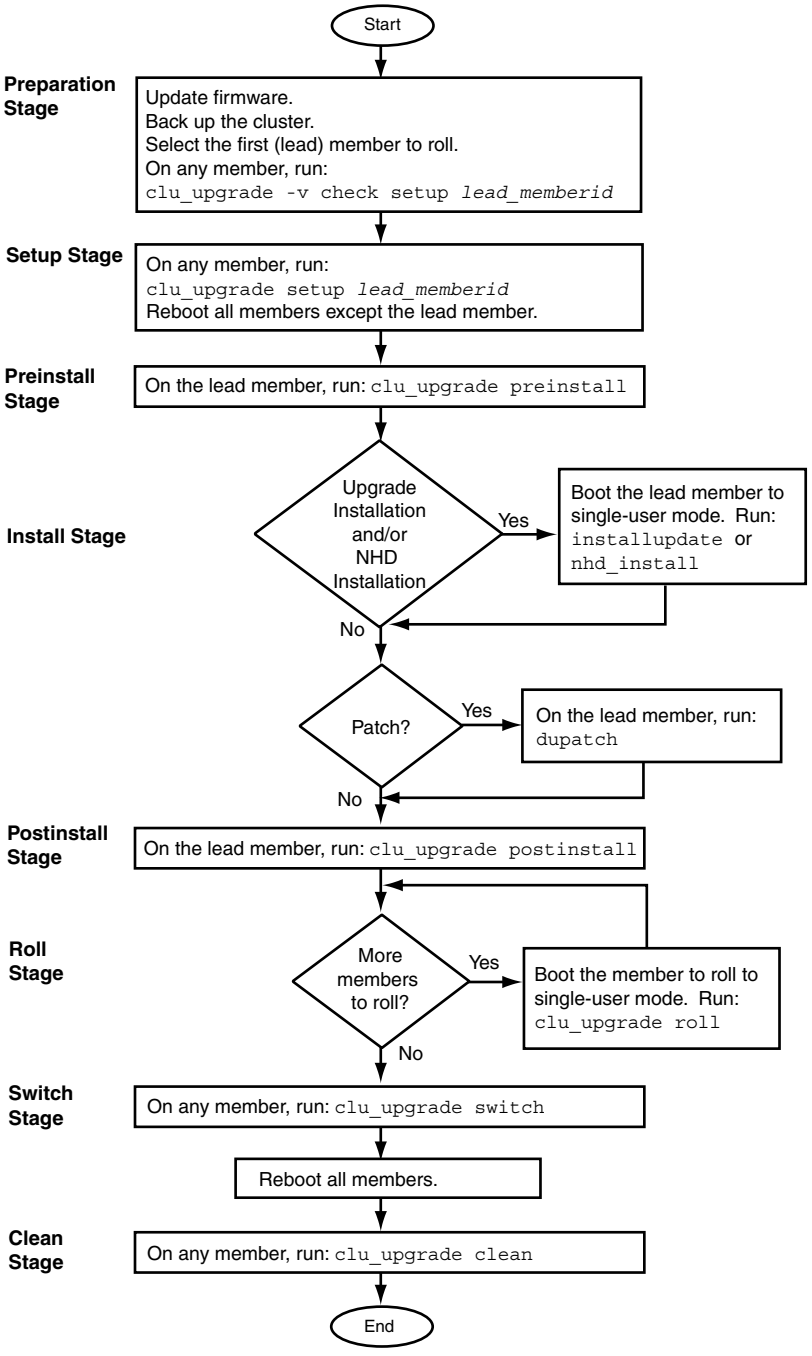
This first part of the chapter contains instructions for performing a rolling upgrade, for displaying the status of a rolling upgrade, and for undoing one or more stages of a rolling upgrade. The latter part of the chapter describes how a rolling upgrade works.

This chapter discusses the following topics:

- The tasks, and combinations of tasks, you can perform during a single rolling upgrade (Section 7.1)
- How to perform a rolling upgrade (Section 7.2)
- How to display the status of a rolling upgrade (Section 7.3)
- How to undo the stages of a rolling upgrade (Section 7.4)
- The commands used during a rolling upgrade (Section 7.5)
- Rolling upgrade stages (Section 7.6)
- Two mechanisms that support rolling upgrades: tagged files (Section 7.7) and version switches (Section 7.8)
- Rolling upgrade and layered products (Section 7.9)
- Rolling upgrade and RIS (Section 7.10)

Figure 7–1 provides a simplified flow chart of the tasks and stages that are part of a rolling upgrade initiated on a Version 5.1A cluster:

**Figure 7–1: Rolling Upgrade Flow Chart**



## 7.1 Rolling Upgrade Supported Tasks

The tasks that you can perform during a rolling upgrade depend on which versions of the base operating system and cluster software are currently running on the cluster. The main focus of this chapter is to describe the behavior of a rolling upgrade that starts on a TruCluster Server Version 5.1A cluster. However, because you may read this chapter in preparation for a rolling upgrade from TruCluster Server Version 5.1 to Version 5.1A, we point out rolling upgrade differences between the two versions.

The following list describes the basic tasks you can perform within a rolling upgrade. You can:

- Upgrade the cluster's Tru64 UNIX base operating system and TruCluster Server software. You perform this type of rolling upgrade to upgrade from the installed version to the next version. (You cannot bypass versions when performing a rolling upgrade of the base operating system and cluster software. You can only roll from one version to the next version. See Table 1–1 for a list of supported upgrade paths.)

  **Note**

  A rolling upgrade updates the file systems and disks that the cluster currently uses. The roll does not update the disk or disks that contain the Tru64 UNIX operating system used to create the cluster (the operating system on which you ran `clu_create`). Although you can boot the original operating system in an emergency when the cluster is down, remember that the differences between the current cluster and the original operating system increase with each cluster update.

- Patch the cluster's current versions of the Tru64 UNIX base operating system and TruCluster Server software.
- Install a New Hardware Delivery (NHD) kit (the cluster must be running TruCluster Server Version 5.1A).

Rolling in a patch kit or a New Hardware Delivery kit uses the same procedure as rolling in a new release of the base operating system and cluster software. The difference is which commands you run during the install stage. For example, to upgrade the base operating system and cluster software, you run `installupdate` in the install stage; to load a patch kit, you run `dupatch` in the install stage; to install an NHD kit, you run `nhd_install` in the install stage. Throughout this chapter, the term rolling upgrade refers to the overall procedure used to roll one or more software kits into a cluster.

You can perform only one rolling upgrade at a time. However, as shown in Figure 7–1, you can perform more than one task during a rolling upgrade.

Table 7–1 and Table 7–2 describe supported combinations of tasks within a single rolling upgrade for both the previous Version 5.1 release and the Version 5.1A release. If your current cluster is at Version 5.1, you are constrained to the tasks listed in Table 7–1; you cannot perform the tasks listed in Table 7–2 until the cluster is upgraded to Version 5.1A.

If the current cluster is running Version 5.1, a rolling upgrade can include the task combinations listed in Table 7–1:

**Table 7–1: Rolling Upgrade Tasks Supported by Version 5.1**

| |
|---|
| An update installation from Version 5.1 to Version 5.1A |
| A patch of Version 5.1 |
| An update installation from Version 5.1 to Version 5.1A followed by a patch of Version 5.1A |

If the current cluster is running Version 5.1A, a rolling upgrade can include the task combinations listed in Table 7–2:

**Table 7–2: Rolling Upgrade Tasks Supported by Version 5.1A**

| |
|---|
| An update installation from Version 5.1A to the next release |
| A patch of Version 5.1A |
| The installation of a New Hardware Delivery (NHD) kit onto a Version 5.1A cluster |
| An update installation from Version 5.1A to the next release of the base operating system and cluster software followed by a patch of the next release[a] |
| An NHD installation onto a Version 5.1A cluster followed by a patch of Version 5.1A software |
| An update installation from Version 5.1A to the next release of the base operating system and cluster software followed by the installation of an NHD kit for the next release[b] |
| An update installation from Version 5.1A to the next release, followed by the installation of an NHD kit for the next release, followed by a patch of the next release[b] |

[a] Within one rolling upgrade, you can combine an upgrade of the base operating system and cluster software with a patch of the new software. This means that during the install stage, you can run `installupdate` on the first member followed by `dupatch` to patch the newly installed software. When you roll the remaining members they automatically get both the new software and the patches.

However, you cannot patch the current software and then upgrade the base operating system and cluster software within one rolling upgrade. This operation requires two rolling upgrades.

[b] Allowed only if you have already installed an NHD kit on the Version 5.1A cluster.

## 7.2 Rolling Upgrade Procedure

Table 7–1 and Table 7–2 list the rolling upgrade tasks and combination of tasks supported for Version 5.1 and Version 5.1A.

In the procedure in this section, unless otherwise stated, run commands in multiuser mode. Each step that corresponds to a stage refers to the section that describes that stage in detail. We recommend that you read the detailed description of stages in Section 7.6 before performing the rolling upgrade procedure.

Some stages of a rolling upgrade take longer to complete than others. Table 7–3 lists the approximate time it takes to complete each stage.

**Table 7–3: Time Estimates for Rolling Upgrade Stages**

| Stage | Duration |
|---|---|
| Preparation | Not under program control. |
| Setup | 45 - 120 minutes.[a] |
| Preinstall | 15 - 30 minutes.[a] |
| Install | The same amount of time it takes to run `installupdate`, `dupatch`, `nhd_install`, or a supported combination of these commands on a single system. |
| Postinstall | Less than 1 minute. |
| Roll (per member) | Patch: less than 5 minutes.<br>Update installation: about the same amount of time it takes to add a member. |
| Switch | Less than 1 minute. |
| Clean | 30 - 90 minutes.[a] |

[a] These stages create, verify, or remove the tagged files required for a rolling upgrade. The time that it takes to run one of these stages depends on the speed of the member executing the command, the speed of the storage, and whether the member executing the command is the CFS server for the root (`/`), `/usr`, and `/var` file systems. Consider relocating these file systems to the member where you will run the `clu_upgrade` command.

You can use the following procedure to upgrade a TruCluster Server Version 5.1 cluster to Version 5.1A, and to upgrade a cluster that is already at Version 5.1A.

1. Prepare the cluster for the rolling upgrade (Section 7.6.1):

    a. Choose one cluster member to be the lead member (the first member to roll). (The examples in this procedure use a member whose `memberid` is `2` as the lead member. The example member's hostname is `provolone`.)

    b. Back up the cluster.

    c. If you will perform an update installation during the install stage, remove any blocking layered products, listed in Table 7–7, that are installed on the cluster.

d. To determine whether the cluster is ready for an upgrade, run the `clu_upgrade -v check setup` *`lead_memberid`* command on any cluster member. For example:

```
# clu_upgrade -v check setup 2
```

If a file system needs more free space, use AdvFS utilities such as `addvol` to add volumes to domains as needed. For disk space requirements, see Section 7.6.1. For information on managing AdvFS domains, see the Tru64 UNIX *AdvFS Administration* manual.

e. Verify that each system's firmware will support the new software. Update firmware as needed before starting the rolling upgrade.

2. Perform the setup stage (Section 7.6.2).

**Notes**

If your current cluster is at Version 5.1A and if you plan to upgrade the base operating system and cluster software during the install stage, mount the device or directory that contains the new TruCluster Server kit before running `clu_upgrade setup`. The `setup` command will copy the kit to the `/var/adm/update/TruClusterKit` directory. (If you are upgrading from Version 5.1 to Version 5.1A, this copy takes place during the preinstall stage.)

If your current cluster is at Version 5.1A and if you plan to install an NHD kit during the install stage, mount the device or directory that contains the new NHD kit before running `clu_upgrade setup`. The `setup` command will copy the kit to the `/var/adm/update/NHDKit` directory.

On any member, run the `clu_upgrade setup` *`lead_memberid`* command. For example:

```
# clu_upgrade setup 2
```

Section 7.6.2 shows the menu displayed by the Version 5.1A `clu_upgrade` command. If you are upgrading to Version 5.1A, your menu will not have the New Hardware Delivery option.

When the setup stage is completed, `clu_upgrade` prompts you to reboot all cluster members except the lead member.

3. One at a time, reboot all cluster members except the lead member. Do not start the preinstall stage until these members are either rebooted or halted.

4. Perform the preinstall stage (Section 7.6.3).

**Note**

If you are upgrading your cluster from Version 5.1 to Version 5.1A, mount the device or directory that contains the new TruCluster Server kit before running `clu_upgrade preinstall`. The `preinstall` command will copy the kit to the `/var/adm/update/TruClusterKit` directory.

On the lead member, run the following command:

```
# clu_upgrade preinstall
```

If your current cluster is at Version 5.1A, the `preinstall` command gives you the option of verifying or not verifying the existence of the tagged files created during the setup stage.

- If you have just completed the setup stage and have done nothing that would delete any of the tagged files, you can skip this test.
- If you completed the setup stage a while ago and are not sure what to do, let `preinstall` test the correctness of the tagged files.

5. Perform the install stage (Section 7.6.4).

**Note**

During the install stage you load the new software on the lead member, in effect rolling that member. When you perform the roll stage, this new software is propagated to the remaining members of the cluster.

The `clu_upgrade` command does not load software during the install stage. The loading of software is controlled by the commands you run: `installupdate`, `dupatch`, or `nhd_install`.

See Table 7–1 and Table 7–2 for the list of rolling upgrade tasks and combination of tasks supported for Version 5.1 and Version 5.1A.

a. See the Tru64 UNIX *Installation Guide* for detailed information on using the `installupdate` command.

   See the Tru64 UNIX and TruCluster Server *Patch Kit Installation Instructions* that came with your patch kit for detailed information on using the `dupatch` command.

   See the Tru64 UNIX *New Hardware Delivery Release Notes and Installation Instructions* that came with your NHD kit for detailed information on using the `nhd_install` command.

b. If the software you are installing requires that its installation command be run from single-user mode, halt the system and boot the system to single-user mode:

```
# shutdown -h now
>>> boot -fl s
```

**Note**

Halting and booting the system ensures that it provides the minimal set of services to the cluster and that the running cluster has a minimal reliance on the member running in single-user mode. In particular, halting the member satisfies services that require the cluster member to have a status of DOWN before completing a service failover. If you do not first halt the cluster member, there is a high probability that services will not fail over as expected.

When the system reaches single-user mode, run the following commands:

```
# init s
# bcheckrc
# lmf reset
```

c. Run the `installupdate`, `dupatch`, or `nhd_install` command.

6. (Optional) After the lead member performs its final reboot with its new custom kernel, you can perform the following manual tests before you roll any additional members:

   a. Verify that the newly rolled lead member can serve the shared root (/) file system.

      i. Use the `cfsmgr` command to determine which cluster member is currently serving the root file system. For example:

```
# cfsmgr -v -a server /

 Domain or filesystem name = /
 Server Name = polishham
 Server Status : OK
```

      ii. Relocate the root (/) file system to the lead member. For example:

```
# cfsmgr -h polishham -r -a SERVER=provolone /
```

b. Verify that the lead member can serve applications to clients. Make sure that the lead member can serve all important applications that the cluster makes available to its clients.

   You decide how and what to test. We suggest that you thoroughly exercise critical applications and satisfy yourself that the lead member can serve these applications to clients before continuing the roll. For example, you can:

   - Manually relocate CAA services to the lead member. For example, to relocate the application resource named `cluster_lockd` to lead member `provolone`:

     ```
     # caa_relocate cluster_lockd -c provolone
     ```

   - Temporarily modify the default cluster alias selection priority attribute, `selp`, to force the lead member to serve all client requests directed to that alias. For example:

     ```
     # cluamgr -a alias=DEFAULTALIAS,selp=100
     ```

     The lead member is now the end recipient for all connection requests and packets addressed to the default cluster alias.

     From another member or from an outside client, use services such as `telnet` and `ftp` to verify that the lead member can handle alias traffic. Test client access to all important services that the cluster provides.

     When you are satisfied, reset the alias attributes on the lead member to their original values.

7. Perform the postinstall stage (Section 7.6.5).

   On the lead member, run:

   ```
   # clu_upgrade postinstall
   ```

8. Perform the roll stage (Section 7.6.6).

   One at a time, on each member of the cluster that has not rolled,[1] do the following:

   a. Halt the member system and boot it to single-user mode. For example:

      ```
      # shutdown -h now
      >>> boot -fl s
      ```

   b. When the system reaches single-user mode, run the following commands:

[1] The lead member was rolled during the install stage. Therefore, you do not perform the roll stage on the lead member.

```
# init s
# bcheckrc
# lmf reset
```

c. Roll the member:

```
# clu_upgrade roll
```

After preparing the new member for a reboot, `clu_upgrade` displays a prompt that asks whether you want to reboot at this time. Unless there is something specific you want to examine before you reboot, enter **yes**. (If you enter **yes**, note that it may take approximately half a minute before the actual reboot occurs.)

**Note**

The roll actually takes place during the reboot. The `clu_upgrade roll` command sets up the `it`(8) scripts that will be run during the reboot. When you reboot, the `it` scripts roll the member, build a customized kernel, and then reboot again so the member will be running on its new customized kernel. When the member boots its new customized kernel, it has completed its roll and is no longer running on tagged files.

d. Continue to roll members until all members of the cluster have rolled.

9. Perform the switch stage (Section 7.6.7).

After all members have rolled, run the `switch` command on any member.

```
# clu_upgrade switch
```

10. One at a time, reboot each member of the cluster.
11. Perform the clean stage (Section 7.6.8).

Run the following command on any member to remove the tagged (`.Old..`) files from the cluster and complete the upgrade.

```
# clu_upgrade clean
```

## 7.3 Displaying the Status of a Rolling Upgrade

The `clu_upgrade` command provides the following options for displaying the status of a rolling upgrade. You can run status commands at any time.

- To display the overall status of a rolling upgrade: `clu_upgrade -v` or `clu_upgrade -v status`.

- To determine whether you can run a stage: `clu_upgrade check [stage]`. If you do not specify a *stage*, `clu_upgrade` tests whether the next stage can be run.
- To determine whether a stage has started or completed: `clu_upgrade started stage` or `clu_upgrade completed stage`.
- To determine whether a member has rolled: `clu_upgrade check roll memberid`.
- To verify whether tagged files have been created for a layered product: `clu_upgrade tagged check [prod_code [prod_code ...]]`. If you do not specify a product code, `clu_upgrade` inspects all tagged files in the cluster.

**Notes**

During a roll, there might be two versions of the `clu_upgrade` command in the cluster — an older version used by members that have not yet rolled, and a newer version (if included in the update distribution or patch kit). The information that is displayed by the `status` command might differ depending on whether the command is run on a member that has rolled. Therefore, if you run the `status` command on two members, do not be surprised if the format of the displayed output is not the same.

If you run `clu_upgrade status` after running `installupdate`, `clu_upgrade` will display a message indicating that the install stage is complete. However, the install stage is not really complete until you run the `clu_upgrade postinstall` command.

## 7.4 Undoing a Stage

The `clu_upgrade undo` command provides the ability to undo a rolling upgrade that has not completed the switch stage. You can undo any stage except the switch stage and the clean stage. You must undo stages in order; for example, if you decide to undo a rolling upgrade after completing the preinstall stage, you undo the preinstall stage and then undo the setup stage.

**Note**

Before undoing any stage, we recommend that you read the relevant version of the *Cluster Release Notes* to determine whether there are restrictions related to the undoing of any stage.

To undo a stage, use the `undo` command with the stage that you want to undo. The `clu_upgrade` command determines whether the specified stage

is a valid stage to undo. Table 7–4 outlines the requirements for undoing a stage:

**Table 7–4: Undoing a Stage**

| Stage to Undo | Command | Comments |
|---|---|---|
| Setup | `clu_upgrade undo setup` | You must run this command on the lead member. In addition, no members can be running on tagged files when you undo the setup stage.<br>Before you undo the setup stage, use the `clu_upgrade -v status` command to determine which members are running on tagged files. Then use the `clu_upgrade tagged disable` *`memberid`* command to disable tagged files on those members. (See Section 7.7 for information about tagged files and the commands used to manipulate them.)<br>When no members are running on tagged files, run the `clu_upgrade undo setup` command on the lead member. |
| Preinstall | `clu_upgrade undo preinstall` | You must run this command on the lead member. |
| Install | `clu_upgrade undo install` | You can run this command on any member except the lead member.<br>Halt the lead member. Then run the `clu_upgrade undo install` command on any member that has access to the halted lead member's boot disk. When the command completes, boot the lead member. |
| Postinstall | `clu_upgrade undo postinstall` | You must run this command on the lead member. |
| Roll | `clu_upgrade undo roll` *`memberid`* | You can run this command on any member except the member whose roll stage will be undone.<br>Halt the member whose roll stage is being undone. Then run the `clu_upgrade undo roll` *`memberid`* command on any other member that has access to the halted member's boot disk. When the command completes, boot the halted member. The member will now be using tagged files. |

## 7.5 Rolling Upgrade Commands

The `clu_upgrade` command, described in `clu_upgrade`(8), controls the overall flow of a rolling upgrade and ensures that the stages are run in order. During the install stage, you run one or more of `installupdate`, `dupatch`, or `nhd_install` to load and install software. These commands are rolling upgrade aware; they are modified to understand which actions they are allowed to take during the install and roll stages of a rolling upgrade.

When you start a rolling upgrade, the cluster is running the software from the previous release. For the first part of any rolling upgrade, you are running the `clu_upgrade` command that is already installed on the cluster. If a new version is installed during the rolling upgrade, there may be minor

differences in the on-screen display and behavior between the two versions of the command.

The following two tables show at which stages during a rolling upgrade new versions of upgrade commands, if shipped with the kits being installed, become available during a rolling upgrade:[2]

- Table 7–5 maps commands to stages for a rolling upgrade from Version 5.1 to Version 5.1A.
- Table 7–6 maps commands to stages for a rolling upgrade from Version 5.1A to the next release of the operating system and cluster software, a Version 5.1A patch kit, or the next release of the base operating system followed by a patch of the new software within the same rolling upgrade.

**Table 7–5: Stages and clu_upgrade Versions When Performing a Rolling Upgrade to Version 5.1A**

| Stage | Version 5.1 | Version 5.1A | Comments |
|---|---|---|---|
| Preparation | X | | The currently installed (old) version of `clu_upgrade` is always run in this stage. |
| Setup | X | | The currently installed (old) version of `clu_upgrade` is always run in this stage. |
| Preinstall | X | | The currently installed (old) version of `clu_upgrade` is always run in this stage. |
| Install | X | | If performing an update installation, you start the installation with the current cluster's version of `installupdate`; the new version of `installupdate` becomes available during the update installation. The update installation also installs the new version of `clu_upgrade`.<br>A patch kit always installs the latest version of `dupatch`. |
| Postinstall | | X | If a new version of `clu_upgrade` was installed during the install stage, all members use the new version. |
| Roll | | X | If a new version of `clu_upgrade` was installed during the install stage, all members use the new version. |

[2] The `clu_upgrade version` command displays the version number for `clu_upgrade`. Note that the `clu_upgrade` version numbers do not correspond with the version numbers of the operating system.

**Table 7–5: Stages and clu_upgrade Versions When Performing a Rolling Upgrade to Version 5.1A (cont.)**

| Stage | Version 5.1 | Version 5.1A | Comments |
|---|---|---|---|
| Switch | | X | If a new version of `clu_upgrade` was installed during the install stage, all members use the new version. |
| Clean | | X | If a new version of `clu_upgrade` was installed during the install stage, all members use the new version. |

**Table 7–6: Stages and clu_upgrade Versions When Performing a Rolling Upgrade from Version 5.1A**

| Stage | Version 5.1A | Next Release[a] | Comments |
|---|---|---|---|
| Preparation | X | | The currently installed (old) version of `clu_upgrade` is always run in this stage. |
| Setup | X | | The currently installed (old) version of `clu_upgrade` is always run in this stage.<br>If performing an update installation, the new version of the `clu_upgrade` is extracted from the TruCluster Server kit and installed at `/usr/sbin/clu_upgrade`, replacing the old version. Because this replacement is done before tagged files are created, all members will use the new `clu_upgrade` throughout the remainder of the rolling upgrade. |
| Preinstall | | X | If the rolling upgrade includes an update installation, all members use the new version of `clu_upgrade` installed during the setup stage. (Otherwise, members continue to run the current version of `clu_upgrade`.) |
| Install | | X | If the rolling upgrade includes an update installation, all members use the version of `clu_upgrade` installed during the setup stage.<br>During the update installation, a new version of `installupdate` replaces the old one.<br>A patch kit always installs the latest version of `dupatch`.<br>If performing a patch, and if the patch kit includes a new version of `clu_upgrade`, the new version is installed and will be used by all cluster members starting with the postinstall stage. |

**Table 7–6: Stages and clu_upgrade Versions When Performing a Rolling Upgrade from Version 5.1A (cont.)**

| Stage | Version 5.1A | Next Release[a] | Comments |
|---|---|---|---|
| Postinstall | | X | If a new version of `clu_upgrade` was installed in either the setup stage or the install stage, all members use the new version. |
| Roll | | X | If a new version of `clu_upgrade` was installed in either the setup stage or the install stage, all members use the new version. |
| Switch | | X | If a new version of `clu_upgrade` was installed in either the setup stage or the install stage, all members use the new version. |
| Clean | | X | If a new version of `clu_upgrade` was installed in either the setup stage or the install stage, all members use the new version. |

[a] The next release of Tru64 UNIX and TruCluster Server, a patch kit for Version 5.1A, or the installation of an NHD kit on Version 5.1A.

## 7.6 Rolling Upgrade Stages

The following sections describe each of the rolling upgrade stages.

**Note**

These sections only describe the stages. Use the procedure in Section 7.2 to perform a rolling upgrade.

- Preparation stage (Section 7.6.1)
- Setup stage (Section 7.6.2)
- Preinstall stage (Section 7.6.3)
- Install stage (Section 7.6.4)
- Postinstall stage (Section 7.6.5)
- Roll stage (Section 7.6.6)
- Switch stage (Section 7.6.7)
- Clean stage (Section 7.6.8)

### 7.6.1 Preparation Stage

| Command | Where Run | Run Level |
|---|---|---|
| `clu_upgrade -v check setup` *lead_memberid* | any member | multiuser mode |

During the preparation stage, you back up all important cluster data and verify that the cluster is ready for a roll. Before beginning a rolling upgrade, do the following:

1. Choose one member of the cluster as the first member to roll. This member, known as the **lead member**, must have direct access to the root (`/`), `/usr`, `/var`, and, if used, `i18n` file systems.

   Make sure that the lead member can run any critical applications. You can test these applications after you update this member during the install stage, but before you roll any other members. If there is a problem, you can try to resolve it on this member before you continue. If there is a problem that you cannot resolve, you can undo the rolling upgrade and return the cluster to its pre-roll state. (Section 7.4 describes how to undo rolling upgrade stages.)

2. Back up the clusterwide root (`/`), `/usr`, and `/var` file systems. The backups should include all member-specific files in these file systems. If the cluster has a separate `i18n` file system, back up that file system. In addition, back up any other file systems that contain critical user or application data.

   **Note**

   If you perform an incremental or full backup of the cluster during a rolling upgrade, make sure to perform the backup on a member that is not running on tagged files. If you back up from a member that is using tagged files, you will only back up the contents of the `.Old..` files. Because the lead member never uses tagged files, you can back up the cluster from the lead member (or any other member that has rolled) during a rolling upgrade.

   Most sites have automated backup procedures. If you know that an automatic backup will take place while the cluster is in the middle of a rolling upgrade, make sure that backups are done on the lead member or on a member that has rolled.

3. If you plan to run the `installupdate` command in the install stage, remove any blocking layered products listed in Table 7–7 that are installed on the cluster.

4. Run the `clu_upgrade -v check setup` *`lead_memberid`* command, which verifies that:
   - No rolling upgrade is in progress.
   - All members are running the same versions of the base operating system and cluster software.
   - No members are running on tagged files.
   - There is adequate free disk space.
5. Verify that each system's firmware will support the new software. Update firmware as needed before starting the rolling upgrade.

A cluster can continue to operate during a rolling upgrade because there are two copies of the operating system and cluster software files. (There is only one copy of shared configuration files so that changes made by any member are visible to all members.) This approach makes it possible to run two different versions of the base operating system and the cluster software at the same time in the same cluster. The trade-off is that, before you start an upgrade, you must make sure that there is adequate free space in each of the clusterwide root (`/`), `/usr`, and `/var` file systems, and, if there is a separate domain for the Worldwide Language Support (WLS) subsets, in the `i18n` file system.

A rolling upgrade has the following disk space requirements:

- At least 50 percent free space in root (`/`), `cluster_root#root`.
- At least 50 percent free space in `/usr`, `cluster_usr#usr`.
- At least 50 percent free space in `/var`, `cluster_var#var`, plus, if updating the operating system, an additional 425 MB to hold the subsets for the new version of the base operating system.
- If there is a separate `i18n` domain for the WLS subsets, at least 50 percent free space in that domain.
- No tagged files are placed on member boot partitions. However, programs might need free space when moving kernels to boot partitions. We recommend that you reserve at least 50 MB free space on each member's boot partition.

**Note**

You cannot use the `addvol` command to add volumes to a member's root domain (the a partition on the member's boot disk). Instead, you must delete the member from the cluster, use `diskconfig` or SysMan to configure the disk appropriately, and then add the member back into the cluster.

- If installing a patch kit, see the *Patch Kit Installation Instructions* that came with your patch kit to find the amount of space you will need to install that kit. If installing an NHD kit, see the *New Hardware Delivery Release Notes and Installation Instructions* that came with your NHD kit to find the amount of space you will need to install that kit.

If a file system needs more free space, use AdvFS utilities such as `addvol` to add volumes to domains as needed. For information on managing AdvFS domains, see the Tru64 UNIX *AdvFS Administration* manual. (The AdvFS Utilities require a separate license.) Note that you can expand the clusterwide root (`/`) domain.

**Note**

The `clu_upgrade` command performs a disk space check at the start of a rolling upgrade. However, nothing prevents a cluster member from consuming disk space during a rolling upgrade, and thus creating a situation where a later stage might not have enough disk space.

Remember that disk space is dynamic. If you know that a member will be consuming disk space during a rolling upgrade, add additional space before you start the upgrade.

### 7.6.2 Setup Stage

| Command | Where Run | Run Level |
|---|---|---|
| `clu_upgrade setup` *`lead_memberid`* | any member | multiuser mode |

The setup stage performs the `clu_upgrade check setup` command, creates tagged files, and prepares the cluster for the roll.

The `clu_upgrade setup` *`lead_memberid`* command performs the following tasks:

- Creates the rolling upgrade log file, `/cluster/admin/clu_upgrade.log`. (Section C.3 contains a sample `clu_upgrade.log` file.)
- Makes the `-v check setup` tests listed in Section 7.6.1.
- Prompts you to indicate whether to perform an update installation, install a patch kit, install an NHD kit, or a combination thereof. The following example shows the menu displayed by the TruCluster Server Version 5.1A `clu_upgrade` command:

```
What type of rolling upgrade will be performed?

Selection   Type of Upgrade
------------------------------------------------------------------
```

```
   1      An upgrade using the installupdate command
   2      A patch using the dupatch command
   3      A new hardware delivery using the nhd_install command
   4      All of the above
   5      None of the above
   6      Help
   7      Display all options again
------------------------------------------------------------
Enter your Choices (for example, 1 2 2-3):
```

**Note**

If you are upgrading to TruCluster Server Version 5.1A from TruCluster Server Version 5.1, you will not see this menu because the cluster is still running on Version 5.1.

- If you specify an update installation, copies the relevant kits onto disk:
  - If performing an update installation, copies the cluster kit to `/var/adm/update/TruClusterKit` so that the kit will be available to the `installupdate` command during the install stage. (The `installupdate` command copies the operating system kit to `/var/adm/update/OSKit` during the install stage.) The `clu_upgrade` command prompts for the absolute pathname for the TruCluster Server kit location. On a TruCluster Server Version 5.1A cluster, when performing a rolling upgrade that includes an update installation, remember to mount the TruCluster Server kit before running the `clu_upgrade setup` command.
  - On a TruCluster Server Version 5.1A cluster, if performing an NHD installation, uses the `nhd_install` command to copy the NHD kit to `/var/adm/update/NHDKit`

**Caution**

The files in `/var/adm/update` are critical to the roll process. Do not remove or modify files in this directory. Doing so can cause a rolling upgrade to fail.

- Creates the mandatory set of tagged files for the `OSF` (base), `TCR` (cluster), and `IOS` (Worldwide Language Support) products.

**Caution**

If, for any reason, during an upgrade you need to create tagged files for a layered product, see Section 7.7.

- Sets the `sysconfigtab` variable `rolls_ver_lookup=1` on all members except the lead member. When `rolls_ver_lookup=1`, a member

uses tagged files. As a result, the lead member can upgrade while the remaining members run on the `.Old..` files from the current release.

- Prompts you to reboot all cluster members except the lead member. When the `setup` command completes, reboot these members one at a time so that the cluster can maintain quorum. This reboot is required for each member that will use tagged files in the mixed-version cluster. When the reboots complete, all members except the lead member are running on tagged files.

### 7.6.3 Preinstall Stage

| Command | Where Run | Run Level |
|---|---|---|
| `clu_upgrade preinstall` | lead member | multiuser mode |

The purpose of the preinstall stage is to verify that the cluster is ready for the lead member to run one or more of the `installupdate`, `dupatch`, or `nhd_install` commands.

The `clu_upgrade preinstall` command performs the following tasks:

- Verifies that the command is being run on the lead member, that the lead member is not running on tagged files, and that any other cluster members that are up are running on tagged files.
- (Optional) Verifies that tagged files are present, that they match their product's inventory files, and that each tagged file's AdvFS property is set correctly. (This process can take a while, but not as long as it does to create the tagged files in the setup stage. Table 7–3 provides time estimates for each stage.)
- Makes on-disk backup copies of the lead member's member-specific files.

### 7.6.4 Install Stage

| Command | Where Run | Run Level |
|---|---|---|
| `installupdate` | lead member | single-user mode |
| `dupatch` | lead member | single-user or multiuser mode |
| `nhd_install` | lead member | single-user mode |

If your current cluster is running TruCluster Server Version 5.1, you can perform one of the tasks or combinations of tasks listed in Table 7–1.

If your current cluster is running TruCluster Server Version 5.1A, you can perform one of the tasks or combinations of tasks listed in Table 7–2.

The install stage starts when the `clu_upgrade preinstall` command completes, and continues until you run the `clu_upgrade postinstall` command.

**Note**

If you run `clu_upgrade status` after running `installupdate`, `clu_upgrade` displays a message indicating that the install stage is complete. However, the install stage is not really complete until you run the `clu_upgrade postinstall` command.

The lead member must be in single-user mode to run the `installupdate` command or the `nhd_install` command; single-user mode is recommended for the `dupatch` command. When taking the system to single-user mode, you must halt the system and then boot it to single-user mode.

When the system is in single-user mode, run the `init s`, `bcheckrc`, and `lmf reset` commands before you run the `installupdate`, `dupatch`, or `nhd_install` commands. See the Tru64 UNIX *Installation Guide*, the Tru64 UNIX and TruCluster Server *Patch Kit Installation Instructions*, and the Tru64 UNIX *New Hardware Delivery Release Notes and Installation Instructions* for information on how to use these commands.

**Notes**

During the install stage, you cannot run a `dupatch` command followed by an `installupdate` command. To patch the current software before you perform a rolling upgrade, you must perform two complete rolling upgrade operations: one to patch the current software, and one to perform the update installation.

If an NHD installation is part of a rolling upgrade that includes an update installation, you do not have to manually run `nhd_install`; the `installupdate` command will install the NHD kit. Otherwise, use the `nhd_install` command copied by `clu_upgrade` during the setup stage: `/var/adm/update/NHDKit/nhd_install`.

### 7.6.5 Postinstall Stage

| Command | Where Run | Run Level |
|---|---|---|
| `clu_upgrade postinstall` | lead member | multiuser mode |

The postinstall stage verifies that the lead member has completed an update installation, a patch, or an NHD installation. If an update installation was

performed, `clu_upgrade postinstall` verifies that the lead member has rolled to the new version of the base operating system.

### 7.6.6 Roll Stage

| Command | Where Run | Run Level |
|---|---|---|
| `clu_upgrade roll` | member being rolled | single-user mode |

The lead member was upgraded in the install stage. The remaining members are upgraded one at a time in the roll stage.

The `clu_upgrade roll` command performs the following tasks:

- Verifies that the member is not the lead member, that the member has not already been rolled, and that the member is in single-user mode.
- Backs up the member's member-specific files.
- Sets up the `it`(8) scripts that will be run on reboot to perform the roll.
- Reboots the member. During this boot, the `it` scripts roll the member, build a customized kernel, and reboot with the customized kernel.

**Note**

If you need to add a member to the cluster during a rolling upgrade, you must add the member from a member that has completed its roll.

If a member goes down (and cannot be repaired and rebooted) before all members have rolled, you must delete the member to complete the roll of the cluster. However, if you have rolled all members but one, and this member goes down before it has rebooted in the roll stage, you must delete this member and then reboot any other member of the cluster. (The `clu_upgrade` command runs during reboot and tracks the number of members rolled versus the number of members currently in the cluster; `clu_upgrade` marks the roll stage as completed when the two values are equal. That is why, in the second case, rebooting another member completes the roll stage and lets you continue the rolling upgrade.)

### 7.6.7 Switch Stage

| Command | Where Run | Run Level |
|---|---|---|
| `clu_upgrade switch` | any member | multiuser mode |

The switch stage sets the active version of the software to the new version, which results in turning on any new features that had been deliberately

disabled during the rolling upgrade. (See Section 7.8 for a description of active version and new version.)

The `clu_upgrade switch` command performs the following tasks:

- Verifies that all members have rolled, that all members are running the same versions of the base operating system and cluster software, and that no members are running on tagged files.
- Sets the new version ID in each member's `sysconfigtab` file and running kernel.
- Sets the active version to the new version for all cluster members.

**Note**

After the switch stage completes, you must reboot each member of the cluster, one at a time.

### 7.6.8 Clean Stage

| Command | Where Run | Run Level |
|---|---|---|
| `clu_upgrade clean` | any member | multiuser mode |

The clean stage removes the tagged (`.Old..`) files from the cluster and completes the upgrade.

The `clu_upgrade clean` command performs the following tasks:

- Verifies that the switch stage has completed, that all members are running the same versions of the base operating system and cluster software, and that no members are running on tagged files.
- Removes all `.Old..` files.
- Removes any on-disk backup archives that `clu_upgrade` created.
- If the directory exists, recursively deletes `/var/adm/update/TruClusterKit`, `/var/adm/update/OSKit`, and `/var/adm/update/NHDKit`.
- If an update installation was performed, gives you the option of running the Update Administration Utility (`updadmin`) to manage the files that were saved during an update installation.
- Creates an archive directory for this upgrade, `/cluster/admin/clu_upgrade/history/`*release_version*, and moves the `clu_upgrade.log` file to the archive directory.

## 7.7 Tagged Files

A rolling upgrade updates the software on one cluster member at a time. In order to support two versions of software within the cluster during a roll, `clu_upgrade` creates a set of **tagged files** in the setup stage.

A tagged file is a copy of a current file with `.Old..` prepended to the copy filename, and an AdvFS property (`DEC_VERSION_TAG`) set on the copy. For example, the tagged file for the `vdump` command is named `/sbin/.Old..vdump`. Tagged files are created in the same file system as the original files. This is why you must have adequate free disk space before beginning a rolling upgrade.

Whether a member is running on tagged files is controlled by that member's `sysconfigtab rolls_ver_lookup` variable. The upgrade commands set the value to `1` when a member must run on tagged files, and to `0` when a member must not run on tagged files.

If a member's `sysconfigtab rolls_ver_lookup` attribute is set to `1`, pathname resolution includes determining whether a specified filename has a `.Old..`*filename* copy and whether the copy has the `DEC_VERSION_TAG` property set on it. If both conditions are met, the requested file operation is transparently diverted to use the `.Old..`*filename* version of the file. Therefore, if the `vdump` command is issued on a member that has not rolled, the `/sbin/.Old..vdump` file is executed; if the command is issued on a member that has rolled, the `/sbin/vdump` file is executed. The only member that never runs on tagged files is the lead member (the first member to roll).

**Note**

File system operations on directories are not bound by this tagged file restraint. For example, an `ls` of a directory on any cluster member during a rolling upgrade lists both versions of a file. However, the output of an `ls -ail` command on a member that has not rolled is different from the output on a member that has rolled. In the following examples the `ls -ail` command is run first on a member that has not rolled and then on a member that has rolled. (The `awk` utility is used to print only the inode, size, month and day timestamp, and name of each file.)

The following output from the `ls` command is taken from a cluster member running with tags before it has rolled. The tagged files are the same as their untagged counterparts (same inode, size, and timestamp). When this member runs the `hostname` command, it runs the tagged version (inode 3643).

```
# cd /sbin
# ls -ail hostname .Old..hostname ls .Old..ls init .Old..init |\
awk '{printf("%d\t%d\t%s %s\t%s\n",$1,$6,$7,$8,$10)}'
```

```
3643   16416    Aug 24   .Old..hostname
3648   395600   Aug 24   .Old..init
3756   624320   Aug 24   .Old..ls
3643   16416    Aug 24   hostname
3648   395600   Aug 24   init
3756   624320   Aug 24   ls
```

The following output from the `ls` command is taken from a cluster member running without tags after it has rolled. The tagged files now differ from their untagged counterparts (different inode, size, and timestamp). When this member runs the `hostname` command, it runs the non-tagged version (inode 1370).

```
# cd /sbin
# ls -ail hostname .Old..hostname ls .Old..ls init .Old..init |\
awk '{printf("%d\t%d\t%s %s\t%s\n",$1,$6,$7,$8,$10)}'

3643   16416    Aug 24   .Old..hostname
3648   395600   Aug 24   .Old..init
3756   624320   Aug 24   .Old..ls
1187   16528    Mar 12   hostname
1370   429280   Mar 12   init
1273   792640   Mar 12   ls
```

---

After you create tagged files in the setup stage, we recommend that you run any administrative command, such as `tar`, from a member that has rolled. You can always run commands on the lead member because it never runs on tagged files.

The following rules determine which files have tagged files automatically created for them in the setup stage:

- Tagged files are created for inventory files for the following product codes: base operating system (`OSF`), TruCluster Server (`TCR`), and Worldwide Language Support (`IOS`). (The subsets for each product use that product's three-letter product code as a prefix for each subset name. For example, TruCluster Server subset names start with the TruCluster Server three-letter product code: `TCRBASE520`, `TCRMAN520`, and `TCRMIGRATE520`.)
- By default, files that are associated with other layered products do not have tagged files created for them. Tagged files are created only for layered products that have been modified to support tagged files during a rolling upgrade.

**Caution**

Unless a layered product's documentation specifically states that you can install a newer version of the product on the first rolled member, and that the layered product knows what actions to take in a mixed-version cluster, we strongly

recommend that you do not install either a new layered product or a new version of a currently installed layered product during a rolling upgrade.

---

The `clu_upgrade` command provides several `tagged` command options to manipulate tagged files: `check`, `add`, `remove`, `enable`, and `disable`. When dealing with tagged files, take the following into consideration:

- During a normal rolling upgrade you do not have to manually add or remove tagged files. The `clu_upgrade` command calls the `tagged` commands as needed to control the creation and removal of tagged files.
- If you run a `clu_upgrade tagged` command, run the `check`, `add`, and `remove` commands on a member that is not running on tagged files; for example, the lead member. You can run the `disable` and `enable` commands on any member.
- The target for a `check`, `add`, or `remove` tagged file operation is a product code that represents an entire product. The `clu_upgrade tagged` commands operate on all inventory files for the specified product or products. For example, the following command verifies the correctness of all the tagged files created for the `TCR` kernel layered product (the TruCluster Server subsets):

  ```
  # clu_upgrade tagged check TCR
  ```

  If you inadvertently remove a `.Old..` copy of a file, you must create tagged files for the entire layered product to re-create that one file. For example, the `vdump` command is in the `OSFADVFS`*xxx* subset, which is part of the `OSF` product. If you mistakenly remove `/sbin/.Old..vdump`, run the following command to re-create tagged files for the entire layered product:

  ```
  # clu_upgrade tagged add OSF
  ```

- The `enable` and `disable` commands enable or disable the use of tagged files by a cluster member. You do not have to use `enable` or `disable` during a normal rolling upgrade.

  The `disable` command is useful if you have to undo the setup stage. Because no members can be running with tagged files when undoing the setup stage, you can use the `disable` command to disable tagged files on any cluster member that is currently running on tagged files. For example, to disable tagged files for a member whose ID is 3:

  ```
  # clu_upgrade tagged disable 3
  ```

  The `enable` command is provided in case you make a mistake with the `disable` command.

## 7.8 Version Switch

A version switch manages the transition of the **active version** to the **new version** of an operating system. The active version is the one that is currently in use. The purpose of a version switch in a cluster is to prevent the introduction of potentially incompatible new features until all members have been updated. For example, if a new version introduces a change to a kernel structure that is incompatible with the current structure, you do not want cluster members to use the new structure until all members have updated to the version that supports it.

At the start of a rolling upgrade, each member's active version is the same as its new version. When a member rolls, its new version is updated. After all members have rolled, the switch stage sets the active version to the new version on all members. At the completion of the upgrade, all members' active versions are once again the same as their new versions. The following simple example uses an active version of `1` and a new version of `2` to illustrate the version transitions during a rolling upgrade.

```
All members at start of roll:   active (1)  = new (1)
Each member after its roll:     active (1) != new (2)
All members after switch stage: active (2)  = new (2)
```

The `clu_upgrade` command uses the `versw` command, which is described in `versw`(8), to manage version transitions. The `clu_upgrade` command manages all the version switch activity when rolling individual members. In the switch stage, after all members have rolled, the following command completes the transition to the new software:

```
# clu_upgrade switch
```

## 7.9 Rolling Upgrade and Layered Products

This section discusses the interaction of layered products and rolling upgrades:

- General guidelines (Section 7.9.1)
- Blocking layered products (Section 7.9.2)

### 7.9.1 General Guidelines

The `clu_upgrade setup` command prepares a cluster for a rolling upgrade of the operating system. Do not use the `setld` command to load software onto the cluster between performing the `clu_upgrade setup` command and rolling the first cluster member to the new version. If you install software between performing the `clu_upgrade setup` command and rolling a cluster member to the new version, the new files will not have been

processed by `clu_upgrade setup`. As a result, when you roll the first cluster member, these new files will be overwritten.

If you must load software:

- Wait until at least one member has rolled.
- Install the software on a member that has rolled.

Some layered products have provided information about how to handle them during a rolling upgrade. You can find this information at the following URL:

**`http://www.tru64unix.compaq.com/docs/pub_page/lp_roll.html`**

### 7.9.2 Blocking Layered Products

A blocking layered product is a product that prevents the `installupdate` command from completing. Blocking layered products must be removed from the cluster before starting a rolling upgrade that will include running the `installupdate` command. You do not have to remove blocking layered products when performing a rolling upgrade solely to patch the cluster or install an NHD kit.

Table 7–7 lists blocking layered products for this release.

**Table 7–7: Blocking Layered Products**

| Product Code | Description |
|---|---|
| 3X0 | Open3D |
| 4DT | Open3D |
| ATM | Atom Advanced Developers Kit |
| DCE | Distributed Computing Environment |
| DNA | DECnet |
| DTA | Developer's Toolkit (Program Analysis Tools) |
| DTC | Developer's Toolkit (C compiler) |
| MME | Multimedia Services |
| O3D | Open 3D |
| PRX | PanoramiX Advanced Developers Kit |

**Notes**

The three-letter product codes are the first three letters of subset names. For example, a subset named ATMBASE*xxx* is part of the ATM product (Atom Advanced Developers Kit), which is a blocking layered product. However, a subset named

OSFATMBIN*xxx* contains the letters ATM, but the subset is not part of a blocking layered product; it is a subset in the OSF product (the base operating system).

When a blocking layered product is removed as part of the rolling upgrade, it is removed for all members. Any services that rely on the blocking product will not be available until the roll completes and the blocking layered product is reinstalled.

---

## 7.10 Rolling Upgrade and RIS

When performing the install stage of a rolling upgrade, you can load the base operating system subsets from a CD-ROM or from a Remote Installation Services (RIS) server.

**Note**

You can use RIS only to load the base operating system subsets.

---

To use RIS, you must register both the lead member and the default cluster alias with the RIS server. When registering for operating system software, you must provide a hardware address for each host name. Therefore, you must create a hardware address for the default cluster alias in order to register the alias with the RIS server. (RIS will reject an address that is already in either of the RIS server's `/etc/bootptab` or `/var/adm/ris/clients/risdb` files.)

If your cluster uses the cluster alias virtual MAC (vMAC) feature, register that virtual hardware address with the RIS server as the default cluster alias's hardware address. If your cluster does not use the vMAC feature, you can still use the algorithm that is described in the vMAC section of *Cluster Administration* to manually create a hardware address for the default cluster alias.

A vMAC address consists of a prefix (the default is AA:01) followed by the IP address of the alias in hexadecimal format. For example, the default vMAC address for the default cluster alias `deli` whose IP address is `16.140.112.209` is `AA:01:10:8C:70:D1`. The address is derived in the following manner:

```
Default vMAC prefix:       AA:01
Cluster Alias IP Address:  16.140.112.209
IP address in hex. format: 10.8C.70.D1
vMAC for this alias:       AA:01:10:8C:70:D1
```

Another method for creating a hardware address is to append an arbitrary string of eight hexadecimal numbers to the default vMAC prefix, `AA:01`.

For example, `AA:01:00:00:00:00`. Make sure that the address is unique within the area served by the RIS server. If you have more than one cluster, remember to increment the arbitrary hexadecimal string when adding the next alias. (The vMAC algorithm is useful because it creates an address that has a high probability of being unique within your network.)

# 8

# Upgrading from TruCluster Software Version 1.5 or Version 1.6

This chapter discusses the issues involved when upgrading to TruCluster Server Version 5.1A from TruCluster Production Server Software or TruCluster Available Server Software Version 1.5 or Version 1.6. (Although the primary focus in this chapter is upgrading Production Server and Available Server clusters, Section 8.9 provides information on upgrading a cluster running Memory Channel Software.)

**Note**

For simplicity, this chapter uses the term **ASE** to refer to a Version 1.5 or 1.6 TruCluster Production Server Software cluster or TruCluster Available Server Software, and the term **cluster** to refer to the new TruCluster Server Version 5.1A cluster.

This chapter discusses three upgrade paths for upgrading from TruCluster Production Server Software or TruCluster Available Server Software Version 1.5 or Version 1.6:

- Option 1: Create a separate cluster with new systems and new storage hardware. When the new cluster is fully configured and tested, migrate data from the ASE to the new cluster.
- Option 2: Create a separate cluster with new systems and enough new storage to create the cluster and test applications. When the new cluster is fully configured and tested, physically move the old storage from the ASE to the new cluster.
- Option 3: Upgrade the existing ASE using its hardware and storage. Remove and disconnect one member from the ASE, install and configure Tru64 UNIX Version 5.1A on that system, shut down the remaining systems in the ASE, connect the ASE's storage to the Tru64 UNIX Version 5.1A system, configure the storage, create a single-member cluster, and then add the other systems to the new cluster.

**Note**

For TruCluster Production Server Software or TruCluster Available Server Software products earlier than Version 1.5, you

must perform a full installation of Tru64 UNIX Version 5.1A and TruCluster Server Version 5.1A.

See Table 1–1 for the list of supported upgrade paths for TruCluster products.

---

You can also choose to design your own customized procedure based on your current configuration and your goals. Section 8.8 describes a case study that uses a modified version of the Option 3 procedure. You can find another case study at the following URL, which contains a link to the description of an upgrade of an in-house production-level cluster from TruCluster Production Server Version 1.6 to TruCluster Server Version 5.0A plus a patch kit:

**http://www.tru64unix.compaq.com/docs/highavail/index.htm**

This chapter provides the following information:

- How to decide whether to create a separate cluster or to upgrade an existing cluster (Section 8.1)
- A list of storage and cluster interconnect restrictions (Section 8.2)
- How to prepare for an upgrade (Section 8.3)
- A description of the upgrade scripts used with Option 2 and Option 3 (Section 8.4)
- Option 1: Creation of a separate cluster — use new systems and new storage (migrate data only) (Section 8.5)
- Option 2: Creation of a separate cluster — use new systems and migrate existing storage (move physical storage) (Section 8.6)
- Option 3: Upgrade of an existing ASE — use its systems and storage (Section 8.7)
- An upgrade case study, which uses a modified version of the option 3 procedure (Section 8.8)
- An approach to upgrading a cluster running the TruCluster Server Memory Channel Software product (Section 8.9)

To prepare for an upgrade, read Chapter 1 through Chapter 5.

## 8.1 Determine Whether to Create a Separate Cluster or Upgrade an Existing Cluster

Of the many possible ways to approach an upgrade, this chapter focuses on three distinct methods for upgrading an ASE to a cluster:

1. Option 1: Create a separate cluster with all new system hardware and storage. This cluster can be a minimal two-node configuration designed solely for testing up to a full-scale, production-level cluster.

   This approach (new systems, new storage) lets you create a new cluster without any of the limitations of the current hardware configuration, or any restrictions imposed by the Tru64 UNIX Version 4.*x* operating systems or the Version 1.5 or 1.6 TruCluster Software products. For example, you can build a new no-single-point-of-failure (NSPOF) cluster using HSG80 controllers and Fibre Channel.

   You then run the old ASE side-by-side with the new Version 5.1A cluster. This configuration lets you extensively test applications on the new cluster while still serving clients from the existing ASE. When you are satisfied with the new cluster, migrate application data (not storage hardware) from the old ASE to the new cluster.

   Using a separate, but parallel, cluster with new systems and new storage has the lowest risk because no hardware is shared and no storage is moved from the ASE to the new cluster. Because the new cluster has no legacy hardware, it can take advantage of all TruCluster Server Version 5.1A features; for example, configuring hardware multipathing for redundancy.

   Section 8.5 discusses creating a separate cluster.

2. Option 2: Create a separate cluster with some new storage, but move existing storage from the ASE to the new cluster.

   This approach (new systems, old storage) provides a degree of separation in that you configure new systems and perform application testing without affecting the ASE. However, any limitations in the current ASE storage configuration will become part of the new cluster. These limitations might prevent the cluster from taking advantage of new features; for example, the older storage might not support multipathing.

   Section 8.6 describes how to move storage from an existing ASE to a new cluster, and configure the storage on the new cluster. The section includes a procedure that uses scripts to help migrate and configure storage.

**Note**

One trade-off to consider is data migration: For your site, determine the advantages and disadvantages of copying data (Option 1) versus physically moving storage (Option 2). For example, if an ASE has so much shared storage that the amount of time it would take to back up and restore data to the new cluster is unacceptable, you might decide that physically moving storage is the better solution for your environment.

With Option 1, both the ASE and the cluster can have fully replicated sets of application data, which allows for extensive tuning and testing of the application environment on the new cluster before switching client services from the ASE to the cluster.

With Option 2, the first time the new cluster has access to all the application data from the ASE is when you physically connect the ASE's storage to the cluster. At some point you must shut down the ASE and connect the physical storage to the new cluster.

3. Option 3: Use the existing ASE hardware as the basis for the new cluster, adding Memory Channel hardware (if desired) and storage, and migrating storage and systems as needed.

**Note**

TruCluster Server Version 5.1A supports LAN cluster interconnects. See the *Cluster LAN Interconnect* manual for details.

This approach (old systems, old hardware) has the lowest monetary cost. However, it has, by definition, more risk than running a fully separate cluster. You must remove a member from the working ASE, which reduces the number of systems available to serve clients. Because application testing takes place on a single-member cluster rather than on a multi-member cluster, you will not have the opportunity to do application failover testing.

Section 8.7 discusses this approach. This section also includes a procedure that uses scripts to help migrate and configure the storage known to ASE services.

If you do not know which approach to take, we recommend that you read the entire chapter before making a decision. Use the information in this chapter

as a starting point for designing an approach and a procedure that fit your needs. Section 8.8 and the following URL each describe a upgrade case study.

**http://www.tru64unix.compaq.com/docs/highavail/index.htm**

Reading these case studies might help you decide which upgrade approach to take.

## 8.2 Storage and Cluster Interconnect Restrictions

This section lists general restrictions that you must observe during an upgrade from an ASE to a TruCluster Server cluster. The restrictions are more applicable when upgrading existing hardware, but during any upgrade, the ASE and the new cluster must never both be active on the same storage bus or on the same cluster interconnect.

- **Do not have running systems from the ASE and the new cluster on the same storage bus.**

  Storage is accessed either by the ASE or by the new cluster, not by both. If systems from both the ASE and the new cluster can access the same storage devices, there is a likelihood of data corruption. When migrating storage, either physically disconnect ASE systems from shared storage or make sure the systems are halted and turned off.

- **Do not have running systems from the ASE and the new cluster on the same cluster interconnect.**

  The cluster interconnect hardware is actively connected to the ASE or the new cluster, never to both. If both the ASE and the new cluster are connected to the same cluster interconnect, booting the wrong system can cause a variety of problems (for example, machine checks). When migrating systems, either physically disconnect the existing ASE systems from the cluster interconnect or make sure the systems are halted and turned off.

- **Do not modify the storage topology from the time you halt the ASE system that will become the first member of the new cluster until you map all existing devices to their new-style device names on the new cluster.**

  Changing the storage topology after you start the upgrade but before you map devices to the new-style device names introduces devices that are known only to the Tru64 UNIX Version 5.1A system. This makes it more difficult to ensure that the device mappings are correct. This restriction applies mainly to an upgrade that uses existing hardware. However, if you plan to physically connect existing storage to a separate cluster, the same restriction applies.

## 8.3 Preparing for an Upgrade

The following sections describe how to prepare for an upgrade:

- General requirements (Section 8.3.1)
- Hardware and storage topology requirements (Section 8.3.2)
- File-system requirements (Section 8.3.3)

### 8.3.1 General Requirements

Most of the preparation involves understanding the differences between the existing ASE and the new features and architecture in TruCluster Version 5.1A. When you understand the differences, you can design an upgrade approach that fits your site-specific needs.

- Read the Tru64 UNIX *Technical Overview* and *System Administration* manuals for information on the new AdvFS format, extended SCSI support, and new device naming conventions.
- Read the TruCluster Server *Cluster Hardware Configuration* and *Cluster Technical Overview* manuals to gain a solid understanding of the recommended TruCluster Server Version 5.1A configurations, and to learn about the critical ways in which a TruCluster Server Version 5.1A cluster configuration differs from previous TruCluster configurations, both operationally and with respect to no-single-point-of-failure (NSPOF).
- Read the *Cluster Highly Available Applications* manual, which explains how to run highly available applications in a Version 5.1A cluster. TruCluster Server does not use the ASE paradigm to provide highly available services: there is no `asemgr` command, and there is no `asecdb` database. Instead, TruCluster Server uses the Cluster File System (CFS), cluster application availability (CAA), and cluster alias features to provide highly available applications.

**Note**

If you use third-party applications from Oracle, Informix, or any other vendor, consult with that application's vendor.

- If you plan to use hardware RAID to mirror file systems in the new cluster, the upgrade is a good time to add RAID hardware and take advantage of multipathing.
- Make an extremely detailed plan. Draw diagrams. Use the `cluster_map_create -full` command to create a configuration map of the current environment. Use the cluster monitor (`cmon`) to display and print the map.

If you plan to migrate storage, label all cables and storage. Acquire any needed hardware. Make copies of site-specific files. Decide when and what to back up. Read the sections for all the upgrade options and look at the procedures. Then create a detailed procedure that fits your site and your method of upgrade. Even if you plan to create a separate cluster, reading the procedures will give you an idea of what is involved when migrating storage.

The following URL takes you to the administrator's text log from an Option 3 upgrade of a two-member, in-house cluster:

```
http://www.tru64unix.compaq.com/docs/highavail/migration/migration_log.htm
```

The cluster was upgraded from TruCluster Software Production Server Version 1.6 to TruCluster Server Version 5.0A plus a patch kit. Although this is not to Version 5.1A upgrade, it uses the same planning approach and basic procedural steps.

- Use the checklists in Appendix A to keep a record of the host names, disks, and IP addresses for the new cluster.

### 8.3.2 Hardware and Storage Topology Requirements

The following list contains the most common hardware requirements that can affect an upgrade from an existing ASE. These requirements apply to the Option 2 and Option 3 upgrade paths, which use some or all of the current ASE hardware in the new cluster. The *Cluster Hardware Configuration* manual is the definitive source for TruCluster Server Version 5.1A hardware configurations.

- (Option 2) When configuring the new cluster, remember that each system will need an open SCSI adapter in order to connect the storage from the ASE to the new cluster. (An alternative is to move the ASE adapters with the storage.)
- (Option 2 and Option 3) When the upgrade path involves migrating physical storage, the ASE must have a symmetrical shared storage configuration. (Each shared device used by an ASE service is known to all ASE members by the same special file name; for example, on all members, `rz17c` refers to the same physical device.)

  In addition, a Production Server environment must contain only one ASE.

  The reason for these requirements is that the automated scripts described in Section 8.4 depend on the current ASE environment having unique AdvFS domain names, LSM volumes, and device special file names for all shared storage used by ASE services. These scripts automate the migration of the storage currently in use by ASE services to the new cluster. (A site with an ASE that does not meet these restrictions

can use the manual device name mapping and storage configuration steps in Appendix E.)

If you have not created a configuration map of the current ASE, or if that map is outdated, perform the preparatory steps listed in `cluster_map_create`(8), and then run the following command on one member to create the `/etc/CCM` file:

```
# cluster_map_create cluster_name -full
```

Use the cluster monitor (`cmon`) to display the cluster configuration map. Use this information to decide whether you can use the migration scripts to map storage during an upgrade. (See either the TruCluster Software Products Version 1.5 or Version 1.6 *Administration* manual for more information on the `cluster_map_create` and `cmon` commands.)

- (Option 3) The ASE systems must be systems that are supported by TruCluster Server Version 5.1A. See the TruCluster Server Version 5.1A *Software Product Description* (SPD) for supported-hardware information.
- (Option 3) If the new cluster will require additional storage hardware, we recommend that you add this hardware to the ASE before beginning the upgrade. The new cluster should have shared storage for clusterwide file systems, member boot disks, and, optionally, a quorum disk. We recommend an HSZ or HSG type of storage device on this shared bus so you have the option of mirroring the quorum disk and member boot partitions. (See the note about mirroring clusterwide file systems in Section 2.4 for more information.)

**Note**

Although a quorum disk is somewhat analogous to an ASE tie-breaker disk, there are important differences. An ASE tie-breaker disk must participate in an ASE service. A quorum disk should not contain any valuable data.

We also recommend that you have a spare disk on the private bus of the system on which you will install Tru64 UNIX Version 5.1A. If possible, you want to avoid installing the Version 5.1A Tru64 UNIX operating system on the disk that contains the operating system used by the ASE. If you decide to revert to the ASE, it is easier to boot the ASE operating system disk than to reinstall the older operating system and re-create the ASE environment for this system.

**Notes**

TruCluster Server supports SCSI IDs 0-15. If you do not have SCSI IDs available before starting the upgrade, you can add additional storage after you boot the Tru64 UNIX operating

system and map existing device names to new device names. You must have all storage attached and visible to the Tru64 UNIX system before creating a cluster.

Tru64 UNIX Version 5.1A and TruCluster Server Version 5.1A member boot disks do not have to reside at LUN 0.

---

- (Option 3) If you plan to use LSM in the new cluster, see Section 2.4.2. You will need at least one available partition on a shared drive for the `rootdg` disk group (for redundancy, you should have more than one). You can use the `a` partition of the device that you plan to use for the clusterwide root (`/`) file system, which is usually put in the `b` partition for size reasons.

  If you plan to use hardware RAID to mirror the cluster's file systems, the upgrade is a good time to add RAID hardware and take advantage of multipathing. See the *Cluster Hardware Configuration* manual for information on configuring storage in a TruCluster Server Version 5.1A cluster.

### 8.3.3 File-System Requirements

TruCluster Server Version 5.1A clusters use AdvFS file systems. UFS is supported read-only clusterwide. A cluster member can mount a UFS file system read/write. However, that file system is accessible only by that member.

**Note**

To make it easier to migrate file systems when performing an upgrade, TruCluster Server Version 5.1A includes read/write support for UFS file systems. When you mount a UFS file system in a Version 5.1A cluster for read/write access, the `mount` command `-o server_only` argument is used by default. These file systems are treated as partitioned file systems, which means that the file system is accessible only by the member that mounts it. Other cluster members cannot read from, or write to, the file system. There is no remote access; there is no failover. If you want to mount a UFS file system for read-only access by all cluster members, you must explicitly mount it read-only.

You can mount an AdvFS file system as a partitioned file system by explicitly using the `-o server_only` argument when mounting the file system.

File system partitioning is described in the *Cluster Administration* manual and `mount(8)`.

---

If the current ASE services use any UFS file systems for data that you want to access read/write on the new cluster, we recommend that you migrate those file systems to AdvFS before beginning the upgrade. (If it is acceptable that only one member of the new cluster have read/write access to a UFS file system, you can use file system partitioning.) If you decide to migrate the data, how you perform the data migration depends on whether you have enough storage in your current configuration to create AdvFS domains on new storage, or whether you have to do backups and restores to reuse current storage. You may decide to use the TruCluster Server local support for UFS read/write as part of your migration strategy.

**Note**

If your upgrade path is Option 1 (new systems, new storage), you do not have to convert or modify the file systems used by the ASE. However, the file-system data migration strategy you use should deliver AdvFS file systems to the new cluster in order to provide file systems that all members can both read and write.

---

Version 5.0 and later operating systems create AdvFS domains with an disk structure known as domain version number 4 (DVN4). DVN4 provides support for quota values larger than 2 terabytes (TBs) and increases performance for directories containing thousands of files. Domains created prior to Version 5.0 use DVN3; these domains are recognized by later versions but are not automatically upgraded to the new disk structure.

When the upgrade to TruCluster Server is complete, you can convert AdvFS DVN3 domains to DVN4. Whether you convert, and when you convert, are up to you. Tru64 UNIX Version 5.1A recognizes both DVN3 and DVN4 formats. (You can use the `mkfdmn -V3` option to create old-style AdvFS domains on the new cluster.)

If you plan to reuse current storage, converting from UFS to AdvFS on the existing ASE means you can do the file system conversion at your convenience, but the file system format is DVN3. If you do not convert from UFS to AdvFS on the current ASE, you can do the conversion as part of the upgrade, which increases your down time. You may also have a difficult time reverting to the ASE if you encounter problems during the upgrade.

It takes time to run the `vdump`/`vrestore` utilities on file systems when converting formats. If you want to take advantage of the new AdvFS format on the new cluster, you have these UFS conversion choices:

- Convert UFS file systems to AdvFS DVN3 domains before migrating, and then convert again to AdvFS DVN4 after migrating. You perform two conversions, but the data is available as read/write as soon as it is migrated to the cluster. (You can also convert from UFS to AdvFS on the ASE, but continue to use the AdvFS DVN3 format on the new cluster, converting domains to DVN4 when you need to take advantage of the DVN4 features.)
- Convert to AdvFS after migrating. You perform only one conversion, but the data is available read-only for clusterwide use or read/write for local use only until you perform the conversion. (You can use the UFS read/write local-use-only feature to control when you migrate some UFS file systems to AdvFS domains on the new cluster.)

## 8.4 The Upgrade Scripts Used with Option 2 and Option 3

The following list describes the scripts in the `TCRMIGRATE520` subset that are used in the Option 2 and Option 3 upgrade procedures. The scripts and associated utility programs are available from the TruCluster Server Version 5.1A directory on the Tru64 UNIX Associated Products Volume 2 CD-ROM, in the `TCRMIGRATE520` subset. Use the `setld` command to load the subset on each member of the ASE. The utilities and associated libraries are installed in the `/usr/opt/TruCluster/tools/migrate` directory.

The scripts are KornShell (`ksh`) scripts. If you know shell programming and want to know exactly what the scripts do, you can perform a code inspection of the scripts before running them. (As an alternative to running the storage migration scripts, Appendix E provides manual steps for mapping device names and configuring storage.)

`clu_migrate_check`

(Option 3) Performs general hardware, firmware, and file-system type checks.

Before beginning an Option 3 upgrade, run `clu_migrate_check` on each member of the current ASE.

`clu_migrate_save`

(Option 2 and Option 3) Creates a directory and saves information about the current system, ASE configuration, and the shared storage used by ASE services.

After creating a new cluster and while all shared storage is still connected to the ASE, run `clu_migrate_save` on each member of the ASE that has an online ASE service. The `clu_migrate_save` script gathers the information needed to migrate the storage currently used by ASE services to the new cluster. (This includes the migration of any AdvFS domains or LSM volumes associated with this storage.)

The only change the script makes to the current configuration is to write each shared disk's `rz*` special file name to the `label:` field of that disk's label. However, the script saves the original disk label, which you can restore when running `clu_migrate_configure`. Putting the `rz*` name in the `label:` field makes it possible for `clu_migrate_configure` to map each disk device to its new-style `dsk*` device name.

The `clu_migrate_save` script gives you the option of automatically copying data to the new cluster (Option 2) or the Tru64 UNIX Version 5.1A system (Option 3). The script stores information in the `/var/TruCluster_migration` directory, giving each member its own directory on the target system. It uses the value of each member's `/etc/rc.config` `CLUSTER_NET` variable to create the following naming convention:

`/var/TruCluster_migration/`*`CLUSTER_NET`*

The `clu_migrate_configure` script then uses the information in these directories to map physical storage device names, and to configure storage on the new cluster or Tru64 UNIX system.

`clu_migrate_configure`

(Option 2 and Option 3) Configures storage on the new TruCluster Server Version 5.1A cluster (Option 2) or on the Tru64 UNIX Version 5.1A system (Option 3).

After the ASE systems are turned off and the shared storage is connected to the new cluster, run `clu_migrate_configure` to automatically configure the storage previously managed by the ASE. The `clu_migrate_configure` script merges the information gathered from the `clu_migrate_save` output on the ASE members. It then configures storage — mapping the old-style device names written to each disk's `label:` field to the new-style `dsk` device names, importing LSM volumes, re-creating AdvFS domains, testing mount points, and adding entries to `/etc/fstab` and `/etc/exports`. When the storage configuration is complete, the script restores any saved `label:` field values that were overwritten by `clu_migrate_save`.

Although the `clu_migrate_configure` script will provide device-name mapping for all shared disk devices, it does not configure storage that was not managed by the ASE. It does not migrate entries in an ASE member's `/etc/fstab` file for shared storage if that storage was not managed by the ASE.

The script does not convert existing AdvFS file systems to the new AdvFS format. To use the new AdvFS format, convert AdvFS file systems after completing the upgrade.

`clu_migrate_recover`

(Option 2 and Option 3) Frees device reservations and restores the LSM configuration on an ASE member. During a successful upgrade, you do not run this script.

The `clu_migrate_recover` script is run on the ASE systems to revert back to an ASE as part of the recovery procedure in Section 8.7.2. Run this script only if an upgrade fails to complete successfully.

**Note**

If you converted AdvFS file systems to the new AdvFS format during the upgrade, the `clu_migrate_recover` script does not them convert back to the old format. You must do this manually.

## 8.5 Option 1: Create a Separate Cluster — New Systems and New Storage

A separate cluster lets you test applications and system configurations without any interaction with your current production environment.

Whether you can set up a separate cluster depends on your current hardware configuration, and your ability to obtain new hardware. If you can afford new hardware and have the space to configure a separate cluster, this is the preferred approach.

The following procedure outlines the general steps for this approach:

1. Use the information in the *Cluster Hardware Configuration* manual to design the hardware configuration for the new cluster. Some things to consider are:
   - Creating a no-single-point-of-failure (NSPOF) cluster. (The *Cluster Hardware Configuration* manual provides a section listing the basic hardware requirements for an NSPOF cluster.)
   - Using fast RAID controllers and multipathing to perform hardware mirroring of the quorum disk or member boot partitions. (TruCluster Server does not support LSM mirroring of these file systems.)
   - Providing room for growth. For example, order systems that, when configured, still have open peripheral component interconnect (PCI) slots, creating an expandable network and storage topology.
2. When the hardware is on site, physically connected, and with all firmware and console configuration completed, use the information in this manual to create a TruCluster Server cluster.

**Note**

When selecting IP addresses for the new cluster, do not use those currently in use by the ASE. (Remember that when the new cluster is in operation, clients must direct all NFS mount requests to the default cluster alias or to an alias whose name is in `/etc/exports.aliases`.)

3. Become familiar with the capabilities of the new cluster.

   Using the information in the *Cluster Administration* manual, configure and administer the cluster. For example, create and test data backup and restore procedures, monitor hardware, and tune and balance the cluster.

   Using the information in the *Cluster Highly Available Applications* manual, create and test highly available applications. For each critical application you will deploy, determine which approach best fits that application. After you configure applications and failover policies, force failovers and verify that the results are what you expected. Modify and tune until you are satisfied.

4. Data migration: Because the new cluster is completely separate from the current ASE, the major task is migrating application data from the ASE to the new cluster.

   **Note**

   We recommend that, before beginning data migration, you back up both the ASE and the new cluster.

   You are responsible for deciding how and when to migrate the data used by the ASE to the new cluster. This section does not supply a recommended procedure or set of tools. The following list discusses some of the important points to consider when defining a strategy for your data migration:

   - Whether you use standard `vdump`/`vrestore` utilities or application-specific data migration tools depends on your current environment, the application, and the amount of data. In addition, if some or all of an application's data is stored in UFS format, restore it to the AdvFS format on the new cluster. (See Section 8.3.3 for information on converting from UFS to AdvFS.)

     Database vendors often provide applications that can update remote copies of databases. If you have a database application with remote update capabilities, you can use that feature to migrate data from the ASE to the cluster. When the update is complete (and tested

on the new cluster), move the live IP addresses for the database services to the new cluster.

- If the current ASE is serving multiple applications, you must decide whether to migrate all application data at the same time, or to migrate one application at a time, serving clients from both the ASE and the new cluster. If you decide to migrate applications one at a time, determine the order in which you will migrate them.
- If both the ASE and the cluster will be serving clients during part of the migration, devise a procedure to perform backups and recovery during the transition period.

## 8.6 Option 2: Create a Separate Cluster — New Systems and Existing Storage

This approach is similar to Option 1, with the exception that at some point you plan to physically move the existing ASE storage devices to the new cluster. This section provides the following information:

- Upgrade procedure (Section 8.6.1)
- Procedure for reverting a partial upgrade (Section 8.6.2)

### 8.6.1 Option 2: Upgrade Procedure

The following procedure outlines the general steps for Option 2:

1. Use the information in the *Cluster Hardware Configuration* manual to design the hardware configuration for the new cluster. Some things to consider are:
   - Creating a no-single-point-of-failure (NSPOF) cluster. (The *Cluster Hardware Configuration* manual provides a section listing the basic hardware requirements for an NSPOF cluster.) Note that because you are using the storage hardware from the ASE, any NSPOF limitations in that configuration will affect the new cluster. For example, some disks might not support multipathing.
   - The new cluster must have the shared storage required to create a cluster and add members. In addition, provide some additional shared storage for application testing before the ASE storage is physically moved to the new cluster.
   - Because the new cluster will use the storage hardware from the current ASE for application data, make sure that the new cluster systems have storage adapters that are compatible with the ASE's storage topology. If you plan to move the adapters from the ASE systems to the new systems, make sure that the adapters are supported on the new systems.

- Using fast RAID controllers and multipathing to perform hardware mirroring of the quorum disk or member boot partitions. (TruCluster Server does not support LSM mirroring of these file systems.)
- Providing room for growth. For example, order systems that, when configured, still have open slots, creating an expandable network and storage topology.

2. When the new hardware is on site, and physically connected, with all firmware and console configuration complete, use the information in this manual to create a TruCluster Server Version 5.1A cluster.

**Note**

When selecting IP addresses for the new cluster, do not use those currently in use by the ASE.

3. Become familiar with the capabilities of the TruCluster Server cluster.

   Using the information in the *Cluster Administration* manual, configure and administer the cluster. For example, create and test data backup and restore procedures, monitor hardware, and tune and balance the cluster.

   Using the information in the *Cluster Highly Available Applications* manual, create and test highly available applications. For example, use CAA and cluster alias until you understand the differences between the two subsystems. After you configure applications and failover policies, force failovers and verify that the results are what you expected. Modify and tune until you are satisfied.

4. Up to this point, the ASE and TruCluster Server cluster are entirely separate: what you do on one does not affect the other. Starting with the next step, you are preparing to move storage from the ASE to the new cluster. Before continuing, do the following:

   a. Read the description of the utility scripts in Section 8.4. Read the Option 3 procedure in Section 8.7.1. You will use the scripts and parts of the procedure when moving the storage from the ASE to the cluster.

   b. Edit `/.rhosts` on the new cluster to allow `root` access from each member of the ASE. (This lets `clu_migrate_save` automatically copy information from the ASE members to the `/var/TruCluster_migration` directory on the cluster.)

   c. Back up the new cluster. If anything goes wrong, you have a good starting point for recovery.

5. On each member of the ASE, load the TCRMIGRATE520 subset, which is in the TruCluster Server Version 5.1A directory on the Tru64 UNIX Associated Products Volume 2 CD-ROM. The following example assumes that the CD-ROM is mounted on /mnt:

   ```
   # setld -l /mnt/TruCluster/kit TCRMIGRATE520
   ```

   The migration scripts, utility programs, and libraries are installed in the /usr/opt/TruCluster/tools/migrate directory.

6. Label all storage cables and adapters (in case you ever need to reconnect the storage to the ASE).

7. Make sure that all LSM-based services are online in the ASE.

8. On each member of the ASE that is running an ASE service, run clu_migrate_save:

   ```
   # /usr/opt/TruCluster/tools/migrate/clu_migrate_save
   ```

9. On the TruCluster Server Version 5.1A cluster:

   a. Verify that the files created by clu_migrate_save have been copied to the /var/TruCluster_migration directory.

   b. Shut down and halt the cluster:

      ```
      # shutdown -c now
      ```

   c. Turn off each member of the cluster.

   **Note**

   Whether or not to turn off storage is your decision.

10. On all members of the ASE, do the following:

    a. Using your normal backup procedures, back up all data. Starting with the next step, you are moving storage devices from the ASE to the cluster. If for any reason you need to revert to the ASE, this is your last chance to do a full backup of the current configuration.

    b. Take all ASE services off line. (If you did not need to take them off line when backing up the data, take them off line now.)

    c. Shut down and halt each system.

    d. Turn each system off.

11. On the TruCluster Server cluster, do the following:

    a. Connect the shared storage from the ASE to the cluster systems.

**Note**

TruCluster Server Version 5.1A does not insist that the storage be connected symmetrically to the cluster members in order to be available. However, because the ASE storage contains application data that should be highly available, we recommend that you connect the storage so that it is directly accessible from all members.

b. If you turned off the shared storage, turn it on.

c. Turn on the cluster members.

d. At each console, run the `show dev` command and, to the extent possible, verify that the disk devices used by the ASE are visible. Find and fix any hardware-related problems before booting the cluster.

e. Boot the cluster. If the new storage is not symmetrically configured, make sure to boot all members that have direct connections to the new storage.

**Note**

During the boot phase, the cluster members will discover the new devices and create special device files.

12. On one member of the cluster, run `clu_migrate_configure`. This command verifies that storage devices are visible to the operating system, maps old-style device names to new-style device names, and configures storage.

13. On the cluster, do the following:

    a. Test applications with full storage. Can the applications see the data? Can applications use the data? The earlier application testing established that the application ran without problems. The purpose of this full-storage testing is to satisfy yourself that applications can see and manipulate the data they will use when serving clients.

**Note**

If you encounter problems that you cannot resolve, and you decide to revert back to the ASE, follow the procedure in Section 8.6.2.

b. Start serving clients. Note that NFS clients must direct NFS mount requests to the default cluster alias or to an alias whose name is in /etc/exports.aliases.

14. (Optional) Remove the migration directories and delete the migration subset. On one member of the cluster:

    a. Remove the migration directories:

    ```
    # rm -rf /var/TruCluster_migration
    ```

    b. Delete the migration tools:

    ```
    # setld -d TCRMIGRATE520
    ```

### 8.6.2 Option 2: Reverting a Partial Upgrade

If you encounter problems that you cannot resolve when migrating storage or completing the upgrade, follow these steps to revert to the ASE configuration:

1. Halt the cluster.
2. Turn off each member of the cluster.

**Note**

Whether or not to turn off storage is your decision.

3. Reconnect storage to the ASE. Make sure to connect it exactly the same way it was connected.
4. If you turned the shared storage off, turn it on.
5. Turn on all ASE systems.
6. Boot the ASE members.
7. On each member of the ASE on which you ran clu_migrate_save, run the clu_migrate_recover command:

    ```
    # /usr/opt/TruCluster/tools/migrate/clu_migrate_recover
    ```

8. When all members are up and recovered, on one member run the asemgr command to set all services on line.
9. (Optional) Remove the migration directories and delete the migration subset. On each member of the ASE:

    a. Remove the migration directories:

    ```
    # rm -rf /var/TruCluster_migration
    ```

    b. Delete the migration tools:

    ```
    # setld -d TCRMIGRATE520
    ```

## 8.7 Option 3: Use Existing Systems and Storage

This section explains how to upgrade an existing Version 1.5 or Version 1.6 TruCluster Production Server Software cluster or TruCluster Available Server Software (ASE) to TruCluster Server Version 5.1A using the existing hardware. The section provides the following information:

- Upgrade procedure (Section 8.7.1)
- Procedure for reverting a partial upgrade (Section 8.7.2)

### 8.7.1 Option 3: Upgrade Procedure

The following procedure uses the scripts described in Section 8.4 to automate several migration tasks. In order to use the scripts, the current ASE configuration must meet the hardware and storage topology requirements in Section 8.3.2.

These are the paths you can take through the following procedure:

- Automatic storage configuration: Use the scripts to examine the current system configuration, save information, and automatically configure storage on new cluster.
- Manual storage configuration: Use the scripts to examine the current system configuration and save information, but on the new cluster run `clu_migrate_configure -x`. The `-x` option displays all the configuration information, but, instead of configuring storage, just lists the configuration commands that `clu_migrate_configure` normally uses. You can then examine the command sequence, with the option of creating and manually executing your own series of commands, or rerunning `clu_migrate_configure` without the `-x` option.

For either path, most steps are identical. Steps that are only for those who plan to manually configure storage on the new cluster are marked **MANUAL STORAGE CONFIGURATION ONLY**. Steps that are only for those who are using `clu_migrate_configure` to configure storage on the new cluster are marked **AUTOMATIC STORAGE CONFIGURATION ONLY**.

We recommend that you read the entire procedure before you decide which path best fits your site's requirements.

1. On each member of the ASE, load the `TCRMIGRATE520` subset, which is in the TruCluster Server Version 5.1A directory on the Tru64 UNIX Associated Products Volume 2 CD-ROM. The following example assumes that the CD-ROM is mounted on `/mnt`:

   ```
   # setld -l /mnt/TruCluster/kit TCRMIGRATE520
   ```

   The migration scripts, utility programs, and libraries are installed in the `/usr/opt/TruCluster/tools/migrate` directory.

2. On each member of the current ASE, run `clu_migrate_check`:

   ```
   # /usr/opt/TruCluster/tools/migrate/clu_migrate_check
   ```

   Use the output in conjunction with the requirements listed in Section 8.3.2 to determine whether each system's hardware and firmware are configured properly for an upgrade.

3. Label all storage cables and adapters.

4. Decide which ASE member system will become the first member of the new cluster.

5. Manually relocate ASE services to the remaining member or members of the ASE.

6. Delete the system that will become the first member of the new cluster from the ASE.

7. On the system that will become the first member of the new cluster:

   a. Shut down and halt the system.

   b. At the system console:

      i. Run the `show dev` command and capture the output for all shared storage.

      ii. Make a record of all console variables and values.

      iii. Set the following console variables:

      ```
      >>> set auto_action halt
      >>> set bootdef_dev ""
      >>> set boot_reset on
      ```

      For systems that support the `bus_probe_algorithm` variable:

      ```
      >>> set bus_probe_algorithm new
      ```

      See Section 2.6 for more information on console variables. Note that setting `bus_probe_algorithm` variable on systems that do not use the variable is benign. The variable is cleared at the next `init` or power cycle.

   c. Turn the system off.

   d. After making sure that all cables and connections are labeled, disconnect all shared storage cables (for example, SCSI or Fibre Channel) from the system and terminate adapters as needed; if the existing cables are terminated with Y cables or trilink adapters, you should not need to add any terminators. See the *Cluster Hardware Configuration* manual for information on terminating SCSI adapters.

e. If the system has one or more Memory Channel cluster interconnect adapters, disconnect the cable or cables.

f. If you are adding Memory Channel to the system for use as the cluster interconnect, follow the instructions in the *Cluster Hardware Configuration* manual. Do not connect the adapters to cables at this time. (TruCluster Server Version 5.1A supports LAN hardware for the cluster interconnect. See the *Cluster LAN Interconnect* manual for information on configuring a cluster with a LAN interconnect.)

g. Turn the system on.

h. Use the console `show config` command to determine whether the console and adapter firmware revisions are compatible with Tru64 UNIX Version 5.1A. If they are not, update the firmware as needed.

i. Following the instructions in Chapter 3, perform a full installation of Tru64 UNIX Version 5.1A.

**Note**

We strongly recommend that you do not overwrite the disk or disks containing the operating system used by the ASE. If you encounter problems later on, you can quickly return this system to the ASE as long as these disks are intact.

If you must overwrite the disk or disks, back up the current operating system before installing Tru64 UNIX Version 5.1A.

8. On the Tru64 UNIX Version 5.1A system, do the following:

a. Fully configure the Tru64 UNIX operating system. Follow the instructions in Chapter 3.

b. Install applications.

c. Edit `/.rhosts` on the Tru64 UNIX Version 5.1A system to allow `root` access from the remaining members of the ASE. (This lets `clu_migrate_save` automatically copy information from the remaining ASE members.)

d. Install the TruCluster Server Version 5.1A license and subsets.

**Note**

We recommend that you back up the Version 5.1A system before continuing. If something goes wrong later

in the procedure, you can restore to this point faster than you can install and configure Tru64 UNIX, install applications, and load the TruCluster Server subsets.

---

9. On the ASE, make sure that all LSM-based services are on line.
10. On each member of the ASE that is running an ASE service, run `clu_migrate_save`:

    ```
    # /usr/opt/TruCluster/tools/migrate/clu_migrate_save
    ```

11. On the Tru64 UNIX system:

    a. Verify that the files created by `clu_migrate_save` have been copied to the `/var/TruCluster_migration` directory.

    b. Halt the system and turn it off.

12. On all members of the ASE, do the following:

    a. Using your normal backup procedures, back up all data. Starting with the next step, you are moving storage devices from the ASE to the Tru64 UNIX Version 5.1A system. If for any reason you need to revert to the ASE, this is your last chance to do a full backup of the current configuration.

    b. Take all ASE services off line. (If you did not need to take them off line when backing up the data, take them off line now.)

    c. Shut down and halt each system.

    d. At each system's console, set the following console variables:

    ```
    >>> set auto_action halt
    >>> set bootdef_dev ""
    >>> set boot_osflags A
    >>> set boot_reset on
    ```

    For systems that support the `bus_probe_algorithm` variable:

    ```
    >>> set bus_probe_algorithm new
    ```

    See Section 2.6 for more information on console variables. Note that setting `bus_probe_algorithm` variable on systems that do not use the variable is benign. The variable is cleared at the next `init` or power cycle.

    e. Turn each system off.

**Note**

Do not turn these systems on until instructed to do so.

---

13. On the Tru64 UNIX Version 5.1A system, do the following:

    a. Connect all shared storage to the system. Make sure to cable the storage just as it was cabled when the system was part of the ASE.

    b. Connect the cluster interconnect cable or cables to the system. If the cluster configuration uses a Memory Channel hub, or an Ethernet hub or a switch, connect the cable or cables to the hub or a switch. If you are not using a hub or switch, just connect the cables.

    c. Turn the system on.

    d. At the console, run the `show dev` command and, to the extent possible, verify that all shared devices are visible. (Compare the displayed device list to the information you saved before disconnecting storage.) Find and fix any hardware-related problems before booting the system.

       Also compare the remaining saved console information to the current console.

    e. Boot the system to multi-user mode, and log in.

14. **AUTOMATIC STORAGE CONFIGURATION ONLY:** On the Tru64 UNIX Version 5.1A system, run `clu_migrate_configure`. This command verifies that storage devices are visible to the operating system, maps old-style device names to new-style device names, and sets up storage.

    ```
    # /usr/opt/TruCluster/tools/migrate/clu_migrate_configure
    ```

15. **MANUAL STORAGE CONFIGURATION ONLY:** On the Tru64 UNIX Version 5.1A system, run `clu_migrate_configure -x`:

    ```
    # /usr/opt/TruCluster/tools/migrate/clu_migrate_configure -x
    ```

    The `-x` option displays all the configuration information, but, instead of configuring storage, just lists the configuration commands that `clu_migrate_configure` normally uses. You can then examine the command sequence, with the option of creating and manually executing your own series of commands, or rerunning `clu_migrate_configure` without the `-x` option.

    If, after examining the displayed series of commands, you decide to automatically configure storage, run `clu_migrate_configure`:

    ```
    # /usr/opt/TruCluster/tools/migrate/clu_migrate_configure
    ```

    If, after examining the displayed series of commands, you decide to manually configure storage, do so now. (Section E.2 provides a procedure for manually configuring storage.) When you are finished,

use the following commands to examine the storage configuration and (optional) recover the original disk labels:

a. Run the following LSM commands to display the LSM configuration:

```
# voldisk list
# volprint -thA
```

b. For each AdvFS domain, run the `showfdmn` *`domain`* command.

c. For each AdvFS domain, run the `showfsets` *`domain`* command and verify that the filesets are correct for the domain.

d. (Optional) The `clu_migrate_save` created a `/var/TruCluster_migration/`*`CLUSTER_NET`*`/Packids` file for each ASE member on which it was run. The `Packids` file contains the original disk labels for shared devices known to that member. If the original disk labels contained values in the packid (`label:`) field, you can restore the original labels. To restore the original disk labels, use the `restore_packids` script. Change directory to `/usr/opt/TruCluster/tools/migrate/utils`. Run the following command:

```
# ./restore_packids -f \
  /var/TruCluster_migration/CLUSTER_NET/Packids
```

For each member, replace *`CLUSTER_NET`* with its value. For example, for an ASE member whose *`CLUSTER_NET`* value was `mcclu14`, the command is:

```
# ./restore_packids -f \
  /var/TruCluster_migration/mcclu14/Packids
```

16. On the Tru64 UNIX Version 5.1A system, do the following:

a. Follow the procedures in Chapter 4 and run the `clu_create` command to create a single-member cluster.

b. Halt the system and boot it as a single-member cluster.

17. On the single-member TruCluster Server Version 5.1A cluster, do the following:

a. Set up CAA profiles and scripts for applications. See the *Cluster Highly Available Applications* manual, the *Cluster Administration* manual, and `caa_profile`(8) for information on creating application profiles and using CAA.

b. If you plan to use cluster aliases other than the default cluster alias, specify and join those cluster aliases. See the *Cluster Administration* manual and `cluamgr`(8) for information on configuring cluster aliases.

c. Test applications with full storage. Can the applications see the data? Can applications use the data? The earlier application testing established that the application ran without problems. The purpose of this full-storage testing is to satisfy yourself that applications can see and manipulate the data they will use when serving clients.

**Note**

If you encounter problems that you cannot resolve, and you decide to revert back to the ASE, follow the procedure in Section 8.7.2.

d. Start serving clients. Note that NFS clients must direct all NFS mount requests to the default cluster alias or to an alias whose name is in `/etc/exports.aliases`.

18. One at a time, add the remaining ASE members to the cluster. Follow these steps for each system:

    a. Make sure the system is turned off.

    b. If adding or replacing cluster interconnect adapters (Memory Channel or Ethernet), install the adapter or adapters.

    c. Connect the system to the shared storage.

    d. Connect cluster interconnect cables.

    e. Turn the system on.

    f. Use the console `show config` command to determine whether the console and adapter firmware revisions are compatible with Tru64 UNIX Version 5.1A. If they are not, update firmware as needed.

    g. Following the procedure in Chapter 5, run the `clu_add_member` command on a current cluster member to create a boot disk for the new member. Boot the new member into the cluster.

19. (Optional) Remove the migration directories and delete the migration subset. On one member of the cluster:

    a. Remove the migration directories:

    ```
    # rm -rf /var/TruCluster_migration
    ```

    b. Delete the migration tools:

    ```
    # setld -d TCRMIGRATE520
    ```

### 8.7.2 Option 3: Reverting a Partial Upgrade

If you encounter problems that you cannot resolve when migrating storage or completing the upgrade, follow these steps to revert to the ASE configuration:

1. Halt all systems.
2. Turn all systems off.

**Note**

Whether or not to turn off storage is your decision.

3. If you installed Memory Channel adapters, remove them.
4. If storage is not connected to all systems, reconnect storage. Make sure to connect it exactly the same way it was connected for the ASE.
5. If you turned the shared storage off, turn it on.
6. Turn all systems on.
7. Restore console variables to their previous values.
8. If you used SCSI wide addressing (8-15) during the upgrade, restore the previously saved settings.
9. On each member of the ASE, boot the previous version of the operating system to multi-user mode.
10. On each member of the ASE on which you ran `clu_migrate_save`, run the `clu_migrate_recover` command:

    ```
    # /usr/opt/TruCluster/tools/migrate/clu_migrate_recover
    ```

11. When all members are up and recovered, on one member run the `asemgr` command to set all services online.
12. (Optional) Remove the migration directories and delete the migration subset. On each member of the ASE:

    a. Remove the migration directories:

    ```
    # rm -rf /var/TruCluster_migration
    ```

    b. Delete the migration tools:

    ```
    # setld -d TCRMIGRATE520
    ```

## 8.8 Upgrade Case Study

This section describes an upgrade of a four-member TruCluster Production Server Version 1.6 cluster to TruCluster Server Version 5.0A. The customer had several Version 1.6 production-level clusters and wanted to perform

a test migration on similar hardware before deciding how to upgrade its existing clusters.

**Note**

Although the target version for the case study was Version 5.0A, the basic steps are the same regardless of which 5.*x* version of TruCluster Server you upgrade to.

In addition to the case study described here, the following URL contains a link to the description of an upgrade of an in-house production-level cluster from TruCluster Production Server Version 1.6 to TruCluster Server Version 5.0A plus a patch kit:

```
http://www.tru64unix.compaq.com/docs/highavail/index.htm
```

The pre-upgrade cluster consisted of the following hardware and software:

- 4 AlphaServer GS140 rackmount systems, each with 8 CPUs, 8 GB RAM, and 14 KGPSA-BC adapters.
- 14 Fibre Channel Storage Area Network switches (16-port).
- 14 StorageWorks™ ESA 12000 Storage Array Fibre Channel cabinets. (Each cabinet contained dual-redundant HSG80 array controllers and forty-eight 36-GB disks.)
- 2 StorageWorks ESL9326D Enterprise tape libraries.
- 2 Memory Channel II hubs.
- Tru64 UNIX Version 4.0F with patches.
- TruCluster Production Server Version 1.6 (no ASE services, no LSM). (The cluster was configured to be shipped with Tru64 UNIX Version 5.0A and TruCluster Server Version 5.0A. It was deliberately loaded with Version 4.0F and Production Server Version 1.6 to test an upgrade.)
- Single instance of Oracle 8.1.6 with 112 distributed raw disk (DRD) devices. (The customer planned to run the same binaries on the Version 5.0A cluster to test the success of the upgrade.)

The post-upgrade cluster will run TruCluster Server Version 5.0A on Tru64 UNIX Version 5.0A.

After reviewing the three general upgrade paths, the customer decided to use a modified version of Option 3. The following decisions were made:

- Do not disconnect storage cables or Memory Channel cables.

  Because there were so many cable connections in a cluster of this size, the customer decided that not allowing any disconnects would save time and reduce risk.

- Do not turn off storage, Memory Channel hubs, or systems.

  Because a real upgrade would be controlled remotely from a computer center, the customer decided that taking the systems to the console prompt as needed could replace the recommended power downs. They trusted that the administrator would not boot a system at the wrong time.

- Halt all systems at the same time.

  Because storage and Memory Channel hubs were connected and turned on, they would not keep the Production Server cluster running while installing Tru64 UNIX Version 5.0A.

- Before upgrading, copy all output from `clu_migrate_save` to a file system on a local disk on the system where they would install Tru64 UNIX Version 5.0A. After the Tru64 UNIX Version 5.0A system was installed, they would mount that file system and copy the files to `/var/TruCluster_migration` before running `clu_migrate_configure`.

  In both Option 2 and Option 3, the output from `clu_migrate_save` is copied across the network to a new cluster (Option 2) or to a member of the current cluster that has been physically disconnected from storage and Memory Channel (Option 3). Because the proposed procedure shuts down the entire Version 1.6 Production Server cluster before installing Tru64 UNIX Version 5.0A, the customer needed a method for getting the files to the Version 5.0A system.

The pre-upgrade cluster was somewhat artificial in nature. It was not using LSM, nor were any ASE services defined. A single-instance of Oracle 8.1.6 would be used to test the "success" of the upgrade. If the application ran on the new cluster and could access all its storage, the upgrade would be considered a success.

The following is a summary of the actual upgrade procedure:

1. Loaded the TCRMIGRATE505 subset on all four members of the TruCluster Software Version 1.6 cluster.
2. Ran `clu_migrate_check` on all members.
3. Chose one member as the lead member (the first system to upgrade). Created a `/var/TruCluster_migration` directory.
4. Ran `clu_migrate_save` on all members, using the lead member's `/var/TruCluster_migration` directory as the destination for the `rcp` of the data files created by `clu_migrate_save`.

   Used `disklabel -r` to examine a sampling of disk labels. Verified that the `@rz`*xxx* string representing a disk's current device special file name was embedded in that disk's label.

5. Shut down all members of the Version 1.6 cluster to console mode.

**Note**

The Fibre Channel switch fabric was already configured to take advantage of Version 5.0A support for HSG80 multiple-bus failover. If it were not, this would have been the time to recable the SAN switches. For this upgrade, while all systems were at the console prompt, each HSG80 controller was changed from transparent failover to multiple-bus failover, and then rebooted.

6. At an HSG80 console, used five disks to create a storageset (RAID level 5) with the following partitions (sizes in percentages; total available space after deducting for parity is approximately 145 GB):
   - 6 percent for the Tru64 UNIX Version 5.0A disk (approximately 8.7 GB)
   - 6 percent for the TruCluster Server Version 5.0A disk: root (/), /usr/, and /var (approximately 8.7 GB)
   - 15 percent for each member boot disk (approximately 21.8 GB each)

     Because each GS140 contains 8 GB of memory, each system would need lots of swap space.
   - 1 percent for the quorum disk (approximately 1.45 GB)
   - The remaining partition (27 percent) was reserved for future use (approximately 39.3 GB)

   The following is a summary of the commands used at the HSG80 console:

**Note**

Only some of the console and HSG80 commands that were used to configure storage and boot disks are shown in these examples. If you plan to use boot disks behind an HSG80 controller, use the full procedures described in the TruCluster Server *Cluster Hardware Configuration* manual.

```
HSG14 BOT> show unit

    LUN                                     Uses             Used by
------------------------------------------------------------------------------
  D0                                        R1
  D1                                        R2
  D2                                        R3
  D3                                        R4
  D100                                      R5
  D101                                      R6
```

```
  D102                                    R7
  D103                                    R8

HSG14 BOT> locate d103       ! verify the disk to be deleted
HSG14 BOT> locate cancel
HSG14 BOT> delete unit d103  ! delete the existing unit
HSG14 BOT> show r8           ! check how much space is available

Name          Storageset                    Uses             Used by
------------------------------------------------------------------------------

R8            raidset                       DISK21100
                                            DISK31100
                                            DISK41100
                                            DISK51100
                                            DISK61100
        Switches:
          POLICY (for replacement) = BEST_PERFORMANCE
          RECONSTRUCT (priority) = NORMAL
          CHUNKSIZE = 256 blocks
        State:
          UNKNOWN -- State only available when configured as a unit
        Size:             284389020 blocks


HSG14 BOT> create_partition r8 size=6
HSG14 BOT> create_partition r8 size=6
HSG14 BOT> create_partition r8 size=15
HSG14 BOT> create_partition r8 size=15
HSG14 BOT> create_partition r8 size=15
HSG14 BOT> create_partition r8 size=15
HSG14 BOT> create_partition r8 size=1
HSG14 BOT> create_partition r8 size=largest

Name          Storageset                    Uses             Used by
------------------------------------------------------------------------------
R8            raidset                       DISK21100
                                            DISK31100
                                            DISK41100
                                            DISK51100
                                            DISK61100
        Switches:
          POLICY (for replacement) = BEST_PERFORMANCE
          RECONSTRUCT (priority) = NORMAL
          CHUNKSIZE = 256 blocks
        State:
          UNKNOWN -- State only available when configured as a unit
        Size:             284389020 blocks
        Partitions:
          Partition number        Size               Starting Block     Used by
          ------------------------------------------------------------------------
            1                 17062907 (   8736.20 MB)            0
            2                 17062907 (   8736.20 MB)     17062912
            3                 42657787 (  21840.78 MB)     34125824
            4                 42657787 (  21840.78 MB)     76783616
            5                 42657787 (  21840.78 MB)    119441408
            6                 42657787 (  21840.78 MB)    162099200
            7                  2843643 (   1455.94 MB)    204756992
            8                 76788375 (  39315.64 MB)    207600640

HSG14 BOT> add unit d4 r8 part=1    ! Tru64 UNIX V5.0A disk
HSG14 BOT> add unit d5 r8 part=2    ! TruCluster V5.0A disk
HSG14 BOT> add unit d6 r8 part=3    ! member 1 boot disk
HSG14 BOT> add unit d7 r8 part=4    ! member 2 boot disk
```

```
HSG14 BOT> add unit d8 r8 part=5    ! member 3 boot disk
HSG14 BOT> add unit d9 r8 part=6    ! member 4 boot disk
HSG14 BOT> add unit d10 r8 part=7   ! quorum disk
HSG14 BOT> add unit d11 r8 part=8   ! remaining space

HSG14 BOT> show r8

Name          Storageset                    Uses             Used by
------------------------------------------------------------------------------

R8            raidset                       DISK21100        D10
                                            DISK31100        D11
                                            DISK41100        D4
                                            DISK51100        D5
                                            DISK61100        D6
                                                             D7
                                                             D8
                                                             D9
        Switches:
          POLICY (for replacement) = BEST_PERFORMANCE
          RECONSTRUCT (priority) = NORMAL
          CHUNKSIZE = 256 blocks
        State:
          NORMAL
          DISK21100 (member  0) is NORMAL
          DISK31100 (member  1) is NORMAL
          DISK41100 (member  2) is NORMAL
          DISK51100 (member  3) is NORMAL
          DISK61100 (member  4) is NORMAL
        Size:             284389020 blocks
        Partitions:
          Partition number        Size               Starting Block     Used by
          ------------------------------------------------------------------------
            1                 17062907 (   8736.20 MB)             0      D4
            2                 17062907 (   8736.20 MB)      17062912      D5
            3                 42657787 (  21840.78 MB)      34125824      D6
            4                 42657787 (  21840.78 MB)      76783616      D7
            5                 42657787 (  21840.78 MB)     119441408      D8
            6                 42657787 (  21840.78 MB)     162099200      D9
            7                  2843643 (   1455.94 MB)     204756992      D10
            8                 76788375 (  39315.64 MB)     207600640      D11

HSG14 BOT> show unit

    LUN                                       Uses             Used by
------------------------------------------------------------------------------

  D0                                          R1
  D1                                          R2
  D2                                          R3
  D3                                          R4
  D4                                          R8           (partition)
  D5                                          R8           (partition)
  D6                                          R8           (partition)
  D7                                          R8           (partition)
  D8                                          R8           (partition)
  D9                                          R8           (partition)
  D10                                         R8           (partition)
  D11                                         R8           (partition)
  D100                                        R5
  D101                                        R6
  D102                                        R7

HSG14 BOT> set d4 id=100   ! create user-defined identifiers (UDIDs)
```

```
HSG14 BOT> set d5 id=101
HSG14 BOT> set d6 id=1
HSG14 BOT> set d7 id=2
HSG14 BOT> set d8 id=3
HSG14 BOT> set d9 id=4
```

7. At the lead member's (member 1) console, used the `wwidmgr` command to map console device names to the user-defined IDs (UDIDs) created on the HSG80 for the Tru64 UNIX Version 5.0A disk and this member's boot disk to console device names, and then set `bootdef_dev`:

```
P00>>> set mode diag
Console is in diagnostic mode
P00>>> wwidmgr -quickset -udid 100    # Tru64 UNIX Version 5.0A disk
P00>>> wwidmgr -quickset -udid 1      # member 1 boot disk
     .
     .
     .
Disk assignment and reachability after next initialization:

6000-1fe1-0005-9dc0-0009-0010-4628-00c6
                          via adapter:         via fc nport:        connected:
dgm1.1001.0.7.7          kgpsam0.0.0.7.7      5000-1fe1-0005-9dc3      Yes
dgm1.1002.0.7.7          kgpsam0.0.0.7.7      5000-1fe1-0005-9dc1      No
dgn1.1003.0.10.7         kgpsan0.0.0.10.7     5000-1fe1-0005-9dc2      No
dgn1.1004.0.10.7         kgpsan0.0.0.10.7     5000-1fe1-0005-9dc4      Yes
     .
     .
     .
P00>>> init
     .
     .
     .
P00>>> show device
     .
     .
     .
kgpsam0.0.0.7.7        PGM0        WWN 1000-0000-c922-09f9
dgm100.1001.0.7.7      $1$DGA100   HSG80  V85F
dgm1.1001.0.7.7        $1$DGA1     HSG80  V85F
dgm1.1002.0.7.7        $1$DGA1     HSG80  V85F
dgn1.1003.0.10.7       $1$DGA1     HSG80  V85F
dgn1.1004.0.10.7       $1$DGA1     HSG80  V85F
     .
     .
     .
P00>>> set bootdef_dev dgm1.1001.0.7.7
P00>>> init
     .
     .
     .
```

8. For each remaining member, used the `wwidmgr` command at its console to map a UDID for that member's boot disk to a console device name, and then set `bootdef_dev`:

```
(member 2)
P00>>> set mode diag
Console is in diagnostic mode
P00>>> wwidmgr -s   quickset udid 2
P00>>> init
     .
     .
     .
P00>>> set bootdef_dev dgm2.1001.0.7.7
P00>>> init
     .
     .
     .
(member 3)
P00>>> set mode diag
Console is in diagnostic mode
```

```
P00>>> wwidmgr -quickset -udid 3
P00>>> init
      .
      .
      .
P00>>> set bootdef_dev dgm3.1001.0.7.7
P00>>> init
      .
      .
      .
(member 4)
P00>>> set mode diag
Console is in diagnostic mode
P00>>> wwidmgr -quickset -udid 4
P00>>> init
      .
      .
      .
P00>>> set bootdef_dev dgm4.1001.0.7.7
P00>>> init
      .
      .
      .
```

**Note**

During this initial configuration of the disks that were needed to create a cluster, the customer assigned UDIDs only to bootable devices. In retrospect, it would have made sense to assign UDIDs to the other disks as well. Associating a UDID with a disk makes it easier to track a device through utilities such as `hwmgr`. Because 1000 UDIDs were available for use, there was no need to ration them.

The customer also set only one path to each boot device at the console using the `bootdef_dev` environment variable. After creating the cluster, the customer set multiple boot paths for each member.

9. At the lead member's console, installed Tru64 UNIX Version 5.0A. Configured basic network and time services. Loaded the TruCluster Server Version 5.0A subsets.
10. Mounted the Tru64 UNIX Version 4.0F `usr_domain#usr` on `/mnt` and copied the migration directory containing the storage information collected by `clu_migrate_save` to `/var/TruCluster_migration` on the Version 5.0A system.
11. Ran `clu_migrate_configure -x`. Examined the commands that `clu_migrate_configure` would run.
12. Ran `clu_migrate_configure`. (Because the TruCluster Software Version 1.6 Production Server cluster used neither ASE services nor LSM, `clu_migrate_configure` did not add any entries to `/etc/fstab`, mount any file systems, or create any LSM volumes.)

Used the clu_migrate_configure log file as input for a shell script that mapped the new-style dsk device names to the drd links used by the Oracle test database.

**Note**

The disk devices that were used to install Tru64 UNIX Version 5.0A and create the Version 5.0A cluster were created after the Version 1.6 cluster was shut down. Therefore, clu_migrate_save had no knowledge of those devices, and clu_migrate_configure could not map their nonexistent old-style device names to the new-style device names that were assigned to these disks when Tru64 UNIX Version 5.0A was installed.

13. Ran clu_create to create a single-member cluster.
14. Halted the Tru64 UNIX Version 5.0A system and set multiple boot paths to its cluster boot disk before booting it as a single-member cluster:

```
P00>>> set bootdef_dev dgm1.1001.0.7.7,dgm1.1002.0.7.7,\
dgn1.1003.0.10.7,dgn1.1004.0.10.7
      .
      .
      .
P00>>> init
```

15. Ran clu_add_member to complete the creation of a four-member TruCluster Server Version 5.0A cluster. As with the first member, set multiple boot paths for each member before booting it into the cluster.
16. Ran the Oracle Version 8.1.6 binaries to test whether Oracle could still access the test database that was created on the Version 1.6 Production Server cluster.

    The test was successful.

The migration was complete.

## 8.9 Upgrading a TruCluster Memory Channel Software Cluster

This section provides a generic procedure for customers who plan to upgrade a TruCluster Memory Channel Software cluster to TruCluster Server Version 5.1A.

The discussion in this section makes the following assumptions:

- The goal is a reasonably low-cost upgrade to a TruCluster Server Version 5.1A cluster. Therefore, the preference when adding storage is to use SCSI adapters, cables, and low-end storage containers such as the

UltraSCSI BA356 rather than HSZ70 RAID array controllers or Fibre Channel and HSG80 controllers.

- The Memory Channel cluster is not currently configured with the shared storage required to install TruCluster Server Version 5.1A. In all likelihood there is little or no shared storage; the only connection between members is the Memory Channel. Any storage required by a member is either internal to that member or on a private bus. The major hardware modifications will be the addition of the SCSI adapters, cables, storage containers, and disks needed to create a TruCluster Server Version 5.1A cluster. The existing Memory Channel interconnect and any external network connections will not have to be changed.
- Because most Memory Channel clusters are designed for high performance rather than high availability, the upgrade configuration is not a no-single-point-of-failure (NSPOF) cluster. When adding hardware in preparation for creating the Version 5.1A cluster, you can configure the level of redundancy that best fits your needs.

  The Memory Channel cluster has some level of built-in operating system redundancy because each member has its own operating system. (The failure of a single member's operating system disk does not disable the cluster.) After the upgrade, the TruCluster Server cluster members share the same root (`/`), `/usr`, and `/var` file systems. For this reason, some form of software or hardware RAID is required to make sure that the loss of a single disk does not disable the cluster. Because the upgraded cluster uses low-end storage that does not support hardware RAID controllers, LSM will be used to mirror the shared root (`/`), `/usr`, and `/var` file systems.
- Downtime during the upgrade is not an issue. The Memory Channel cluster will be shut down to add shared storage and to install the TruCluster Server Version 5.1A software. If your environment cannot tolerate downtime, you will have to create a separate TruCluster Server Version 5.1A cluster.

Figure 8–1 shows a basic block diagram of an eight-node cluster. This diagram and several others that show detailed cabling connections and storage layout are in the *Cluster Hardware Configuration* manual.

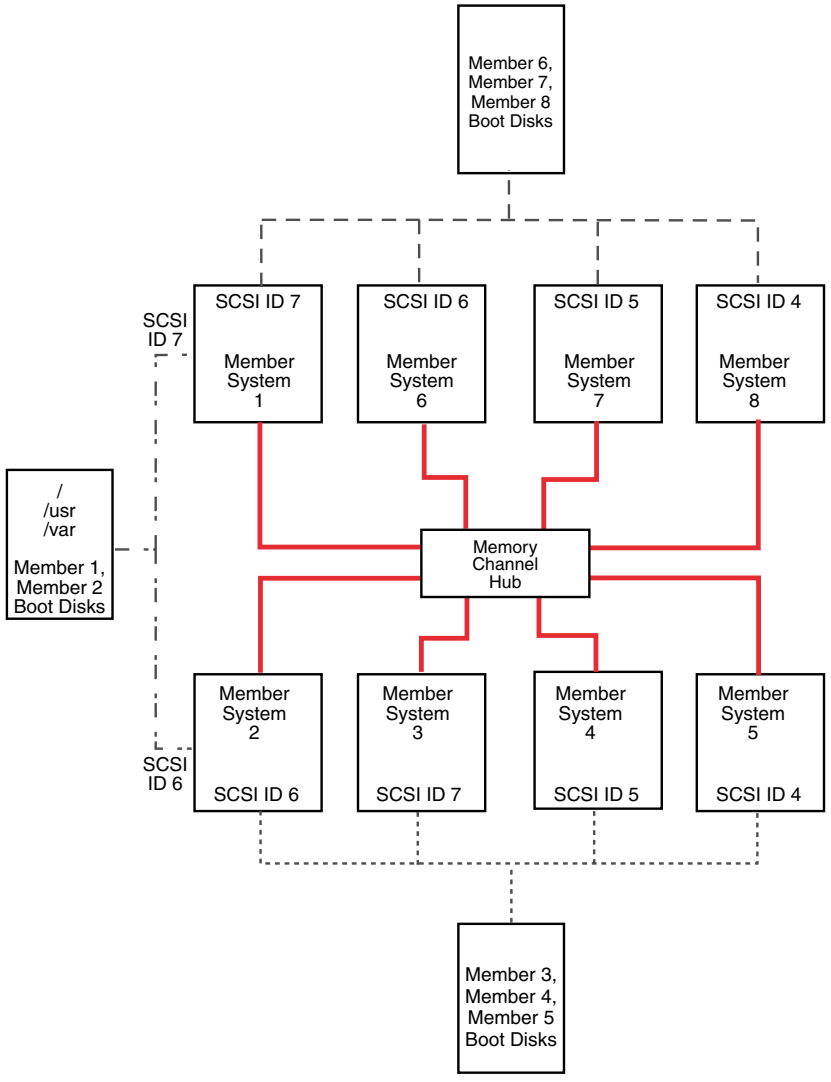
**Notes**

The *Cluster Hardware Configuration* manual has a chapter titled "Configuring an Eight-Member Cluster Using Externally Terminated Shared SCSI Buses," which contains this figure plus several other detailed configuration diagrams. The chapter also provides detailed hardware configuration information, which you need to use in conjunction with the procedure in this section.

The block diagram does not show the storage that contains the LSM mirrors of the root (/), /usr, and /var file systems. The "Introduction" chapter in the *Cluster Hardware Configuration* manual provides a figure that shows how to configure dual SCSI buses for LSM mirroring of these file systems.

---

**Figure 8–1: Block Diagram of an Eight-Node Cluster**



Because we do not know how your Memory Channel cluster is configured, the following procedure might not have all the steps you need to upgrade your cluster. Use the following steps as a starting point for designing a procedure that fits your upgrade requirements.

1. Make sure that the systems in the Memory Channel cluster are supported by TruCluster Server Version 5.1A. For information on supported systems, see the TruCluster Server Version 5.1A SPD. You can find the latest version of the SPD at the following URL:

   **http://www.tru64unix.compaq.com/docs/pub_page/spds.html**

   **Note**

   You cannot use the `clu_migrate_check` script described in Section 8.4. The script is designed for TruCluster Production Server and Available Server clusters.

2. Use the "Configuring an Eight-Member Cluster Using Externally Terminated Shared SCSI Buses" chapter in the *Cluster Hardware Configuration* manual to decide what storage hardware you need to add to the cluster. Based on the number of members in your Memory Channel cluster, decide how many shared SCSI buses you will need to perform an upgrade (you can connect up to four members to one SCSI bus). Then determine how many SCSI adapters, cables, terminators, storage shelves, and disks you will need.

   In addition, read Chapter 1 and Chapter 2 in this installation manual. Decide how to allocate disk space and whether or not to use a quorum disk. Fill out the information checklists in Appendix A.

3. Obtain the hardware, software, and licenses required for the upgrade.

   **Note**

   New versions of the operating system and cluster software usually require new versions of AlphaServer SRM firmware. You can update SRM firmware now or you can wait until you shut down the Memory Channel cluster. To keep downtime to a minimum, upgrade SRM firmware, one system at a time, before shutting down the entire Memory Channel cluster. For more information on SRM firmware, see Section 3.1.

4. Use the `sysconfig -q rm` to display the value of the `rm_rail_style` attribute. Record this value. (Most Memory Channel clusters use multiple-active rail style (`rm_rail_style=0`), but the default style for TruCluster Server Version 5.1A is failover pair (`rm_rail_style=1`).)

5. Decide which system will become the first member of the new cluster. This system must be directly connected to the storage that will contain the TruCluster Server shared root (/), /usr, and /var file systems.
6. If you plan to install Tru64 UNIX on the disk or disks that contains the current operating system, back up the current operating system before continuing.
7. Shut down and halt each system.
8. At each system's console, set the following console variables:

```
>>> set auto_action halt
>>> set bootdef_dev ""
>>> set boot_osflags A
>>> set boot_reset on
```

   For systems that support the bus_probe_algorithm variable:

```
>>> set bus_probe_algorithm new
```

   See Section 2.6 for more information on console variables. Note that setting the bus_probe_algorithm variable on systems that do not use the variable is benign. The variable is cleared at the next init or power cycle.
9. Turn each system off.

**Note**

Do not turn these systems on until instructed to do so.

10. Using the information in the *Cluster Hardware Configuration* manual, add the storage required to create a TruCluster Server Version 5.1A cluster. If you are adding or reconfiguring any other hardware, do it now. If you have not updated SRM firmware, do it now.
11. If using a Memory Channel hub, make sure that the hub is turned on.
12. Turn on the system that will become the first member of the new cluster. At the console prompt, use the console show config command to determine whether the console and adapter firmware revisions are compatible with Tru64 UNIX Version 5.1A. If they are not, update firmware as needed.
13. Following the instructions in Chapter 3, perform a full installation of Tru64 UNIX Version 5.1A.

**Note**

We strongly recommend that you do not overwrite the disk or disks containing the operating system used by the Memory

Channel cluster. If you encounter problems later on, you can quickly return this system to the Memory Channel cluster as long as these disks are intact.

14. On the Tru64 UNIX Version 5.1A system, do the following:

    a. Fully configure the Tru64 UNIX operating system. Follow the instructions in Chapter 3. (See Section 2.4.2 when configuring LSM on the base operating system.)

    b. Install applications.

       When and how you install applications depends on the types of applications and how you were using your Memory Channel cluster. Remember that the TruCluster Server Version 5.1A cluster file systems share the same name space. See the *Cluster Highly Available Applications* manual for information on running applications in a TruCluster Server Version 5.1A cluster.

    c. Install the TruCluster Server Version 5.1A license and subsets.

**Note**

We recommend that you back up the system before continuing. If something goes wrong later in the procedure, you can restore to this point faster than you can install and configure Tru64 UNIX, install applications, and load the TruCluster Server subsets.

15. Follow the procedures in Chapter 4 and run the `clu_create` command to create a single-member cluster.

16. Halt the system and boot it as a single-member cluster.

17. If the Memory Channel cluster systems had the `rm` subsystem attribute `rm_rail_style=0`, set it to `0` on the single-member TruCluster Server Version 5.1A cluster, and then reboot the system:

    a. Modify the `/etc/sysconfigtab` file to include the following stanza:

```
rm:
 rm_rail_style=0
```

    b. Reboot the single-member cluster:

```
# shutdown -r now
```

18. Use LSM to mirror the root (/), /usr, and /var file systems. (See Section 2.4.2, volmigrate(8), volencap(8), and the *Cluster Administration* manual.)
19. One at a time, add the remaining systems to the cluster. Follow these steps for each system:
    a. Turn the system on.
    b. Use the console `show config` command to determine whether the console and adapter firmware revisions are compatible with Tru64 UNIX Version 5.1A. If they are not, update firmware as needed.
    c. Following the procedure in Chapter 5, run the `clu_add_member` command on a current cluster member to create a boot disk for the new member.
    d. Boot the new member into the cluster.

# A

# Information Checklists

**Table A–1: Tru64 UNIX System Attributes**

| Attribute | Value |
|---|---|
| Host name | |
| Host name IP address | |
| Tru64 UNIX root partition (for example, `dsk0a`) | |
| Tru64 UNIX root device from console (for example, `DKA0`) | |
| Tru64 UNIX `/usr` partition (for example, `dsk0g`) | |
| Tru64 UNIX `/var` partition (for example, `dsk0h`) | |

**Table A–2: Cluster Attributes**

| Attribute | Value |
|---|---|
| Cluster name (fully qualified) | |
| Default cluster alias IP address | |
| Clusterwide root (`/`) partition (for example, `dsk1b`) | |
| Clusterwide root (`/`) disk device serial number (for example, WWID) | |
| Clusterwide `/usr` partition (for example, `dsk2c`) | |
| Clusterwide `/usr` disk device serial number (for example, WWID) | |
| Clusterwide `/var` partition (for example, `dsk3c`) | |
| Clusterwide `/var` disk device serial number (for example, WWID) | |
| If using a quorum disk, the disk device (for example, `dsk7`) | |
| If using a quorum disk, quorum disk device serial number (for example, WWID) | |
| If using a quorum disk, the number of votes assigned to the quorum disk | |

**Table A–3: Member Attributes**

| Attribute | 1st Member | 2nd Member | 3rd Member |
|---|---|---|---|
| Member host name (fully qualified)[a] | | | |
| Host name IP address[a] | | | |
| Member ID[b] (`memberid`: 1-63) | | | |
| Number of votes assigned to this member[c] | | | |
| Boot disk (for example, `dsk10`) | | | |
| Boot device from console (for example, `DKC400`) | | | |
| Boot device serial number (for example, WWID) | | | |
| Boot device physical location (bus/target/LUN) | | | |
| Virtual cluster interconnect IP name[d] | | | |
| Virtual cluster interconnect IP address[e] | | | |
| Physical cluster interconnect device name[f] | | | |
| Physical cluster interconnect IP address (LAN only)[g] | | | |
| Additional network interface IP name | | | |
| Additional network interface IP address | | | |
| **Attribute** | **4th Member** | **5th Member** | **6th Member** |
| Member host name (fully qualified)[a] | | | |
| Host name IP address[a] | | | |
| Member ID[b] (`memberid`: 1-63) | | | |
| Number of votes assigned to this member[c] | | | |
| Boot disk (for example, `dsk10`) | | | |
| Boot device from console (for example, `DKC400`) | | | |
| Boot device serial number (for example, WWID) | | | |
| Boot device physical location (bus/target/LUN) | | | |
| Virtual cluster interconnect IP name[d] | | | |

**Table A–3: Member Attributes (cont.)**

| Attribute | 1st Member | 2nd Member | 3rd Member |
|---|---|---|---|
| Virtual cluster interconnect IP address[e] | | | |
| Physical cluster interconnect device name[f] | | | |
| Physical cluster interconnect IP address (LAN only)[g] | | | |
| Additional network interface IP name | | | |
| Additional network interface IP address | | | |

[a] The first member inherits its host name and IP address from the Tru64 UNIX system.
[b] The default member ID for the first member is 1. The default member ID for each additional member is incremented by 1.
[c] If a quorum disk is configured, the default number of votes is 1.
[d] By default, the installation programs offer an IP name set to the short form of the member's host name followed by `-ics0`.
[e] By default, the installation programs offer IP addresses on the 10.0.0 subnet, with the host portion of the address set to the member ID.
[f] Memory Channel or LAN.
[g] Required only for a LAN cluster interconnect. By default, the installation programs offer IP addresses on the 10.1.0 subnet, with the host portion of the address set to the member ID. (The IP name associated with this address is automatically set by the installation programs to `member`*n*`-icstcp0`, where *n* is the member ID.)

# B

# Modifications to System Files

The following system-configuration and installation-related files are created or modified as a result of installing the TruCluster Server Version 5.1A software:

`/.rhosts`

If the file does not exist, `clu_create` creates it. The `clu_create` command adds the fully qualified host name associated with the cluster alias. The `clu_create` and `clu_add_member` commands add the IP name (or IP names, for LAN interconnects) associated with each cluster interconnect interface.

`/cluster/admin/.member`*n*`.cfg`

These files are installation-configuration files; there is one for each member. The files are created by and appended to by `clu_create` or `clu_add_member`. Each file contains a list of name/value pairs representing the selections made when running `clu_create` or `clu_add_member` for that particular member.

These files are maintained by `clu_create` and `clu_add_member`. Do not edit them.

`/cluster/admin/.member.list`

This text file is used by `clu_create` and `clu_add_member` to keep track of assigned cluster member IDs.

This file is maintained by `clu_create` and `clu_add_member`. Do not edit it.

`/cluster/admin/clu_check_config.log`

This log file is maintained by `clu_check_config`.

`/cluster/admin/clu_create.log`

This log file is maintained by `clu_create`. Each time you run `clu_create`, it appends a log of the session to this file.

`/cluster/admin/clu_add_member.log`

This log file is maintained by `clu_add_member`. Each time you run `clu_add_member`, it appends a log of the session to this file.

`/cluster/admin/clu_upgrade`

This directory is where `clu_upgrade` stores log files from completed upgrades.

`/cluster/admin/clu_upgrade.log`

This log file is maintained by `clu_upgrade`. When a rolling upgrade completes, `clu_upgrade` moves the log file to the `/cluster/admin/clu_upgrade/history/`*release_version* directory.

`/etc/fdmns`

The `clu_create` command creates the domain directories for the clusterwide root (`/`), `/usr`, `/var`, and optionally, `i18n` file systems, and for the first cluster member. The `clu_add_member` command creates a domain directory for each additional member.

`/etc/gated.conf`

If `gated` is configured on the Tru64 UNIX system, the file is copied to the first member and propagated to the remaining members. If `gated` is not configured, it is created for each member by the installation scripts. In a cluster, `/etc/gated.conf` is a CDSL whose target file is `/cluster/members/{memb}/etc/gated.conf`.

**Note**

The `aliasd` daemon creates a `/etc/gated.conf.member`*n* for each cluster member. The daemon starts `gated` using this file as `gated`'s configuration file rather than the member's `/cluster/members/{memb}/etc/gated.conf` file.

`/etc/hosts`

The `clu_create` and `clu_add_member` commands add the IP address and interface name (or addresses and names, for LAN interconnects) associated with each member's cluster interconnect interface. The `clu_create` command adds the IP address, fully qualified host name, and unqualified host name for the default cluster alias.

/etc/ifaccess.conf

Both clu_create and clu_add_member add deny entries for all noncluster-interconnect network interfaces on a cluster member. The deny entries prevent IP packets coming from outside the cluster from being directed to the cluster interconnect through these interfaces.

/etc/inittab

Both clu_create and clu_add_member add a sysconfig command line for cms (cluster mount service).

/etc/rc.config

This file is copied by clu_create from the system running Tru64 UNIX to the first cluster member. Created from .new..rc.config by clu_add_member for all other members. Table B–1 lists the configuration variables that are set during installation.

/etc/sysconfigtab

The installation scripts initialize several sysconfigtab attributes. For some attributes, the scripts prompt for information; for others, the scripts silently set the values. (See Table B–2 for a list of configuration variables initialized by the installation procedure.)

/etc/sysconfigtab.cluster

The clu_create and clu_add_member commands set the value of cluster_expected_votes.

Table B–3 lists the configuration variables initialized by the installation procedure.

/sys/conf/*HOSTNAME*.list

This file registers the cluster software with the kernel before the installation procedure calls the doconfig program. *HOSTNAME* is the host name of the system.

/sys/*HOSTNAME*/vmunix

A new kernel is built and placed in /sys/*HOSTNAME*/vmunix, where *HOSTNAME* is the name of the configuration file specified during installation. The clu_create and clu_add_member commands copy each member's vmunix file to that member's boot partition.

Table B–1 lists the /etc/rc.config variables that are created or modified by clu_create or clu_add_member.

**Table B–1: /etc/rc.config Variables**

| Variable | Comment |
|---|---|
| BASE_VERSION | Version of the Tru64 UNIX operating system on which TruCluster Server is installed. If set on the current member, set by clu_add_member to the value of the current member. |
| CLU_BOOT_FILESYSTEM | Domain and fileset for this member's boot disk. Set by both clu_create and clu_add_member. |
| CLU_NEW_MEMBER | Set to 1 during installation; set to 0 after the member boots into the cluster for the first time. |
| CLU_VERSION | Version number of the installed TruCluster Server product. If set on the current member, set by clu_add_member to the value of the current member. |
| CLUSTER_NET | IP name of the system's virtual cluster interconnect interface. The installation procedure adds a CLUSTER_NET entry and sets its value to the user-specified interface name. |
| GATED | If neither GATED nor ROUTED is set to yes on the Tru64 UNIX system, clu_create sets GATED to yes. Set by clu_add_member to the value of the current member. |
| GATED_FLAGS | Set by clu_add_member to the value of the current member. |
| HOSTNAME | Set to the user-specified value when adding a member. |
| IFCONFIG_*n* | IP address and the netmask for the cluster virtual interconnect interface. The installation procedure adds an IFCONFIG_*n* entry. |
| IMC_AUTO_INIT | If set to 1, libimc will be automatically initialized whenever the system boots. This initialization involves reserving approximately 4.5 MB for the library. The default value at installation is 0; the library is not initialized. If set on the current member, set by clu_add_member to the value of the current member. |
| IMC_MAX_ALLOC | Determines the maximum aggregate amount, in MB, of Memory Channel address space that libimc can allocate for its use across the cluster. If the value of this variable differs among cluster members, the largest value specified is used. The default value at installation is 10. If set on the current member, set by clu_add_member to the value of the current member. |
| IMC_MAX_RECV | Determines the maximum aggregate amount, in MB, of physical memory that libimc can map for reading Memory Channel address space. This limit is node-specific. The default value at installation is 10. If set on the current member, set by clu_add_member to the value of the current member. |

**Table B–1: /etc/rc.config Variables (cont.)**

| Variable | Comment |
|---|---|
| MAX_NETDEVS | If not set, set to 24 by clu_add_member. |
| NETDEV_*n* | Network device name (ics0) for the system's virtual cluster interconnect interface. The installation procedure adds a NETDEV_*n* entry. For a LAN interconnect, if you specify an existing NetRAIN set, the installation procedures use the NETDEV_*n* entry associated with that set. |
| NRCONFIG_*n* | Interfaces in a NetRAIN interface set. If the member uses an existing NetRAIN set for a LAN interconnect, cleared by clu_create. |
| NRDEV_*n* | NetRAIN interface name. If the member uses an existing NetRAIN set for a LAN interconnect, cleared by clu_create. |
| NR_DEVICES*n* | Number of NetRAIN devices. If the member uses an existing NetRAIN set for a LAN interconnect, cleared by clu_create. |
| NUM_NETCONFIG | Number of configured network devices. The clu_create command increments the value of NUM_NETCONFIG, unless you specify an existing NetRAIN set for the cluster interconnect. The clu_add_member command sets the value of NUM_NETCONFIG to 1. |
| PAGERAW | Set to 1 by clu_add_member. |
| ROUTER | If set on the current member, set by clu_add_member to the value of the current member. |
| TCR_INSTALL | If set to TCR, the installation was successful. If set to BAD, the installation was unsuccessful. If set on the current member, set by clu_add_member to the value of the current member. |
| TCR_PACKAGE | Set to TCR. If set on the current member, set by clu_add_member to the value of the current member. |
| XLOGIN | If set on the current member, set by clu_add_member to the value of the current member. |
| XNTPD_CONF | If set on the current member, set by clu_add_member to the value of the current member. If not set on the current member, set to YES. |

**Table B–1: /etc/rc.config Variables (cont.)**

| Variable | Comment |
|---|---|
| XNTPD_OPTS | If set on the current member, set by clu_add_member to the value of the current member. If not set on the current member, set to -g. |
| XNTP_SERV*n* | All members of the cluster are configured as peers of each other. A fudge command is placed in each member's ntp.conf file and, if fewer than three servers are configured, server 127.127.1.0 is added to each member's rc.config file. |

Table B–2 lists the /etc/sysconfigtab variables that are modified or created by clu_create or clu_add_member.

**Table B–2: /etc/sysconfigtab Attributes**

| Subsystem | Attribute | Comment |
|---|---|---|
| clubase | cluster_interconnect | Interconnect type used for cluster internode communications. If a LAN interconnect is used, the clu_create and clu_add_member commands set this attribute to tcp. If Memory Channel is used, the attribute is set it to mct. |
| | cluster_name | Unqualified cluster name. Set by both clu_create and clu_add_member. |
| | cluster_expected_votes | Sum of all votes held by the cluster members plus the vote assigned to the quorum disk (if configured). |
| | cluster_node_inter_name | IP name associated with this member's cluster interconnect interface; for example, polishham-ics0. The installation procedure prompts for this information. |
| | cluster_node_name | Unqualified host name of this member. |
| | cluster_node_votes | Fixed number of votes that a node contributes toward quorum. Each node with a nonzero value is considered a voting member. |
| | cluster_qdisk_major | Major device number of the h partition on the quorum disk. |

**Table B–2: /etc/sysconfigtab Attributes (cont.)**

| Subsystem | Attribute | Comment |
|---|---|---|
| | cluster_qdisk_minor | Minor device number of the h partition on the quorum disk. |
| | cluster_qdisk_votes | Fixed number of votes that the quorum disk contributes toward quorum. |
| | cluster_seqdisk_major | Major device number of the h partition on a member's boot disk. |
| | cluster_seqdisk_minor | Minor device number of the h partition on a member's boot disk. |
| drd | drd_nopr_list | First entry on the list of hardware IDs for devices on which the device request dispatcher subsystem will not perform persistent reservations. If set on the current member, set by clu_add_member to the value of the current member. |
| generic | act_vers_high | Version number used during an upgrade to synchronize changes across the cluster. Do not modify. |
| | act_vers_low | Version number used during an upgrade to synchronize changes across the cluster. Do not modify. |
| | memberid | Cluster memberid for this member. Each member must have a unique memberid (1–63). The installation procedure prompts for the value of this attribute. |
| | msgbuf_size | If less than 16384, set to 16384 by clu_create. Set to the value of the current member by clu_add_member. |
| | new_vers_high | Version number used during an upgrade to synchronize changes across the cluster. Do not modify. |
| | new_vers_low | Version number used during an upgrade to synchronize changes across the cluster. Do not modify. |

**Table B–2: /etc/sysconfigtab Attributes (cont.)**

| Subsystem | Attribute | Comment |
|---|---|---|
| | rolls_ver_lookup | Set to 0 by clu_create and clu_add_member. Set to 1 by clu_upgrade for members using tagged files during a rolling upgrade or patch. Reset to 0 by clu_upgrade when the member has rolled. Do not modify. |
| ics_ll_tcp | ics_tcp_inetaddr0 | Set to the IP address of the physical cluster interconnect device (for example, 10.1.0.1) by clu_create and clu_add_member. (LAN interconnect only) |
| | ics_tcp_netmask0 | Subnet mask for the cluster interconnect. Set by clu_create and clu_add_member to 255.255.255.0. (LAN interconnect only) |
| | ics_tcp_adapter0 | Name of the physical cluster interconnect device (for example, tu0 or nr0). Set by clu_create and clu_add_member. (LAN interconnect only) |
| | ics_tcp_nr0 | If the ics_tcp_adapter0 attribute indicates a NetRAIN set, this attribute is an array indicating the device names of the network adapters that make up the set. (LAN interconnect only) |
| lsm | lsm_rootdev_is_volume | Set to 0 by clu_create and clu_add_member. |
| sec | acl_mode | If set, set to the value of the current system or member. If not set, set to disable. Set by clu_create and clu_add_member. |
| vm | swapdevice | Swap device for this member. The installation procedure uses the b partition of the device you specify as this member's boot disk as the value of this attribute. |
| | vm_page_free_reserved | Set to 20 by clu_create and clu_add_member. |
| | vm_page_free_min | Set to 30 by clu_create and clu_add_member. |

Table B–3 lists the /etc/sysconfigtab.cluster variables that are modified or created by clu_create or clu_add_member.

**Table B–3: /etc/sysconfigtab.cluster Attributes**

| Subsystem | Attribute | Comment |
|---|---|---|
| clubase | cluster_expected_votes | Sum of all votes held by the cluster members plus the vote assigned to the quorum disk (if configured). Set by clu_create and clu_add_member. |

# C

# Installation Examples

This chapter provides samples of the logs written by:

- `clu_create` (Section C.1)
- `clu_add_member` (Section C.2)
- `clu_upgrade` (Section C.3)

## C.1 clu_create Log

Each time you run `clu_create`, it writes log messages to `/cluster/admin/clu_create.log`. Example C–1 shows a sample `clu_create` log file.

**Example C–1: Sample clu_create Log File**

```
Do you want to continue creating the cluster? [yes]: Return

Each cluster has a unique cluster name, which is a hostname
used to identify the entire cluster.

Enter a fully-qualified cluster name []: deli.zk3.dec.com
Checking cluster name: deli.zk3.dec.com

You entered 'deli.zk3.dec.com' as your cluster name.
Is this correct? [yes]: Return

The cluster alias IP address is the IP address associated with the
default cluster alias.  (192.168.168.1 is an example of an IP address.)

Enter the cluster alias IP address []: 16.140.112.209
Checking cluster alias IP address: 16.140.112.209

You entered '16.140.112.209' as the IP address for the default cluster alias.
Is this correct? [yes]: Return

The cluster root partition is the disk partition (for example, dsk4b)
that will hold the clusterwide root (/) file system.

    Note: The default 'a' partition on most disks is not large
    enough to hold the clusterwide root AdvFS domain.

Enter the device name of the cluster root partition []: dsk1b
Checking the cluster root partition: dsk1b

You entered 'dsk1b' as the device name of the cluster root partition.
Is this correct? [yes]: Return

The cluster usr partition is the disk partition (for example, dsk4g)
that will contain the clusterwide usr (/usr) file system.
```

## Example C–1: Sample clu_create Log File (cont.)

```
    Note: The default 'g' partition on most disks is usually
    large enough to hold the clusterwide usr AdvFS domain.

Enter the device name of the cluster usr partition []: dsk2c
Checking the cluster usr partition: dsk2c

You entered 'dsk2c' as the device name of the cluster usr partition.
Is this correct? [yes]: Return

The cluster var device is the disk partition (for example, dsk4h)
that will hold the clusterwide var (/var) file system.

    Note: The default 'h' partition on most disks is usually
    large enough to hold the clusterwide var AdvFS domain.

Enter the device name of the cluster var partition []: dsk3c
Checking the cluster var partition: dsk3c

You entered 'dsk3c' as the device name of the cluster var partition.
Is this correct? [yes]: Return

Do you want to define a quorum disk device at this time? [yes]: Return
The quorum disk device is the name of the disk (for example, 'dsk5')
that will be used as this cluster quorum disk.

Enter the device name of the quorum disk []: dsk7
Checking the quorum disk device: dsk7

You entered 'dsk7' as the device name of the quorum disk device.
Is this correct? [yes]: Return

By default the quorum disk is assigned '1' vote(s).
To use this default value, press Return at the prompt.

The number of votes for the quorum disk is an integer usually 0 or 1.
If you select 0 votes then the quorum disk will not contribute votes to the
cluster. If you select 1 vote then the quorum disk must be accessible to
boot and run a single member cluster.

Enter the number of votes for the quorum disk [1]: Return
Checking number of votes for the quorum disk: 1

You entered '1' as the number votes for the quorum disk.
Is this correct? [yes]: Return

The default member ID for the first cluster member is '1'.
To use this default value, press Return at the prompt.

A member ID is used to identify each member in a cluster.
Each member must have a unique member ID, which is an integer in
the range 1-63, inclusive.

Enter a cluster member ID [1]: Return
Checking cluster member ID: 1

You entered '1' as the member ID.
Is this correct? [yes]: Return

By default the 1st member of a cluster is assigned '1' vote(s).
Checking number of votes for this member: 1
```

## Example C–1: Sample clu_create Log File (cont.)

```
Each member has its own boot disk, which has an associated
device name; for example, 'dsk5'.

Enter the device name of the member boot disk []: dsk10
Checking the member boot disk: dsk10

You entered 'dsk10' as the device name of this member's boot disk.
Is this correct? [yes]: Return

Device 'ics0' is the default virtual cluster interconnect device
Checking virtual cluster interconnect device: ics0

The virtual cluster interconnect IP name 'pepicelli-ics0' was formed by
appending '-ics0' to the system's hostname.
To use this default value, press Return at the prompt.

Each virtual cluster interconnect interface has a unique IP name (a
hostname) associated with it.

Enter the IP name for the virtual cluster interconnect [pepicelli-ics0]: Return
Checking virtual cluster interconnect IP name: pepicelli-ics0

You entered 'pepicelli-ics0' as the IP name for the virtual cluster interconnect.
Is this name correct? [yes]: Return

The virtual cluster interconnect IP address '10.0.0.1' was created by
replacing the last byte of the default virtual cluster interconnect network
address '10.0.0.0' with the previously chosen member ID '1'.
To use this default value, press Return at the prompt.

The virtual cluster interconnect IP address is the IP address
associated with the virtual cluster interconnect IP name.  (192.168.168.1
is an example of an IP address.)

Enter the IP address for the virtual cluster interconnect [10.0.0.1]: Return
Checking virtual cluster interconnect IP address: 10.0.0.1

You entered '10.0.0.1' as the IP address for the virtual cluster interconnect.
Is this address correct? [yes]: Return

What type of cluster interconnect will you be using?

    Selection   Type of Interconnect
------------------------------------------------------------------------
         1      Memory Channel
         2      Local Area Network
         3      None of the above
         4      Help
         5      Display all options again
------------------------------------------------------------------------
Enter your choice [1]: 1
You selected option '1' for the cluster interconnect
Is that correct? (y/n) [y]: Return

Device 'mc0' is the default physical cluster interconnect interface device
Checking physical cluster interconnect interface device name(s): mc0


You entered the following information:
```

## Example C–1: Sample clu_create Log File (cont.)

```
    Cluster name:                                          deli.zk3.dec.com
    Cluster alias IP Address:                              16.140.112.209
    Clusterwide root partition:                            dsk1b
    Clusterwide usr  partition:                            dsk2c
    Clusterwide var  partition:                            dsk3c
    Clusterwide i18n partition:                            Directory-In-/usr
    Quorum disk device:                                    dsk7
    Number of votes assigned to the quorum disk:           1
    First member's member ID:                              1
    Number of votes assigned to this member:               1
    First member's boot disk:                              dsk10
    First member's virtual cluster interconnect device name: ics0
    First member's virtual cluster interconnect IP name:     pepicelli-ics0
    First member's virtual cluster interconnect IP address:  10.0.0.1
    First member's physical cluster interconnect devices     mc0
    First member's NetRAIN device name                       Not-Applicable
    First member's physical cluster interconnect IP address  Not-Applicable

If you want to change any of the above information, answer 'n' to the
following prompt. You will then be given an opportunity to change your
selections.
Do you want to continue to create the cluster? [yes]: Return

Creating required disk labels.
  Creating disk label on member disk : dsk10
  Initializing cnx partition on member disk : dsk10h
  Creating disk label on quorum disk : dsk7
  Initializing cnx partition on quorum disk : dsk7h

Creating AdvFS domains:
  Creating AdvFS domain 'root1_domain#root' on partition '/dev/disk/dsk10a'.
  Creating AdvFS domain 'cluster_root#root' on partition '/dev/disk/dsk1b'.
  Creating AdvFS domain 'cluster_usr#usr' on partition '/dev/disk/dsk2c'.
  Creating AdvFS domain 'cluster_var#var' on partition '/dev/disk/dsk3c'.

Populating clusterwide root, usr, and var file systems:
  Copying root file system to 'cluster_root#root'.
...
  Copying usr file system to 'cluster_usr#usr'.
......................
  Copying var file system to 'cluster_var#var'.
..

Creating Content Dependent Symbolic Links (CDSLs) for file systems:
  Creating CDSLs in root file system.
  Creating CDSLs in usr  file system.
  Creating CDSLs in var  file system.
  Creating links between clusterwide file systems

Populating member's root file system.

Modifying configuration files required for cluster operation:
  Creating /etc/fstab file.
  Configuring cluster alias.
  Updating /etc/hosts - adding IP address '16.140.112.209' \
    and hostname 'deli.zk3.dec.com'
  Updating member-specific /etc/inittab file with 'cms' entry.
  Updating /etc/hosts - adding IP address '10.0.0.1' and hostname 'pepicelli-ics0'
  Updating /etc/rc.config file.
  Updating /etc/sysconfigtab file.
  Retrieving cluster_root major and minor device numbers.
```

**Example C–1: Sample clu_create Log File (cont.)**

```
  Creating cluster device file CDSLs.
  Updating /.rhosts - adding hostname 'deli.zk3.dec.com'.
  Updating /etc/hosts.equiv - adding hostname 'deli.zk3.dec.com'
  Updating /.rhosts - adding hostname 'pepicelli-ics0'.
  Updating /etc/hosts.equiv - adding hostname 'pepicelli-ics0'
  Updating /etc/ifaccess.conf - adding deny entry for 'sl0'
  Updating /etc/ifaccess.conf - adding deny entry for 'tu0'
  Updating /etc/ifaccess.conf - adding deny entry for 'tun0'
  Updating /etc/cfgmgr.auth - adding hostname 'pepicelli.zk3.dec.com'
  Finished updating member1-specific area.

Building a kernel for this member.
  Saving kernel build configuration.
  The kernel will now be configured using the doconfig program.

*** KERNEL CONFIGURATION AND BUILD PROCEDURE ***

Saving /sys/conf/PEPICELLI as /sys/conf/PEPICELLI.bck


*** PERFORMING KERNEL BUILD ***
 Working....Tue May  8 15:54:11 EDT 2001

The new kernel is /sys/PEPICELLI/vmunix
  Finished running the doconfig program.

  The kernel build was successful and the new kernel
   has been copied to this member's boot disk.
  Restoring kernel build configuration.

Updating console variables
  Setting console variable 'bootdef_dev' to dsk10
  Setting console variable 'boot_dev' to dsk10
  Setting console variable 'boot_reset' to ON
  Saving console variables to non-volatile storage

clu_create: Cluster created successfully.

To run this system as a single member cluster it must be rebooted.
If you answer yes to the following question clu_create will reboot the
system for you now. If you answer no, you must manually reboot the
system after clu_create exits.
Would you like clu_create to reboot this system now? [yes]: Return
Shutdown at 15:56 (in 0 minutes) [pid 23642]
```

## C.2 clu_add_member Log

Each time you run `clu_add_member`, it writes log messages to `/cluster/admin/clu_add_member.log`. Example C–2 shows a sample `clu_add_member` log file.

## Example C–2: Sample clu_add_member Log File

```
Do you want to continue adding this member? [yes]: Return

Each cluster member has a hostname, which is assigned to the HOSTNAME
variable in /etc/rc.config.

Enter the new member's fully qualified hostname []: polishham.zk3.dec.com
Checking member's hostname: polishham.zk3.dec.com

You entered 'polishham.zk3.dec.com' as this member's hostname.
Is this name correct? [yes]: Return

The next available member ID for a cluster member is '2'.
To use this default value, press Return at the prompt.

A member ID is used to identify each member in a cluster.
Each member must have a unique member ID, which is an integer in
the range 1-63, inclusive.

Enter a cluster member ID [2]: Return
Checking cluster member ID: 2

You entered '2' as the member ID.
Is this correct? [yes]: Return

By default, when the current cluster's expected votes are greater then 1,
each added member is assigned 1 vote(s). Otherwise, each added member is
assigned 0 (zero) votes.
To use this default value, press Return at the prompt.

The number of votes for a member is an integer usually 0 or 1
Enter the number of votes for this member [1]: Return
Checking number of votes for this member: 1

You entered '1' as the number votes for this member.
Is this correct? [yes]: Return

Each member has its own boot disk, which has an associated
device name; for example, 'dsk5'.

Enter the device name of the member boot disk []: dsk12
Checking the member boot disk: dsk12

You entered 'dsk12' as the device name of this member's boot disk.
Is this correct? [yes]: Return

Device 'ics0' is the default virtual cluster interconnect device
Checking virtual cluster interconnect device: ics0

The virtual cluster interconnect IP name 'polishham-ics0' was formed by
appending '-ics0' to the system's hostname.
To use this default value, press Return at the prompt.

Each virtual cluster interconnect interface has a unique IP name (a
hostname) associated with it.

Enter the IP name for the virtual cluster interconnect [polishham-ics0]: Return
Checking virtual cluster interconnect IP name: polishham-ics0

You entered 'polishham-ics0' as the IP name for the virtual cluster interconnect.
Is this name correct? [yes]: Return

The virtual cluster interconnect IP address '10.0.0.2' was created by
```

## Example C–2: Sample clu_add_member Log File (cont.)

```
replacing the last byte of the virtual cluster interconnect network address
'10.0.0.0' with the previously chosen member ID '2'.
To use this default value, press Return at the prompt.

The virtual cluster interconnect IP address is the IP address
associated with the virtual cluster interconnect IP name.  (192.168.168.1
is an example of an IP address.)

Enter the IP address for the virtual cluster interconnect [10.0.0.2]: Return
Checking virtual cluster interconnect IP address: 10.0.0.2

You entered '10.0.0.2' as the IP address for the virtual cluster interconnect.
Is this address correct? [yes]: Return

Device 'mc0' is the default physical cluster interconnect interface device
To use this default value, press Return at the prompt.

The physical cluster interconnect interface device is the name of the
physical device(s) which will be used for low level cluster node
communications. Examples of the physical cluster interconnect interface
device name are: tu0, ee0, and nr0.

Enter the physical cluster interconnect device name(s) [mc0]: Return
Checking physical cluster interconnect interface device name(s): mc0

You entered 'mc0' as your physical cluster interconnect interface
device name(s). Is this correct? [yes]: Return

Each cluster member must have its own registered TruCluster Server
license. The data required to register a new member is typically located on
the License PAK certificate or it may have been previously placed on your
system as a partial or complete license data file. If you are prepared to
enter this license data at this time, clu_add_member can configure the new
member to use this license data. If you do not have the license data at this
time you can enter this data on the new member when it is up and running.
Do you wish to register the TruCluster Server license for this new member at
this time? [yes]: no

You entered the following information:

    Member's hostname:                                polishham.zk3.dec.com
    Member's ID:                                      2
    Number of votes assigned to this member:          1
    Member's boot disk:                               dsk12
    Member's virtual cluster interconnect devices:    ics0
    Member's virtual cluster interconnect IP name:    polishham-ics0
    Member's virtual cluster interconnect IP address: 10.0.0.2
    Member's physical cluster interconnect devices:   mc0
    Member's NetRAIN device name:                     Not-Applicable
    Member's physical cluster interconnect IP address: Not-Applicable
    Member's cluster license:                         Not Entered

If you want to change any of the above information answers 'n' to the
following prompt. You will then be given an opportunity to change your
selections.
Do you want to continue to add this member? [yes]: Return

Creating required disk labels.
  Creating disk label on member disk : dsk12
  Initializing cnx partition on member disk : dsk12h
```

## Example C–2: Sample clu_add_member Log File (cont.)

```
Creating AdvFS domains:
  Creating AdvFS domain 'root2_domain#root' on partition '/dev/disk/dsk12a'.

Creating cluster member-specific files:
  Creating new member's root member-specific files
  Creating new member's usr  member-specific files
  Creating new member's var  member-specific files
  Creating new member's boot member-specific files

Modifying configuration files required for new member operation:
  Updating /etc/hosts - adding IP address '10.0.0.2' and hostname 'polishham-ics0'
  Updating /etc/rc.config
  Updating /etc/sysconfigtab
  Updating member-specific /etc/inittab file with 'cms' entry.
  Updating /etc/securettys - adding ptys entry
  Updating /.rhosts - adding hostname 'polishham-ics0'
  Updating /etc/hosts.equiv - adding hostname 'polishham-ics0'
  Updating /etc/cfgmgr.auth - adding hostname 'polishham.zk3.dec.com'
  Configuring cluster alias.
  Configuring Network Time Protocol for new member
  Adding interface 'pepicelli-ics0' as an NTP peer to member 'polishham.zk3.dec.com'
  Adding interface 'polishham-ics0' as an NTP peer to member 'pepicelli.zk3.dec.com'

Configuring automatic subset configuration and kernel build.

clu_add_member: Initial member 2 configuration completed successfully.
From the newly added member's console, perform the following steps to
complete the newly added member's configuration:

  1. Set the console variable 'boot_osflags' to 'A'.
  2. Identify the console name of the newly added member's boots device.

     >>> show device

     The newly added member's boot device has the following properties:

     Manufacturer: DEC
     Model: HSG80
     Target: IDENTIFIER=4
     Lun: UNKNOWN
     Serial Number: SCSI-WWID:01000010:6000-1fe1-0006-3f10-0009-0270-0619-0005

     Note: The SCSI bus number may differ when viewed from different members.

  3. Boot the newly added member using genvmunix:

      >>> boot -file genvmunix <new-member-boot-device>

     During this initial boot the newly added member will:

     o  Configure each installed subset.

     o  Attempt to build and install a new kernel. If the system cannot
        build a kernel, it starts a shell where you can attempt to build
        a kernel manually. If the build succeeds, copy the new kernel to
        /vmunix. When you are finished exit the shell using ^D or 'exit'.

     o  The newly added member will attempt to set boot related console
        variables and continue to boot to multi-user mode.

     o  After the newly added member boots you should setup your system
```

**Example C–2: Sample clu_add_member Log File (cont.)**

```
        default network interface using the appropriate system management
        command.
```

## C.3 clu_upgrade Log

Each time you perform a rolling upgrade, `clu_upgrade` writes log messages to `/cluster/admin/clu_upgrade.log`. When the rolling upgrade is complete, `clu_upgrade` moves the log file to the `/cluster/admin/clu_upgrade/history/`*`release_version`* directory. Example C–3 shows a sample `clu_upgrade` log file for a rolling upgrade of a cluster from Version 5.1 to Version 5.1A. (The log is slightly reformatted for readability).

**Example C–3: Sample clu_upgrade Log File**

```
################################################################################
clu_upgrade Command: upgrade Stage:
      On Host: pepicelli.zk3.dec.com Started at: Thu May 17 13:34:23 EST 2001
--------------------------------------------------------------------------------
Retrieving cluster status.
Retrieving upgrade-related system configuration.

This is the cluster upgrade program.
You have indicated that you want to perform the 'setup' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return
Marking stage 'setup' as 'started'.

What type of upgrade will be performed?

1) Rolling upgrade using the installupdate command
2) Rolling patch using the dupatch command
3) Both a rolling upgrade and a rolling patch
4) Exit cluster software upgrade

Enter your choice:1

Backing up member-specific data for member: 1
...
Creating tagged files.
.....................................................................\
.....................................................................\
.....................................................................\
.....................................................
.........................................
The cluster upgrade 'setup' stage has completed successfully.
Reboot all cluster members except member: '1'
Marking stage 'setup' as 'completed'.

The 'setup' stage of the upgrade has completed successfully.
--------------------------------------------------------------------------------
clu_upgrade Command: upgrade Stage: setup
     On Host: pepicelli.zk3.dec.com Finished at: Thu May 17 15:48:12 EST 2001
```

## Example C–3: Sample clu_upgrade Log File (cont.)

```
################################################################################
clu_upgrade Command: boot Stage: preinstall
      On Host: polishham.zk3.dec.com Started at: Thu May 17 16:01:15 EST 2001
--------------------------------------------------------------------------------
Retrieving cluster status.
Retrieving upgrade-related system configuration.

################################################################################
clu_upgrade Command: boot Stage: preinstall
      On Host: pepicelli.zk3.dec.com Started at: Thu May 17 16:16:16 EST 2001
--------------------------------------------------------------------------------
Retrieving cluster status.
Retrieving upgrade-related system configuration.

################################################################################
clu_upgrade Command: upgrade Stage:
      On Host: pepicelli.zk3.dec.com Started at: Thu May 17 16:21:18 EST 2001
--------------------------------------------------------------------------------
Retrieving cluster status.
Retrieving upgrade-related system configuration.

This is the cluster upgrade program.
You have indicated that you want to perform the 'preinstall' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return
Checking tagged files.
.....................................................................\
.....................................................
Marking stage 'preinstall' as 'started'.
Enter the full pathname of the cluster kit mount point ['???']: /mnt/TruCluster/kit

A cluster kit has been found in the following location:

/mnt/TruCluster/kit

This kit has the following version information:

'Tru64 UNIX TruCluster(TM) Server Software X5.1A-8 (Rev 1139) autokit'

Is this the correct cluster kit for the update being performed? [yes]: Return
Saving cluster kit '/mnt/TruCluster/kit' to '/var/adm/update/TruClusterKit/'.
Marking stage 'preinstall' as 'completed'.
The cluster upgrade 'preinstall' stage has completed successfully.
On the lead member, perform the following steps before running
the installupdate command:

# shutdown now

[NOTE: As stated in the TruCluster Server Version 5.1 Release Notes,
the correct method for taking a cluster member to single-user mode is
to halt the member, and then boot it to single-user mode.]

In single-user mode enter:

# /sbin/bcheckrc

The 'preinstall' stage of the upgrade has completed successfully.
--------------------------------------------------------------------------------
clu_upgrade Command: upgrade Stage: preinstall
     On Host: pepicelli.zk3.dec.com Finished at: Fri May 18 08:49:48 EST 2001
```

## Example C–3: Sample clu_upgrade Log File (cont.)

```
################################################################################
clu_upgrade Command: check Stage: install
     On Host:  Started at: Fri May 18 14:18:58 EST 2001
--------------------------------------------------------------------------------
Retrieving cluster status.
Retrieving upgrade-related system configuration.

Checking install...
The 'install'  stage of cluster upgrade is ready to be run.
################################################################################
clu_upgrade Command: check Stage: install
            On Host:  Started at: Fri May 18 14:21:43 EST 2001
--------------------------------------------------------------------------------
Retrieving cluster status.
Retrieving upgrade-related system configuration.

Checking install...
The 'install'  stage of cluster upgrade is ready to be run.
################################################################################
clu_upgrade Command: clu_upgrade boot
      On Host: pepicelli.zk3.dec.com Invoked at: Fri May 18 17:48:21 EST 2001
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
clu_upgrade Command: clu_upgrade boot
      On Host: pepicelli.zk3.dec.com Exited at: Fri May 18 17:48:24 EST 2001
################################################################################
clu_upgrade Command: clu_upgrade upgrade postinstall
      On Host: pepicelli.zk3.dec.com Invoked at: Sun May 20 16:44:10 EST 2001
--------------------------------------------------------------------------------

This is the cluster upgrade program.
You have indicated that you want to perform the 'postinstall' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return
Marking stage 'postinstall' as 'started'.
Marking stage 'postinstall' as 'completed'.

The 'postinstall' stage of the upgrade has completed successfully.
--------------------------------------------------------------------------------
clu_upgrade Command: clu_upgrade upgrade postinstall
       On Host: pepicelli.zk3.dec.com Exited at: Sun May 20 16:44:19 EST 2001
################################################################################
clu_upgrade Command: clu_upgrade upgrade roll
            On Host:  Invoked at: Mon May 21 09:41:00 EST 2001
--------------------------------------------------------------------------------

This is the cluster upgrade program.
You have indicated that you want to perform the 'roll' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return

Backing up member-specific data for member: 2

*** START UPDATE INSTALLATION (Mon May 21 09:42:43 EST 2001) ***
    FLAGS:

Checking for installed supplemental hardware support...

Completed check for installed supplemental hardware support
```

## Example C–3: Sample clu_upgrade Log File (cont.)

```
Checking for retired hardware...done.

Initializing new version information (OSF)...done
Initializing new version information (TCR)...done
Initializing the list of member specific files for member2...done

Update Installation has detected the following update installable
products on your system:

 Tru64 UNIX V5.1 Operating System ( Rev 732 )
 Tru64 UNIX TruCluster(TM) Server Software V5.1 (Rev 389)

These products will be updated to the following versions:

 Tru64 UNIX X5.1A-8 Operating System ( Rev 1751 )
 Tru64 UNIX TruCluster(TM) Server Software X5.1A-8 (Rev 1139) autokit

It is recommended that you update your system firmware and perform a
complete system backup before proceeding.  A log of this update
installation can be found at /var/adm/smlogs/update.log.

Do you want to continue the Update Installation?  (y/n) []: y
Do you want to select optional kernel components?  (y/n) [n]: n
Do you want to archive obsolete files?  (y/n) [n]: n
USER SETTINGS:
--------------

*** Checking for conflicting software ***

The following software may require reinstallation after the Update
Installation is completed:

 Advanced Printing Software

Do you want to continue the Update Installation?  (y/n) [y]: Return

*** Determining installed Operating System software ***

 Working....Mon May 21 09:46:40 EST 2001

*** Determining installed Tru64 UNIX TruCluster(TM) Server \
    Software V5.1 (Rev 389) software ***

*** Determining kernel components ***

 Working....Mon May 21 09:48:45 EST 2001

*** Checking for file type conflicts ***

*** Checking for obsolete files ***

*** Checking file system space ***

Update Installation is now ready to begin modifying the files necessary
to reboot the cluster member off of the new OS. Please check the
/var/adm/smlogs/update.log and /var/adm/smlogs/it.log files for errors
after the installation is complete.

Do you want to continue the Update Installation? (y/n) [n]: y

*** Starting configuration merges for Update Install ***
```

## Example C–3: Sample clu_upgrade Log File (cont.)

```
  *** Merging new file ./etc/.new..sysconfigtab into
  existing ./etc/../cluster/members/member2/boot_partition/etc/sysconfigtab

  Merging /etc/../cluster/members/member2/boot_partition/etc/sysconfigtab
  Merge completed successfully.

 The critical files needed for reboot have been moved into place. The
 system will now reboot with the generic kernel for Compaq Computer
 Corporation Tru64 UNIX X5.1A-8 and complete the rolling upgrade for
 this member (member2).

*** END UPDATE INSTALLATION (Mon May 21 09:59:18 EST 2001) ***

The 'roll' stage has completed successfully.  This
member must be rebooted in order to run with the newly installed software.
Do you want to reboot this member at this time? []: y
You indicated that you want to reboot this member at this time.
Is that correct? [yes]: Return

The 'roll' stage of the upgrade has completed successfully.
################################################################################
clu_upgrade Command: clu_upgrade boot
      On Host: polishham.zk3.dec.com Invoked at: Mon May 21 11:04:05 EST 2001
--------------------------------------------------------------------------------
Marking stage 'roll' as 'completed'.
--------------------------------------------------------------------------------
clu_upgrade Command: clu_upgrade boot
       On Host: polishham.zk3.dec.com Exited at: Mon May 21 11:04:09 EST 2001
################################################################################
clu_upgrade Command: clu_upgrade upgrade switch
      On Host: pepicelli.zk3.dec.com Invoked at: Mon May 21 11:19:27 EST 2001
--------------------------------------------------------------------------------

This is the cluster upgrade program.
You have indicated that you want to perform the 'switch' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return
Marking stage 'switch' as 'started'.
Initiating version switch on cluster members
..Successful switch of the version identifiers

Marking stage 'switch' as 'completed'.
The cluster upgrade 'switch' stage has completed successfully.
All cluster members must be rebooted before running the 'clean' command.

The 'switch' stage of the upgrade has completed successfully.
--------------------------------------------------------------------------------
clu_upgrade Command: clu_upgrade upgrade switch
       On Host: pepicelli.zk3.dec.com Exited at: Mon May 21 11:20:02 EST 2001
################################################################################
clu_upgrade Command: clu_upgrade boot
      On Host: polishham.zk3.dec.com Invoked at: Mon May 21 11:28:28 EST 2001
--------------------------------------------------------------------------------
Marking stage 'switch' as 'completed'.
--------------------------------------------------------------------------------
clu_upgrade Command: clu_upgrade boot
       On Host: polishham.zk3.dec.com Exited at: Mon May 21 11:28:31 EST 2001
################################################################################
clu_upgrade Command: clu_upgrade boot
```

## Example C–3: Sample clu_upgrade Log File (cont.)

```
      On Host: pepicelli.zk3.dec.com Invoked at: Mon May 21 11:34:20 EST 2001
--------------------------------------------------------------------------------
Marking stage 'switch' as 'completed'.
--------------------------------------------------------------------------------
clu_upgrade Command: clu_upgrade boot
       On Host: pepicelli.zk3.dec.com Exited at: Mon May 21 11:34:24 EST 2001
################################################################################
clu_upgrade Command: clu_upgrade upgrade clean
      On Host: pepicelli.zk3.dec.com Invoked at: Mon May 21 12:38:54 EST 2001
--------------------------------------------------------------------------------

This is the cluster upgrade program.
You have indicated that you want to perform the 'clean' stage of the
upgrade.

Do you want to continue to upgrade the cluster? [yes]: Return
.Marking stage 'clean' as 'started'.

Deleting tagged files.
.....................................................................\
.....................................................................\
.........................................

Removing back-up and kit files
.................................

The Update Administration Utility is typically run after an update
installation to manage the files that are saved during an update installation.

Do you want to run the Update Administration Utility at this time? [yes]: Return

The Update Installation Cleanup utility is used to clean up backup
files created by Update Installation.  Update Installation can create
two types of files: .PreUPD and .PreMRG.  The .PreUPD files are
copies of unprotected customized system files as they existed prior
to running Update Installation.  The .PreMRG files are copies of
protected system files as they existed prior to running Update
Installation.

Please make a selection from the following menu.

        Update Installation Cleanup Main Menu
        -------------------------------------
        c) Unprotected Customized File Administration (.PreUPD)
        p) Pre-Merge File Administration (.PreMRG)
        x) Exit This Utility

        Enter your choice: x
Exiting /usr/sbin/updadmin...
Marking stage 'clean' as 'completed'.

The 'clean' stage of the upgrade has completed successfully.
--------------------------------------------------------------------------------
clu_upgrade Command: clu_upgrade upgrade clean
       On Host: pepicelli.zk3.dec.com Exited at: Mon May 21 13:38:55 EST 2001
```

# D

# Cluster-Related Messages in System Log Files

The following sections show excerpts from `kern.log` files in the `/var/adm/syslog.dated/`*date* directories:

- Startup messages after running `clu_create` (Section D.1)
- Startup messages after running `clu_add_member` (Section D.2)

These messages track normal cluster startup operations; therefore, in addition to providing some level of assurance that cluster formation and recovery operations are proceeding in an orderly fashion, they also provide a starting point for troubleshooting cluster-related problems.

## D.1 Startup Messages After Running clu_create

Example D–1 shows a transcript of a portion of the startup messages displayed during a reboot of the first cluster member system after running `clu_create`. This information is also written to `/var/adm/syslog.dated/`*date*`/kern.log`.

**Example D–1: Startup Messages After Running clu_create**

```
Alpha boot: available memory from 0x1506000 to 0x7f58000
Compaq Tru64 UNIX X5.1A-8 (Rev. 1747); Tue May  8 15:54:43 EDT 2001
physical memory = 128.00 megabytes.
available memory = 106.32 megabytes.
using 417 buffers containing 3.25 megabytes of memory
Firmware revision: 5.8-10
PALcode: UNIX version 1.79-63
AlphaServer DS20 500 MHz
pci1 (primary bus:1) at nexus
pci2 (primary bus:1 subordinate bus:2) at pci1 slot 7
itpsa0 at pci2 slot 0
IntraServer ROM Version V2.0 (c)1998
scsi0 at itpsa0 slot 0 rad 0
itpsa1 at pci2 slot 1
IntraServer ROM Version V2.0 (c)1998
scsi1 at itpsa1 slot 0 rad 0
tu0: DECchip 21140: Revision: 2.2
tu0: auto negotiation capable device
tu0 at pci2 slot 2
tu0: DEC TULIP (10/100) Ethernet Interface, hardware address: 00-06-2B-00-FC-99
tu0: auto negotiation off: selecting 10BaseT (UTP) port: half duplex
pci0 (primary bus:0) at nexus
isa0 at pci0
gpc0 at isa0
```

## Example D–1: Startup Messages After Running clu_create (cont.)

```
gpc1 not probed
ace0 at isa0
ace1 at isa0
lp0 at isa0
fdi0 at isa0
fd0 at fdi0 unit 0
ata0 at pci0 slot 105 (slot 5, function 1)
ata0: CYPRESS 82C693
scsi2 at ata0 slot 0 rad 0
ata1 at pci0 slot 205 (slot 5, function 2)
ata1: CYPRESS 82C693
scsi3 at ata1 slot 0 rad 0
usb0 at pci0 slot 305 (slot 5, function 3)
comet0:  Card type 'Elsa GLoria' with 8MB framebuffer memory.
comet0 at pci0 slot 7
emx0 at pci0 slot 8
KGPSA-CA : Driver Rev 1.32a : F/W Rev 3.03X2(1.11) : wwn 1000-0000-c924-4b7b
emx0: Using console topology setting of : Fabric
scsi4 at emx0 slot 0 rad 0
mchan0: Module revision = 34
mchan0: jumpered as HUB configuration
mchan0 at pci0 slot 9
Created FRU table binary error log packet
kernel console: comet0
dli: configured
NetRAIN configured.
i2c: Server Management Hardware Present
ATM Subsystem configured with 1 restart threads
ATMUNI: configured
ATMSIG: 3.x (module=uni3x) configured
ILMI: 3.x (module=ilmi) configured
ATM IP: configured
ATM LANE: configured.
ATM IFMP: configured
TruCluster Server X5.1A-8  (Rev. 1126); 05/08/01 04:56
clubase: configured
Configuring RDG to use Memory Channel
ics_hl: Configuring memory channel as transport.
icsnet: configured
drd configured 0
kch: configured
dlm: configured
Starting CFS daemons
Registering CFS Services
Initializing CFSREC ICS Service
Registering CFSMSFS remote syscall interface
Registering CMS Services
TNC kproc_creator_daemon: Initialized and Ready
rm primary: mchan0, hubslot = 4, phys_rail 0 (size 512 MB)
rm primary: log_rail 0 (size 512 MB), phys_rail 0 (mchan0)
ics_mct: icsinfo set for node 1
ics_mct: Declaring this node up 1
CNX MGR: insufficient votes to form cluster: have 1 need 2
CNX QDISK: Adding 1 quorum disk vote toward formation.
CNX MGR: Cluster deli incarnation 0x7c505 has been formed
CNX MGR: Founding node id is 1 csid is 0x10001
CNX MGR: membership configuration index: 1 (1 additions, 0 removals)
CNX MGR: Node pepicelli 1 incarn 0x7c505 csid 0x10001 has been added to the cluster
dlm: resuming lock activity
kch: resuming activity
CNX QDISK: Successfully claimed quorum disk, adding 1 vote.
```

**Example D–1: Startup Messages After Running clu_create (cont.)**

```
CNX MGR: quorum (re)gained, (re)starting cluster operations.
Joining versw kch set.
Waiting for cluster mount to complete
clsm: checking for peer configurations
clsm: initialized
vm_swap_init: swap is set to eager allocation mode
CMS: Joining deferred filesystem sets
cluster alias subsystem enabled
Environmental Monitoring Subsystem Configured.
```

See the *Cluster Administration* manual for descriptions of information and error messages generated by TruCluster Server.

## D.2 Startup Messages After Running clu_add_member

Example D–2 shows a transcript of a portion of the startup messages displayed during the first boot of the second cluster member system after running `clu_add_member`. This information is also written to `/var/adm/syslog.dated/`*`date`*`/kern.log`.

**Example D–2: Startup Messages After Running clu_add_member**

```
Alpha boot: available memory from 0x179a000 to 0xff58000
Compaq Tru64 UNIX X5.1A-8 (Rev. 1747); Tue May  8 16:24:12 EDT 2001
physical memory = 256.00 megabytes.
available memory = 231.74 megabytes.
using 909 buffers containing 7.10 megabytes of memory
Firmware revision: 5.8-10
PALcode: UNIX version 1.79-63
AlphaServer DS20 500 MHz
pci1 (primary bus:1) at nexus
pci2 (primary bus:1 subordinate bus:2) at pci1 slot 7
itpsa0 at pci2 slot 0
IntraServer ROM Version V2.0 (c)1998
scsi0 at itpsa0 slot 0 rad 0
itpsa1 at pci2 slot 1
IntraServer ROM Version V2.0 (c)1998
scsi1 at itpsa1 slot 0 rad 0
tu0: DECchip 21140: Revision: 2.2
tu0: auto negotiation capable device
tu0 at pci2 slot 2
tu0: DEC TULIP (10/100) Ethernet Interface, hardware address: 00-06-2B-01-1F-5A
tu0: auto negotiation off: selecting 10BaseT (UTP) port: half duplex
mchan0: Module revision = 34
mchan0: jumpered as HUB configuration
mchan0 at pci1 slot 8
pci0 (primary bus:0) at nexus
isa0 at pci0
gpc0 at isa0
gpc1 not probed
ace0 at isa0
ace1 at isa0
lp0 at isa0
fdi0 at isa0
fd0 at fdi0 unit 0
ata0 at pci0 slot 105 (slot 5, function 1)
ata0: CYPRESS 82C693
scsi2 at ata0 slot 0 rad 0
ata1 at pci0 slot 205 (slot 5, function 2)
ata1: CYPRESS 82C693
```

## Example D–2: Startup Messages After Running clu_add_member (cont.)

```
scsi3 at ata1 slot 0 rad 0
usb0 at pci0 slot 305 (slot 5, function 3)
comet0:  Card type 'Elsa GLoria' with 8MB framebuffer memory.
comet0 at pci0 slot 7
emx0 at pci0 slot 9
KGPSA-CA : Driver Rev 1.32a : F/W Rev 3.02A1(1.11) : wwn 1000-0000-c922-4aac
emx0: Using console topology setting of : Fabric
scsi4 at emx0 slot 0 rad 0
Created FRU table binary error log packet
kernel console: comet0
dli: configured
NetRAIN configured.
i2c: Server Management Hardware Present
ATM Subsystem configured with 1 restart threads
ATMUNI: configured
ATMSIG: 3.x (module=uni3x) configured
ILMI: 3.x (module=ilmi) configured
ATM IP: configured
ATM LANE: configured.
ATM IFMP: configured
TruCluster Server X5.1A-8  (Rev. 1126); 05/08/01 04:56
clubase: configured
Configuring RDG to use Memory Channel
ics_hl: Configuring memory channel as transport.
icsnet: configured
drd configured 0
kch: configured
dlm: configured
Starting CFS daemons
Registering CFS Services
Initializing CFSREC ICS Service
Registering CFSMSFS remote syscall interface
Registering CMS Services
TNC kproc_creator_daemon: Initialized and Ready
rm slave: mchan0, hubslot = 2, phys_rail 0 (size 512 MB)
rm slave: log_rail 0 (size 512 MB), phys_rail 0 (mchan0)
ics_mct: icsinfo set for node 2
ics_mct: Declaring this node up 2
ics_mct: icsinfo set for node 1
CNX MGR: Join operation complete
CNX MGR: membership configuration index: 4 (3 additions, 1 removals)
CNX MGR: quorum (re)gained, (re)starting cluster operations.
CNX MGR: Node polishham 2 incarn 0x17396 csid 0x20002 has been added to the cluster
Joining versw kch set.
ics_mct: Declaring this node up 1
CNX MGR: Node pepicelli 1 incarn 0x7c505 csid 0x10001 has been added to the cluster
kch: resuming activity
dlm: resuming lock activity
clsm: incoming CNX data: '…a^D'
clsm: checking for peer configurations
clsm: initialized
Waiting for cluster mount to complete
vm_swap_init: swap is set to eager allocation mode
CMS: Joining deferred filesystem sets
CNX QDISK: Successfully claimed quorum disk, adding 1 vote.
cluster alias subsystem enabled
Environmental Monitoring Subsystem Configured.
Can't find an OSF-BASE, UNIX-WORKSTATION, or UNIX-SERVER license PAK
```

# E

# Manual Storage Configuration When Upgrading

This appendix contains information for sites that are upgrading from TruCluster Software Version 1.5 or Version 1.6 to TruCluster Server Version 5.1A but that either cannot or choose not to use the Option 2 or Option 3 storage mapping and configuration scripts described in Chapter 8.

In general and where possible, we recommend that you use the procedures and scripts described in Chapter 8. However, if your storage topology, system configurations, or site policy make it impossible to do so, this appendix describes how to manually gather storage configuration information in the Available Server Environment (ASE) and how to manually configure storage on the new Tru64 UNIX system or single-member cluster. You are responsible for mapping old-style (`rz*`) device names to new-style (`dsk*`) device names.

You must create your own upgrade procedure. Read all of Chapter 8 and this appendix, decide which upgrade option is a reasonable starting point for your upgrade, and then modify that option's procedure.

## E.1 Manually Gathering Device and Storage Configuration Information

This section replaces the steps in Option 2 and Option 3 that use the `clu_migrate_save` script to gather information about the current ASE configuration and storage environment.

First, read Section 8.3.2 and create an up-to-date configuration map for the current ASE.

In Chapter 8, the `clu_migrate_save` script captures the current shared storage configuration including LSM and AdvFS configuration information. The `clu_migrate_configure` script reads the gathered information and configures storage on the new Tru64 UNIX system or single-member cluster. However, if you plan to manually configure storage on the new cluster after connecting the physical devices, you will not run `clu_migrate_configure` and therefore need to manually gather storage configuration information for the members of the ASE.

**Note**

In addition to manually gathering information, we recommend running `clu_migrate_save` on the current ASE members. Other than gathering storage configuration information, `clu_migrate_save` writes each device's special file name (`rz*`) to the `label:` field of that device's disk label. (The script also saves the the original device label as described in Chapter 8; you can restore the original labels after upgrading.)

We also recommend running `clu_migrate_configure -x` on the new Tru64 UNIX system or TruCluster Server cluster. The `clu_migrate_configure -x` command does not configure storage; it lists the commands it would run if invoked without the `-x` option and displays a mapping of old-style `rz*` device names to new-style (`dsk*`) device names. However, you must have run `clu_migrate_save` on the ASE members in order for `clu_migrate_configure -x` to provide this mapping.

If you do not use the scripts to help map old-style device names to new-style device names, exercise extreme care when manually mapping device names. You must know the physical location of each device, and be able to use this knowledge, and utilities like `hwmgr` and `scu`, to determine which `dsk*` name is assigned to each device during the upgrade. Determining which `dsk*` the new system or cluster assigns to a device previously known as `rz*` is not a trivial task; it is the primary reason for providing the scripts. Although you do not have to use the migration scripts to configure storage on the new Tru64 UNIX system or TruCluster Server cluster, they are highly recommended for making sure that you know which `rz*` device is now known as `dsk*`.

On each member of the ASE:

- Run the `sys_check -all` command (Version 114 or higher) to save system information and create a storage map. For example:

  ```
  # /usr/sbin/sys_check -all > file.html
  ```

  **Note**

  If you do not have the `sys_check` utility on the system, you can get it from the following URL:

  `ftp://ftp.digital.com/pub/DEC/IAS/sys_check`

- Save both the `/var/ase/config/asecdb` database and a text copy of the database.

For example:

```
# cp /var/ase/config/asecdb asecdb.copy
# asemgr -d -C > asecdb.txt
```

- Save information about AdvFS domains and file sets. For example, change directory to `/etc/fdmns` and capture the output of the following commands:
  - Use the `ls -lR` command to list all domains and associated devices.
  - Use the `showfdmn *` command to display information about file domains and volumes.
  - Use the `showfsets` command to display fileset information for each domain.
- If you are using LSM, run the `volsave` command to save the LSM configuration information for all disk groups. (All ASE services must be on line before you run the `volsave` command.)

  ```
  # volsave -d volsave.output
  ```

- If you are not using the `clu_migrate_save` and `clu_migrate_configure -x` commands to map device names, you must manually map old-style (`rz*`) device names to their new-style (`dsk*`) counterparts in order to configure storage on the new system. To help with the mapping, use the `scu` command to create a list of the old-style device names and their attributes. Some suggested attributes are: vendor, serial number, and bus/target/LUN (not applicable if `ase_fix_config` was used to renumber SCSI buses). Enter the following `scu` commands to display these attributes, and save the output to files:

  ```
  # scu -f device show device
  # scu -f device show inq page serial
  # scu -f device show nexus
  ```

  For example:

  ```
  # scu -f /dev/rrz28g show device | grep -E "Vendor|Product|Firmware"
            Vendor Identification: DEC
           Product Identification: RZ26L    (C) DEC
          Firmware Revision Level: 440C

  # scu -f /dev/rrz28g show inq page serial | grep "Product Serial"
  Product Serial Number: PCB=420240831056(ZG40831056 ?); \
  HDA=0000000042181869

  # scu -f /dev/rrz28g show nexus
  Device: RZ26L, Bus: 3, Target: 4, Lun: 0, Type: Direct Access
  ```

- Copy the files that contain saved configuration information to the new Tru64 UNIX system or single-member cluster.

## E.2 Manually Configuring Storage on the New Tru64 UNIX System or TruCluster Server Cluster

This section replaces the steps in Option 2 and Option 3 that use the `clu_migrate_configure` script to configure storage on the new Tru64 UNIX system or single-member cluster.

**Note**

If you did not run `clu_migrate_save` on the ASE members, you cannot use `clu_migrate_configure -x` to display device-name mappings. Before continuing, perform a manual mapping of all old-style `rz*` device names to the new-style `dsk*` device names. In the following procedure, substitute the results of your own mapping for those provided by `clu_migrate_configure -x`.

If you ran `clu_migrate_save` on the ASE members, on the Tru64 UNIX system or the single-member cluster, run `clu_migrate_configure -x`:

```
# /usr/opt/TruCluster/tools/migrate/clu_migrate_configure -x
```

The `clu_migrate_configure -x` command displays a mapping of old-style device names to new-style device names. Use this information when configuring the storage that was controlled by the ASE.

The following steps provide some guidance when configuring storage:

1. If you are not using the `clu_migrate_save` and `clu_migrate_configure -x` commands to map device names, manually map old-style (`rz*`) device names to new-style (`dsk`) device names. Use the `scu` command on the new system to help map devices to their new device names. For example:

```
# scu -f /dev/rdisk/dsk5g show device | grep -E "Vendor|Product|Firmware"
          Vendor Identification: DEC
         Product Identification: RZ26L    (C) DEC
        Firmware Revision Level: 440C

# scu -f /dev/rdisk/dsk5g show inq page serial | grep "Product Serial"
Product Serial Number: PCB=420240831056(ZG40831056 ?); \
HDA=0000000042181869

# scu -f /dev/rdisk/dsk5g show nexus
Device: RZ26L, Bus: 1, Target: 4, Lun: 0, Type: Direct Access
```

   Using the `scu` information that you collected from the ASE members, create a map of old-style device names to new-style device names. Note that the `hwmgr` command is also a useful tool when manually mapping device names.

   If you look at the `scu` examples in Section E.1, you will see that this device was known as `rz28` in the ASE. Note that the bus numbers in

the `show nexus` output are not the same. Because `ase_fix_config` was run in the ASE, bus numbers are not be the same on both systems, and are not a reliable piece of information for mapping devices.

2. Using the LSM information from the saved `asecdb` database, the output from `sys_check -all`, and the device mapping table that you created manually or with `clu_migrate_configure -x`, configure each device and LSM disk group.

**Note**

The LSM configuration information on disk has changed format. If you need to revert to the ASE from this point on, you will need to retore the LSM information when you import on an ASE system.

For every new LSM device, enter:

```
# voldisk define device
# voldisk online device
```

For every disk group, enter:

```
# voldg import disk_group
```

For every volume, enter:

```
# volume -g disk_group start volume
```

For example:

```
# voldisk define dsk5g
# voldisk online dsk5g
# voldg import toolsdg
lsm:voldg: WARNING: Volume vol01: \
Temporarily renumbered due to conflict
# volume -g toolsdg start vol01
# volume -g toolsdg start vol02
```

You can ignore the warnings.

3. To prepare LSM to update names following the next reboot, enter the `lsmupdate_setup` command:

```
# /sbin/lsm.d/bin/lsmupdate_setup
```

4. Using the AdvFS information from the saved `asecdb` database, the output from `sys_check -all`, and the device mapping table that you manually created or with `clu_migrate_configure -x`, manually re-create the AdvFS domains that were on the ASE, mapping the old-style `rz` device names to new-style `dsk` device names and creating the appropriate `/etc/fdmns` entries.

For example:

```
# mkdir /etc/fdmns/data1_domain
# cd /etc/fdmns/data1_domain
# ln -s /dev/disk/dsk6g

# mkdir /etc/fdmns/tools_dmn
# cd /etc/fdmns/tools_dmn
# ln -s /dev/vol/toolsdg/vol01 toolsdg.vol01
```

For each domain, enter the `showfsets` *domain* command and verify that the filesets are correct for the domain.

5. Mount the domains. For each domain, run the `showfdmn` *domain* command.
6. Add file systems to the `/etc/fstab` file. Also update any other configuration files that contain storage information, such as `/etc/exports`.
7. Reboot the system to configure LSM with the new device names:

```
# shutdown -r now
```

8. Enter the following LSM commands to examine the LSM configuration:

```
# voldisk list
# volprint -thA
```

# Index

## A



## B



## C

## D

## E

## F



## G



## H



## I



## K



## L

## M



## N

## O



## P



## Q



## R

## S



## T

## U



## V



## W



## X