Oracle® Solaris 10 1/13 Installation Guide: Flash Archives (Creation and Installation)

ORACLE®

Contents

Preface

This book provides planning information and instructions for creating flash archives and using Flash Archive to install the Oracle Solaris OS on multiple systems.

This book does not include instructions about how to set up system hardware or other peripherals. This book is only for installing UFS file systems. Flash Archive cannot be used for Oracle Solaris ZFS installations.

Note – This Oracle Solaris release supports systems that use the SPARC and x86 families of processor architectures. The supported systems appear in the *Oracle Solaris OS: Hardware Compatibility Lists*. This document cites any implementation differences between the platform types.

In this document, these x86 related terms mean the following:

- x86 refers to the larger family of 64-bit and 32-bit x86 compatible products.
- x64 relates specifically to 64-bit x86 compatible CPUs.
- "32-bit x86" points out specific 32-bit information about x86 based systems.

For supported systems, see the *Oracle Solaris OS: Hardware Compatibility Lists*.

Who Should Use This Book

This book is intended for system administrators who are responsible for installing the Oracle Solaris OS. These procedures are advanced Oracle Solaris installation information for enterprise system administrators who manage multiple Oracle Solaris machines in a networked environment.

Related Books

The following table lists documentation for system administrators.

Description	Information
Do you need system requirements or high-level planning information? Or want a high-level overview of Oracle Solaris ZFS installations, GRUB, a feature of Oracle Solaris, based booting, Oracle Solaris Zones partitioning technology, or creating RAID-1 volumes?	*Oracle Solaris 10 1/13 Installation Guide: Planning for Installation and Upgrade*
Do you need to install a single system from DVD or CD media? The Oracle Solaris installation program steps you through an installation.	*Oracle Solaris 10 1/13 Installation Guide: Basic Installations*
Do you need to upgrade or patch your system with almost no downtime? Save system downtime when upgrading by using Live Upgrade, a feature of Oracle Solaris.	*Oracle Solaris 10 1/13 Installation Guide: Live Upgrade and Upgrade Planning*
Do you need to install a secure installation over the network or Internet? Use WAN boot to install a remote client. Or, do you need to install over the network from a network installation image? The Oracle Solaris installation program steps you through an installation.	*Oracle Solaris 10 1/13 Installation Guide: Network-Based Installations*
Do you need to install Oracle Solaris on multiple machines? Use JumpStart to automate your installation.	*Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*
Do you need to back up your system?	Chapter 19, "Backing Up and Restoring UFS File Systems (Overview/Tasks)," in *System Administration Guide: Devices and File Systems*
Do you need troubleshooting information, a list of known problems, or a list of patches for this release?	*Oracle Solaris Release Notes*
Do you need to verify that your system works on Oracle Solaris?	SPARC: *Oracle Solaris Sun Hardware Platform Guide*
Do you need to check on which packages have been added, removed, or changed in this release?	*Oracle Solaris Package List*
Do you need to verify that your system and devices work with Oracle Solaris SPARC and x86 based systems and other third-party vendors?	Solaris Hardware Compatibility List for x86 Platforms

Description	Information
Do you want to learn more about installing a ZFS root pool?	Chapter 5, "ZFS Root File System Installation Planning," in *Oracle Solaris 10 1/13 Installation Guide: Planning for Installation and Upgrade*

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P–1   Typographic Conventions

Typeface	Description	Example
AaBbCc123	The names of commands, files, and directories, and onscreen computer output	Edit your .login file.
		Use ls -a to list all files.
		machine_name% you have mail.
AaBbCc123	What you type, contrasted with onscreen computer output	machine_name% **su**
		Password:
aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is rm *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*.
		A *cache* is a copy that is stored locally.
		Do *not* save the file.
		Note: Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows UNIX system prompts and superuser prompts for shells that are included in the Oracle Solaris OS. In command examples, the shell prompt indicates whether the command should be executed by a regular user or a user with privileges.

TABLE P–2 Shell Prompts

Shell	Prompt
Bash shell, Korn shell, and Bourne shell	`$`
Bash shell, Korn shell, and Bourne shell for superuser	`#`
C shell	`machine_name%`
C shell for superuser	`machine_name#`

◆ ◆ ◆ **CHAPTER 1**

1

Flash Archive Overview

This book provides instructions for creating flash archives and using Flash Archive, a feature of Oracle Solaris, to install the Oracle Solaris OS on multiple systems.

For a description of limitations when creating or installing a flash archive, see Table 2–1.

Note – If you want an overview of all Oracle Solaris installation methods, see "Choosing an Oracle Solaris Installation Method" in *Oracle Solaris 10 1/13 Installation Guide: Planning for Installation and Upgrade*.

Flash Archive Introduction

The Flash Archive installation feature enables you to use a single reference installation of the Oracle Solaris OS on a system, which is called the master system. Then, you can replicate that installation on a number of systems, which are called clone systems. You can replicate clone systems with a Flash Archive initial installation that overwrites all files on the system or with a Flash Archive update that only includes the differences between two system images. A differential update changes only the files that are specified and is restricted to systems that contain software consistent with the old master image.

What's New in the Oracle Solaris 10 08/11 Release

Starting with the Oracle Solaris 10 08/11 release, the ZFS file system has the following installation enhancements:

- You can use the `luupgrade` command of Live Upgrade, a feature of Oracle Solaris, to install a ZFS root flash archive to a ZFS rooted alternate boot environment.
- You can use the interactive text-mode installation method to install a system with a ZFS flash archive.

- You can use the `-D` option of the Live Upgrade `lucreate` command to create a separate dataset for `/var` when you migrate a UFS root file system to a ZFS root file system.

For detailed instructions and limitations, see Chapter 4, "Installing and Booting an Oracle Solaris ZFS Root File System," in *Oracle Solaris ZFS Administration Guide*.

Unlike the ZFS Flash Archive installation in the previous releases, a flash archive created on a ZFS root master system does not contain all the existing boot environments. Instead, the archive only contains the active ZFS boot environment. The archive does not include those datasets that are excluded explicitly with the `-D` option of the `lucreate` command and the user data present in the top-level pool dataset. The swap and dump volumes are not included into the archive but are created when the flash archive is installed.

For more information on ZFS Flash Archive creation and installation , refer to Chapter 4, "Installing and Booting an Oracle Solaris ZFS Root File System," in *Oracle Solaris ZFS Administration Guide*.

Auto Registration and Flash Archives

Auto Registration, a feature of Oracle Solaris, is new in the Oracle Solaris 10 9/10 release. When you install or upgrade your system, configuration data about your system is, on rebooting, automatically communicated through the existing service tag technology to the Oracle Product Registration System.

If you create a flash archive based on a master system that was installed with a release prior to the Oracle Solaris 10 9/10 release, that archive does **not** include Auto Registration. Auto Registration has no impact on your work with that archive.

If you create a flash archive based on a master system that was installed with the Oracle Solaris 10 9/10 release or a later release, that archive does include Auto Registration unless it was specifically disabled on the master system.

How Does Auto Registration Affect Flash Archives?

For any archive based on the Oracle Solaris 10 9/10 release or a later release, Auto Registration is enabled by default unless it was specifically disabled on the master system. When you install the flash archive or upgrade a clone system with the differential flash archive, configuration data about that installed or upgraded system is, on rebooting, automatically communicated through the existing service tag technology to the Oracle Product Registration System.

Auto Registration uses support credentials and proxy information that you provide before or during an installation or upgrade. The means of providing that credential and proxy information depends on which installation method is used, as shown in the following table.

TABLE 1–1   Auto Registration Impact

Installation Method	Auto Registration Impact
Interactive installation	During the installation of a flash archive, you are prompted in the installer screens to provide your support credentials and, if needed, proxy information. After the installation, the system is registered on reboot. If you do not provide support credentials, an anonymous registration occurs on reboot.
JumpStart, a feature of Oracle Solaris	You can provide your support credentials and proxy information by using the auto_reg keyword in the sysidcfg file prior to the installation of an archive or prior to an upgrade with a differential flash archive. If you do not use this keyword, you are prompted to provide this information during the installation of the archive or during the upgrade. After the installation or upgrade, the system is registered on reboot. If you do not provide that information, an anonymous registration occurs on reboot.
Live Upgrade	The flash archive uses the same Auto Registration settings, including support credentials and proxy information, that were specified on the master system. As long as Auto Registration was not disabled on the master system, the archive system is, after the upgrade, automatically registered on reboot.
Network installations, including WAN boot installations	You can provide your support credentials and proxy information by using the auto_reg keyword in the sysidcfg file prior to the network installation of a flash archive. If you do not use this keyword, you are prompted during the network installation to provide this information. The archive is registered when the system reboots after the installation. If you do not provide that information, an anonymous registration occurs on reboot.

Flash Archives and ZFS Root Pools

Starting with the Solaris 10 10/09 release, you can set up a JumpStart profile to identify a flash archive of a ZFS root pool.

A flash archive can be created on a system that is running a UFS root file system or a ZFS root file system. A flash archive of a ZFS root pool contains the entire pool hierarchy except for the swap and dump volumes, and any excluded datasets. The swap and dump volumes are created when the flash archive is installed.

You can use the Flash Archive installation method as follows:

1. Generate a flash archive that can be used to install and boot a system with a ZFS root file system.

2. Perform a JumpStart installation of a system by using a ZFS Flash archive.

> **Note** – Creating a ZFS flash archive backs up an entire root pool, not individual boot environments. Individual datasets within the pool can be excluded by using the `flarcreate` and `flar` command's `-D` option.

For detailed instructions and limitations, see "Installing a ZFS Root File System (Oracle Solaris Flash Archive Installation)" in *Oracle Solaris ZFS Administration Guide*.

Installing Clone Systems With an Initial Installation

You can install a master system with a flash archive for an initial installation by using any installation method: Oracle Solaris installation program, JumpStart, Live Upgrade, or WAN boot. All files are overwritten. The Flash Archive installation is a five-part process.

1. Install the master system.

    You select a system and use any of the Oracle Solaris installation methods to install the Oracle Solaris OS and any other software.

2. (Optional) Prepare customization scripts to reconfigure or customize the clone system before or after installation.

3. Create the flash archive.

    The flash archive contains a copy of all of the files on the master system, unless you excluded some nonessential files.

4. Install the flash archive on clone systems.

    The master system and the clone system must have the same kernel architecture. For further information, see "How to Install a Sun4U Flash Archive on a Sun 4V Machine" on page 21.

    When you install the flash archive on a system, all of the files in the archive are copied to that system. The newly installed system now has the same installation configuration as the original master system, which is why it is called a *clone system*. Some customization is possible:

    - Scripts can be used for customization.
    - You can install extra packages with a flash archive by using the JumpStart installation method. The packages must be from outside the software group being installed or a third-party package.

5. (Optional) Save a copy of the master image.

    If you plan to create a differential archive, the master image must be available and identical to the image installed on the clone systems.

For step-by-step instructions, see "Installing the Master System" on page 32.

The following figure shows an installation of clone systems with an initial installation. All files are overwritten.

FIGURE 1–1 Solaris Flash Initial Installation
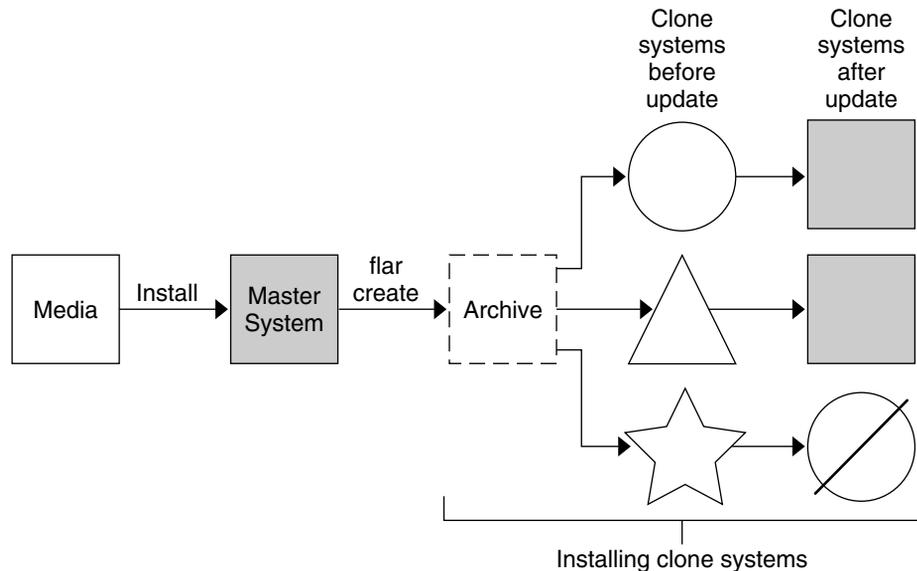
A system running any operating environment

A system with no operating environment

A system with a different architecture

Update fails

Updating Clone Systems With a Flash Archive Differential Archive

If you have a clone system and want to update that system, you can create a differential archive that contains only the differences between two images, the unchanged master image and an updated master image. When you update a clone system with a differential archive, only the files that are in the differential archive are changed. You can choose to install a Flash Archive differential archive with the JumpStart installation method or Live Upgrade. An update is a five-part process.

1. Prepare the master system with changes.

Before changes are made, the master system should be running a duplicate of the original archive.

Note – If the master system is not running a duplicate of the original archive, the differences between the two system images might result in a large differential archive. Consequently, installing the differential archive could be time consuming. Use an initial installation with a full archive in this case.

---

2. (Optional) Prepare customization scripts to reconfigure or customize the clone system before or after installation.

3. Mount the directory of a copy of the saved-unchanged master image.

    This second image is to be used to compare the two system images. Access the image by one of the following methods:

    -   Mounted from a Live Upgrade boot environment
    -   Mounted from a clone system over NFS
    -   Restored from backup by using the `ufsrestore` command

4. Create the differential archive with the `-A` option of the `flarcreate` command.

5. Install the differential archive on clone systems with JumpStart or use Live Upgrade to install the differential archive on an inactive boot environment.

The following figure shows the creation and installation of a differential archive. A master image is updated with some modifications. These modifications could be as simple as the addition, reconfiguration, or deletion of a few files, or as complex as propagating patches. The updated master image is compared to the unchanged master image. The differences between the two images become the differential archive. The archive can be used to update other clone systems that are currently using the unchanged master image. If the clone system has already been modified or is not running the unchanged master image, the update fails. If you have many changes to make on the clone systems, you can do an initial installation at any time.

FIGURE 1–2   Solaris Flash Update
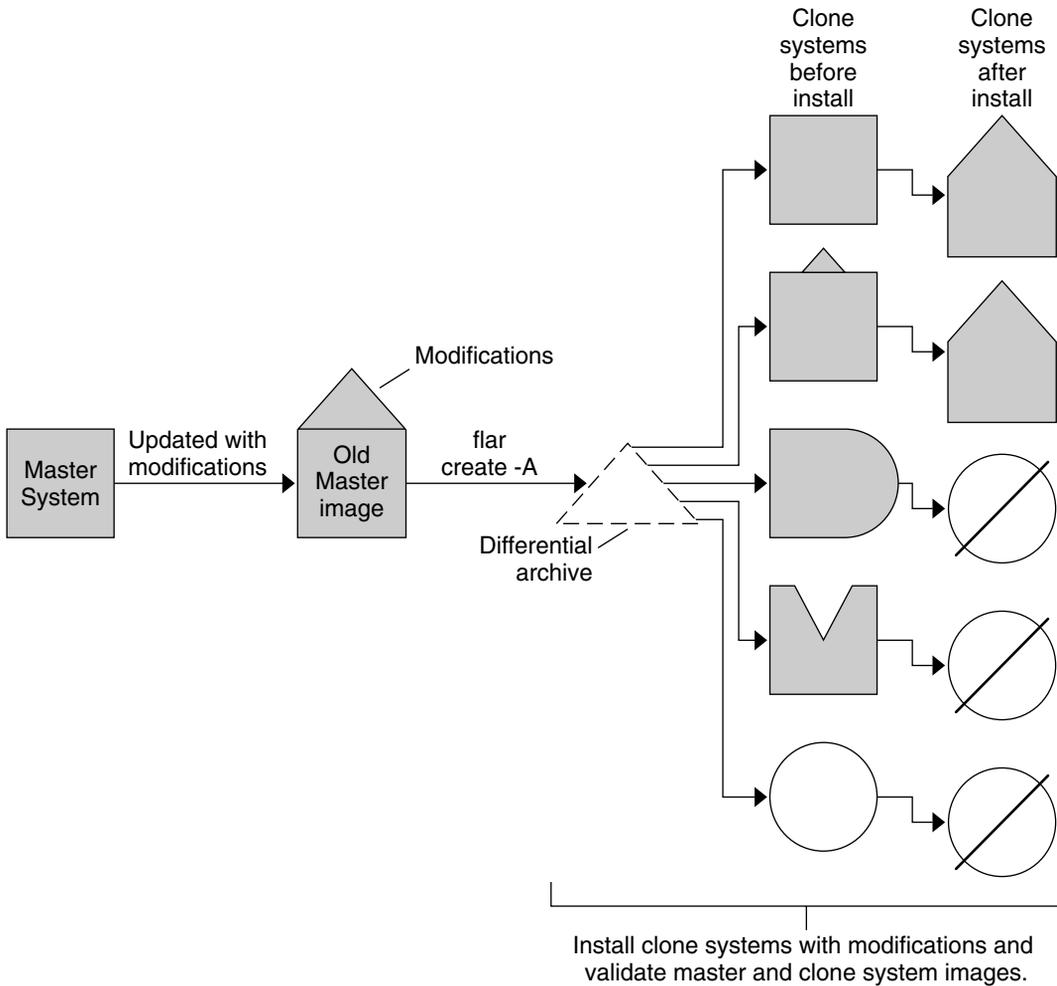
Exact duplicate of master

Duplicate of master but contains small changes

Duplicate of master but contains extra files

Duplicate of master but missing some files

Created from different master or installed separately

Update fails

2

Flash Archive Planning

This chapter provides information necessary for planning a Flash Archive installation in your environment.

Planning Your Flash Archive Installation

Before you create and install a flash archive, you must make some decisions about how you want to install the Oracle Solaris OS on your systems. The first time that you install a system, you install with a full archive that is an initial installation. After a system has been installed with an archive, the system can be updated with a differential archive. The differential archive installs only the differences between two archives.

Note – Starting with the Solaris 10 10/09 release, you can set up a JumpStart profile to identify a flash archive of a ZFS root pool.

Review the following limitations before creating and installing a flash archive.

TABLE 2–1   Limitations When Creating and Installing a Flash Archive

Limitation	Description
When installing the Oracle Solaris OS with a flash archive, the archive and the installation media must contain identical operating system versions.	For example, if the archive is an Oracle Solaris 10 OS and you are using DVD media, then you must use Oracle Solaris 10 DVD media to install the archive. If the operating systems versions do not match, the installation on the target system fails.

TABLE 2–1   Limitations When Creating and Installing a Flash Archive   *(Continued)*

Limitation	Description
A flash archive cannot be properly created when a non-global zone is installed.	The Flash Archive feature is not compatible with the Oracle Solaris Zones partitioning technology. If you create a flash archive, the resulting archive is not installed properly when the archive is deployed under these conditions: ■ The archive is created in a non-global zone ■ The archive is created in a global zone that has non-global zones installed
Starting with the Solaris 10 10/08 release, if you are installing a flash archive from a release before the Solaris 10 10/08 release, the archive fails to install.	If you need to install a flash archive from a previous release, boot from the previous release and install the archive.

Designing an Initial Installation of the Master System

The first task in the Flash Archive installation process is to install a system, the master system, with the configuration that you want each of the clone systems to have. You can use any of the Oracle Solaris installation methods to install an archive on the master system. The installation can be a subset or a complete installation of the Oracle Solaris OS. After you complete the installation, you can add or remove software or modify any configuration files. Some limitations to installing the master system are as follows:

■ The master system and the clone systems must have the same kernel architectures. For example, you can only use an archive that was created from a master system that has a Sun4U architecture to install clones with a Sun4U architecture. For sample instructions, see "How to Install a Sun4U Flash Archive on a Sun 4V Machine" on page 21.

■ You must install the master system with the exact configuration that you want on each of the clone systems. The decisions that you make when you design the installation of the master system depend on the following items:

  ■ The software that you want to install on the clone systems

  ■ Peripheral devices that are connected to the master system and the clone systems

  ■ The architecture of the master system and the clone systems

Note – If you already have installed clone systems and want to update these systems with a new configuration, see "Planning to Create the Flash Archive Differential Archive for an Update" on page 26.

▼ How to Install a Sun4U Flash Archive on a Sun 4V Machine

This procedure is limited to the following types of installations:

- Simple installations only and not complex installations such as the following:
  - Installations with zones
  - Installations with attached storage
  - Installations with fibre attached or with SAN in use
- For installing through the primary interface only. See CR 6772769.
- Installations that do not involve volume-managed root (encapsulated).
- For a UFS root system only. A flash archive installation of a ZFS root system uses a different installation mechanism.

Note – In order for a flash archive to be installed on different system types, the Entire PlusOEM distribution needs to be installed on the master system. Therefore, you must start with a Sun4U machine that has been installed with the Entire PlusOEMSoftware Group, so that all the driver packages are in the image, even if these packages are not in use. For further information about this requirement, see "SPARC: Supporting Peripheral Devices Not Found on the Master System" on page 23

---

1 Create a flash archive on a Sun4U machine so that the archive can be installed on a Sun4V machine.

Use one of the following options to add Sun4V as a supported architecture for the archive.

- **Modify the `/var/sadm/system/admin/.platform` file.**

  a. Add the following line to the `/var/sadm/system/admin/.platform` file:

    ```
    PLATFORM_GROUP=sun4v
    ```

  b. (Optional) Verify that the Sun4V platform group is supported by using the following command:

    ```
    # flar info archive-dir | grep content_architectures
    ```

    This command should display the following results:

    ```
    content_architectures=sun4u,sun4v
    ```

  c. Create the flash archive.

    For more information about creating flash archives, see "Creating a Flash Archive" on page 37.

- **Use the -Uoption of the `flarcreate` command during the flash archive creation process to add Sun4V as a supported architecture for the archive.**

  a. **Issue the `flarcreate` command with the -Uoption.**

     ```
     # flarcreate -n archive-name -U "content_architectures=sun4u,sun4v"
     \-c -x /archive-dir /archive-dir/archive-name.flar
     ```

     /*archive-dir* is the location for the archive.

     For example:

     ```
     # flarcreate -n S10U5hybrid -U "content_architectures=sun4u,sun4v"
     \-c -x /data /data/S10U5hybrid.flar
     ```

  b. **(Optional) Verify that the Sun4V platform group is supported by using the following command:**

     ```
     # flar info archive-dir | grep content_architectures
     ```

     This command should display the following results:

     ```
     content_architectures=sun4u,sun4v
     ```

2   Install the Flash Archive on the Sun4V machine.

You can use JumpStart and a net image to deploy the flash archive.

At this point, the Sun4V machine might not boot. Do not try to patch the machine at this stage. If the machine is allowed to reboot after using JumpStart, you will probably see a message such as:

```
Boot device: /pci@780/pci@0/pci@9/scsi@0/disk@0,0:a  File and args:
Boot load failed.
The file just loaded does not appear to be executable.
```

3   Upgrade the Sun4V machine using either a network image or a DVD image.

For example, you could use a Solaris 10 Update 6 JumpStart image and boot the Sun4V image from that network image, selecting the upgrade option.

In this example, the upgrade completes with the following issues:

- Where both .u and .v versions of a package were available, both versions will be installed. See CR 6846077.

- The /var/sadm/system/admin/.platform file contains incorrect information. See CR 6523030.

- Any third party .v packages are not part of the Oracle Solaris image. So, third party packages will probably not be upgraded.

4   Boot the Sun4V machine.

You can now apply patches to the machine as needed.

Customizing the Oracle Solaris Installation on the Master System

After you install the Oracle Solaris OS on the master system by using any of the Oracle Solaris installation methods, you can add or delete software and modify system configuration information as necessary. To customize the master system's software, you can do the following:

- You can remove software that you determine is not necessary to install on the clone systems. To see a list of software that is installed on the master system, use the Product Registry. For detailed instructions, refer to *Oracle Solaris Administration: Basic Administration*.

- You can install software that is included in the Oracle Solaris release or software that is not delivered as part of the Oracle Solaris OS. All of the software that you install on the master system is included in the flash archive and is installed on the clone systems.

- You can alter configuration files on the master system. For example, you can modify the /etc/inet/inetd.conf file to restrict the daemons that the system runs. All of the modifications that you make are saved as part of the flash archive and are installed on the clone systems.

For an overview of other customizations, see "Customizing an Archive's Files and Directories" on page 27.

Creating Archives for SPARC and x86 Systems

If you want to install Oracle Solaris software by using a flash archive on both SPARC and x86 systems, you must create a separate flash archive for each platform. Use the flash archive that was created from the SPARC master system to install SPARC systems. Use the flash archive that was created from the x86 master system to install x86 systems.

SPARC: Supporting Peripheral Devices Not Found on the Master System

When choosing the drivers to install on the master system, note the following dependencies:

- The type of peripheral devices attached to both the master system and the clone system
- The type of software group installed

The Entire Plus OEM Software Group installs all drivers regardless of the hardware that is present on the system. Other software groups provide limited support. If you install another software group and the clone systems have different peripheral devices than the master system, you need to install the appropriate drivers on the master system before you create the archive.

Note – In order for a flash archive to be installed on different system types, the Entire Plus OEM distribution needs to be installed on the master system.

You can install support for peripherals on clone systems that are different from the master system in by installing the Entire Plus OEM Software Group or installing selected packages.

Type of Installation	Description
Install the Entire Plus OEM Software Group	The Entire Plus OEM Software Group is the largest software group available. This group contains every package that is found in the Oracle Solaris OS. The Entire Plus OEM Software Group installs all drivers regardless of the hardware that is present on the system. A flash archive that is created with the Entire Plus OEM Software Group works on any clone system that has peripheral devices supported by the installed release of the Oracle Solaris OS.
	Note – In order for a flash archive to be installed on different system types, the Entire Plus OEM distribution needs to be installed on the master system.
	Installing master systems with the Entire Plus OEM Software Group guarantees compatibility with other peripheral configurations. However, the Entire Plus OEM software sroup requires at least 2.9 GB of disk space. The clone systems might not have the space that is required to install the Entire Plus OEM software group.
Install other software groups	If you install the master system with the following software groups, you are limiting the support for peripherals. The master system supports only the peripheral devices that are attached to the master system at the time of installation. ■ Reduced Networking software group ■ Core software group ■ End User software group ■ Developer software group ■ Entire software group
	Installing these software groups could result in your clone systems failing to have all the drivers needed. For example, if you install the Entire software group on a master system that has a GX CG6 frame buffer, only the GX CG6 frame buffer driver is installed. This situation is not a problem if all the clone systems that you want to install have the GX CG6 frame buffer or no frame buffer.
Install selected packages	When you install the master system, you can install only the packages that you need for the master system and the clone systems. By selecting specific packages, you can install only support for the peripherals that you know exist on the master system or clone systems.

Planning the Creation of a Flash Archive

You can create an archive from the master system for an initial installation. Or, if you have already installed an archive on clone systems, you can create a differential archive from two system images. The differential archive installs only the differences between the two images.

Planning to Create the Flash Archive for an Initial Installation

After you install the master system, the next task in the Flash Archive installation process is to create a flash archive. Files on the master system are copied to a flash archive along with various pieces of identification information. You can create a flash archive while the master system is running in multiuser mode or single-user mode. You can also create a flash archive after you boot from one of the following:

- Oracle Solaris Operating System DVD
- Oracle Solaris Software - 1 CD
- An image of the Oracle Solaris Software CDs and the Oracle Solaris Languages CDs

Note – Starting with the Oracle Solaris 10 9/10 release, only a DVD is provided. Oracle Solaris Software CDs are no longer provided.

Caution – A flash archive cannot be properly created when a non-global zone is installed. The Flash Archive feature is not compatible with the Oracle Solaris Zones partitioning technology. If you create a flash archive, the behavior of the archive when deployed is undefined in these conditions:

- The archive is created in a non-global zone
- The archive is created in a global zone that has non-global zones installed

Creating Flash Archive With RAID-1 Volumes

You can create a flash archive when you have Solaris Volume Manager RAID-1 volumes configured. The Flash Archive creation software removes all RAID-1 volume information from the archive to keep the integrity of the clone system. With JumpStart you can rebuild the RAID-1 volumes by using a JumpStart profile. With Live Upgrade you create a boot environment with RAID-1 volumes configured and install the archive. The Oracle Solaris installation program cannot be used to install RAID-1 volumes with a flash archive.

- For examples of RAID-1 volumes in JumpStart profiles, see "Profile Examples" in *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*.

- For examples of Live Upgrade boot environments configured with RAID-1 volumes, see "Creating a New Boot Environment" in *Oracle Solaris 10 1/13 Installation Guide: Live Upgrade and Upgrade Planning*.

Note – Veritas VxVM stores configuration information in areas not available to Flash Archive. If Veritas VxVM file systems have been configured, you should not create a flash archive. Also, Oracle Solaris installations, including the JumpStart and Live Upgrade methods do not support rebuilding VxVM volumes at installation time. Therefore, if you are planning to deploy Veritas VxVM software using a flash archive, the archive must be created prior to configuring the VxVM file systems. The clone systems must be then configured individually after the archive has been applied and the system rebooted.

Creating an Archive That Contains Large Files

The default copy method that is used when you create a flash archive is the cpio utility. Individual file sizes cannot be over 4 GB. If you have large individual files, you can create an archive with the pax copy method. The flarcreate command with the -L pax option uses the pax utility to create an archive without limitations on individual file sizes. Individual file sizes can be greater than 4 GB.

Planning to Create the Flash Archive Differential Archive for an Update

If you have a clone system that is already installed with an archive and want to update it, you can create a differential archive that contains only the differences between two images, the unchanged master image and an updated master image. The differences between these two images is the differential archive.

- One image is running on the master system that was the original software installed on the clone system. This image might need be installed on the master system if it was saved in a directory for future use.
- The other image is to be accessed and used for comparison. This image contains the new additions or deletions that will be installed on the clone systems.

After you update a clone system with a differential archive, only the files that are in the differential archive are changed on the clone system. Scripts can be used to customize the archive before or after installation, which is especially helpful for reconfiguration.

You can install a Flash Archive differential archive with the JumpStart installation method, or use Live Upgrade to install a differential archive on an inactive boot environment.

An unchanged master image should be saved after the initial installation so this image can be accessed by any of the following methods:

- A Live Upgrade boot environment mounted on a directory that uses the `lumount` command. For a description of a Live Upgrade boot environment, see Chapter 2, "Live Upgrade (Overview)," in *Oracle Solaris 10 1/13 Installation Guide: Live Upgrade and Upgrade Planning*.

- A clone system that is mounted over NFS with root permissions.

- A system backup that can be restored with the `ufsdump` command.

For step-by-step instructions, see "How to Create a Flash Archive Differential Archive With an Updated Master Image" on page 43.

Customizing an Archive's Files and Directories

When you create a flash archive, some files and directories that are to be copied from the master system can be excluded. If you have excluded a directory, you can also restore specified files or subdirectories under that directory. For example, you could create an archive that excludes all files and directories in `/a/aa/bb/c`. The content of the `bb` subdirectory could be included. The only content would then be in the `bb` subdirectory.

> ⚠ **Caution** – Use the `flarcreate` file-exclusion options with caution. If you exclude some directories, others that you were unaware of might be left in the archive, such as system configuration files. The system would then be inconsistent and the installation would not work. Excluding directories and files is best used with data that can easily be removed without disrupting the system, such as large data files.

The following table lists the `flarcreate` command options that can exclude files and directories and restore files and subdirectories.

How Specified?	Options That Exclude	Options That Include
Specify the name of the directory or file	-x *exclude-dir/filename*	-y *include-dir/filename*
Use a file that contains a list	-X *list-filename*	-f *list-filename*
	-z *list-filename*	-z *list-filename*

For descriptions of these options, see Table 6–6.

For examples of customizing an archive, see "Examples of Creating a Flash Archive and Customizing Files" on page 41.

Customizing an Archive With Scripts

After the software is installed on the master system, special scripts can be run during creation, installation, postinstallation and first reboot. These scripts enable you to do the following:

- Configure applications on clone systems.

  You can use a JumpStart script for some uncomplicated configurations. For more complicated configurations, special configuration-file processing might be necessary on the master system or before or after installation on the clone system.

- Protect local customizations on clone systems.

  Local preinstallation and postinstallation scripts reside on the clone. These scripts protect local customizations from being overwritten by the Flash Archive software.

- Identify nonclonable, host-dependent data that enables you to make the archive host independent.

  Host independence is enabled by modifying such data or excluding it from the archive. An example of host-dependent data is a log file.

- Validate software integrity in the archive during creation.

- Validate the installation on the clone system.

Guidelines for Creating a Custom Script

When creating scripts other than the reboot script, follow these guidelines to ensure that the script does not corrupt the OS or otherwise disrupt the system. These guidelines enable the use of Live Upgrade, which creates a new boot environment for installation of the OS. The new boot environment can be installed with an archive while the current system is running.

Note – These guidelines are not for reboot scripts that are allowed to run daemons or make other types of modification to the root (/) file system.

- Scripts must not affect the currently running system. The currently running OS might not be the one running when the flash archive is installed.

- Scripts must not start or stop any daemon processes.

- Scripts must not depend on the output of commands such as ps, truss, or uname, which are dependent on the OS. These commands report information about the currently running system.

- Scripts must not send any signals or otherwise affect any currently running processes.

- Scripts can use standard UNIX commands that facilitate shell scripting such as expr, cp, and ls.

For an overview of Live Upgrade, see Chapter 2, "Live Upgrade (Overview)," in *Oracle Solaris 10 1/13 Installation Guide: Live Upgrade and Upgrade Planning*.

Flash Archive Sections

A flash archive contains the following sections. Some sections can be used by you to identify and customize the archive and view status information on the installation. For a further description of each section, see Chapter 6, "Flash Archive (Reference)."

TABLE 2–2 Flash Archive Sections

Section Name	Informational Only	Description
Archive cookie	X	The first section contains a cookie that identifies the file as a Flash Archive.
Archive identification		The second section contains keywords with values that provide identification information about the archive. Some identification information is supplied by the archive software. Other specific identification information can be added through options to the flarcreate command.
User-defined		You can define and insert sections to customize the archive after the archive identification section. The flash archive does not process any sections that you insert. For example, a section could contain a description of the archive or perhaps a script to check the integrity of an application.
Manifest	X	This section is produced for a Flash Archive differential archive and is used for validating a clone system. The manifest section lists the files on a system to be retained, added to, or deleted from the clone system. This section is informational only, lists the files in an internal format, and cannot be used for scripting.
Predeployment, Postdeployment, Reboot	X	This section contains internal information that the Flash Archive software uses before and after installing an OS image. Any scripts that you have provided are included in this section.
Summary		This section contains messages about the archive creation. The section also records the activities of predeployment and postdeployment scripts. You can view the success of the installation in this section by writing a script to send output to this section.
Archive files	X	The archive files section contains the files that have been gathered from the master system.

When to Create the Archive for an Initial Installation

Create the archive when the system is in as static a state as possible. Create the archive after software is installed on the master system and before software is configured.

Where to Store the Flash Archive

After you create the flash archive, you can save the archive on the hard disk of the master system or on a tape. After you save the archive, you can copy this archive to any file system or media that you choose, for example:

- NFS server
- HTTP or HTTPS server
- FTP server
- Tape
- CD, DVD
- Diskette
- Local drive of clone system that you want to install

Compressing the Archive

When you create the flash archive, you can specify that the archive be saved as a compressed file by using the compress(1) utility. An archive that is compressed requires less disk storage space and creates less congestion when you install the archive over a network.

Planning the Installation of a Flash Archive

The final task in the Flash Archive installation process is to install a flash archive on clone systems. You can use any of the Oracle Solaris installation methods to install a flash archive on clone systems.

Note – Starting with the Oracle Solaris 10 9/10 release, Auto Registration is enabled by default. The impact of Auto Registration on your work with Flash Archive varies depending on which installation method is used.

If you are using a pre-Oracle Solaris 10 9/10 archive, there is no Auto Registration impact.

For step-by-step instructions for installing a flash archive through the various installation methods, see the following resources:

- Oracle Solaris installation program: "Installing a Flash Archive With the Oracle Solaris Installation Program" on page 52.
- JumpStart installation program: "To Prepare to Install a Flash Archive With a JumpStart Installation" in *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*.
- Live Upgrade: "Installing Flash Archives on a Boot Environment" in *Oracle Solaris 10 1/13 Installation Guide: Live Upgrade and Upgrade Planning*.

3

Creating Flash Archives (Tasks)

This chapter provides the procedures for creating a flash archive. These procedures include installing a master system and then creating a Flash Archive from that master system. You can also create a differential archive if you have previously installed an archive on a clone system. When the differential archive is created, two images are compared: the unchanged master image and an updated image. The differential archive installs only the differences between the two images. Also, procedures to create scripts are provided to reconfigure or otherwise customize the archive.

For limitations when creating or installing a Flash Archive, see Table 2–1.

Creating a Flash Archive (Task Maps)

TABLE 3–1 Task Map: Creating a Flash Archive to Install With an Initial Installation

Task	Description	For instructions
Install your chosen software configuration on the master system.	Determine the configuration that meets your needs and use any of the Oracle Solaris installation methods to install the master system.	"Installing the Master System" on page 32
(Optional) Create customization scripts.	Determine whether you need to create scripts to do the following: ■ Customize or reconfigure the archive ■ Protect local changes on clone systems	"Creating Customization Scripts" on page 33
Create the flash archive.	Use the `flarcreate` command to create an archive.	"How to Create a Flash Archive for an Initial Installation" on page 37
(Optional) Save a copy of the archive.	Keep a copy of the archive for future comparison to update a clone system with a differential archive.	"How to Create a Flash Archive for an Initial Installation" on page 37

TABLE 3–2   Task Map: Creating a Flash Archive to Update a Clone System

Task	Description	For instructions
Prepare the master image.	Make changes to the unchanged master image such as adding or deleting packages or installing patches.	"Installing the Master System" on page 32
(Optional) Create customization scripts.	Determine whether you need to create scripts to do the following: ■ Customize or reconfigure the archive ■ Protect local changes on clone systems	"Creating Customization Scripts" on page 33
Create the Flash Archive differential archive.	1. Mount the unchanged master image. 2. Use the flarcreate command to compare the two images and create the differential archive.	"How to Create a Flash Archive Differential Archive With an Updated Master Image" on page 43

Installing the Master System

You install the master system with the software configuration that you want other systems to have. You can install clone systems with an initial installation that overwrites all files on the system or with an update that only includes the differences between two images. For an initial installation, use any of the Oracle Solaris installation methods to install the Oracle Solaris OS on the master system.

You can then customize the installation:

■ Delete software.
■ Add software.
■ Modify configuration files.
■ Add support for peripheral devices on the clone system.

After the master system installation is complete, you can create custom scripts or use the flarcreate command to create the archive.

■ To create custom scripts, see "Creating Customization Scripts" on page 33.
■ To create the archive, see "Creating a Flash Archive" on page 37.

If you have previously installed an archive on a clone system, you can update that system with changes by using a differential archive. The changes are made to the original image such as installing patches, or adding and removing packages. The differential archive overwrites only the files specified in the archive. To find out how to update the original master image and create a differential archive, see "How to Create a Flash Archive Differential Archive With an Updated Master Image" on page 43.

For more information about how to install the Oracle Solaris OS on the master system, see "Choosing an Oracle Solaris Installation Method" in *Oracle Solaris 10 1/13 Installation Guide: Planning for Installation and Upgrade*.

Creating Customization Scripts

Scripts can customize the archive. Use these scripts for the following purposes:

- A precreation script validates the archive at creation time and prepares the archive for later customization, especially differential archives. This script also can create a user-defined section in the archive.
- A predeployment script validates the archive during installation and prepares the archive for later customization.
- A postdeployment script reconfigures a new system image on a clone system.
- A reboot script processes a final reconfiguration after the system is rebooted.

For guidelines about creating scripts, see "Guidelines for Creating a Custom Script" on page 28.

Creating a Precreation Script

This script runs during archive creation. The script has various uses.

- Validates the contents and the integrity of the software. The script fails the archive creation if the integrity is broken.
- Prepares products for later customization on clone system.
- Registers other installation scripts dynamically during archive creation.
- Adds a message to the flash-creation summary file. The message must be short and record only that scripts were started and finished and the results. You can view the results in the summary section.

For guidelines about creating the precreation script, see "Guidelines for Creating a Custom Script" on page 28. Then store the script in the /etc/flash/precreation directory.

EXAMPLE 3–1 Excerpts From a Precreation Script

The following examples are excerpts from a precreation script.

- To log the start time in the summary section, use the following example:

    ```
    echo "Myapp precreation script started">> $FLASHDIR/summary
    ```

- To check the software integrity, use the flcheck command. This command cannot be used at the command line. The syntax for this command is as follows:

    ```
    flcheck Software component files and directories ...| -
    ```

EXAMPLE 3–1 Excerpts From a Precreation Script *(Continued)*

For example, to validate the files and directories, use the following example:

```
flcheck Software component files and directories
If Not in selection - refuse creation
```

```
echo "Myapp integrity damage">>$FLASHDIR/summary
```

Or, to keep new files and directories that are unexpected and not fail the archive creation, use the following example:

```
flcheck Software component files and directories
If Not in selection include by force
flinclude Software component
```

- To register deployment scripts and data:

    - Copy the script to the /etc/flash/predeployment directory:

    - To register the script dynamically during archive creation, copy the script to the $FLASHDIR/predeployment directory.

- To add application-specific data in a user-defined section, use the following example:

    ```
    cp custom-section $FLASHDIR/custom_sections/Myapp
    ```

- To log the success of the installation in the summary section

    ```
    echo "product one flash preparation started." >>$FLASH_DIR/summary
    ...
    echo "product one flash preparation finished successfully">>$FLASH_DIR/summary
    ```

EXAMPLE 3–2 Precreation Script

```
#!/bin/sh
echo "Test precreation script started" >> $FLASH_DIR/summary
cat /opt/TestApp/critical_file_list | flcheck -
if [ $? != 0 ]; then
        echo "Test precreation script failure" >> $FLASH_DIR/summary
        exit 1
fi
echo "Test precreation script started" >> $FLASH_DIR/summary
/opt/TestApplication/license_cloning
        $FLASH_DIR/predeployment/.TestApplicationLicenceTransfer \
        $FLASH_DIR/custom_sections/TestApplicationLicenceCounter
echo "Test precreation script finished" >> $FLASH_DIR/summary
exit 0
```

Using a Precreation Script to Create a User-Defined Archive Section

A precreation script can create a user-defined section in the archive to provide specific application information. This section is intended for archive maintenance. The script must be

put in the $FLASH_DIR/sections directory. The flash archive does not process a user-defined section. For example, a section could contain a description of the archive or perhaps a script to check the integrity of an application.

A user-defined section requires the following format:

- Must be line oriented
- Must terminate with newline (ASCII 0x0a) characters
- Can have unlimited length of individual lines
- Must encode binary data by using base64 or a similar algorithm

The precreation script is run before the installation of the archive. If the script is meant to validate the archive, it is kept in the archive. If the script is meant to preserve local configurations on the clone system, it is kept on the clone system. This script also can analyze and collect local data necessary for later customization. For example, client-specific information can be saved before being overwritten by files about to be extracted. This information can then be used in the final stage after extraction.

For guidelines about creating the predeployment script, see "Guidelines for Creating a Custom Script" on page 28.

Store the script in one of the following directories:

- For archive validation, store in the /etc/flash/predeployment directory.

- If you are referencing with a precreation script, store in the $FLASH_DIR/preinstall directory.

- If you are preserving configurations on a clone system, provide the path to the script that is stored on the clone system with the local_customization keyword in the jumpstart profile.

EXAMPLE 3–3 Predeployment Script

```
#!/bin/sh
$FLASH_DIR/TestApplication/check_hardware
if [ $? != 0 ]; then
        echo Unsupported hardware
        exit 1
fi
$FLASH_DIR/TestApplication/check_licence_key
if [ $? != 0 ]; then
    echo No license for this host
    exit 1
fi
        $FLASH_DIR/TestApplication/deploy_license_key \
        $FLASH_DIR/TestApplication/.TestApplicationLicenceTransfer
        $FLASH_DIR/TestApplication/save_data_files $FLASH_DIR/flash
exit 0
```

Creating a Postdeployment Script

The postdeployment script is kept in the archive or stored in a local directory on the clone system and runs after installation. The script reconfigures a new system image on a clone system. If the script is stored in the archive, the changes affect all the clone systems. If the script is stored in a local directory on the clone system, the changes affect only the clone system. For example, client-specific information that is saved by a predeployment script can be applied to the clone environment, completing the installation.

Postdeployment scripts can also be used to clean up files after the archive is installed. For example, log files such as the files in /var/adm can be cleaned out.

Note – Not all log files need a script for cleanup. Log files in /var/tmp can be excluded when the archive is created.

For guidelines about creating the postdeployment script, see "Guidelines for Creating a Custom Script" on page 28.

Store the script in one of the following directories:

- To affect all clone systems, store the script in the /etc/flash/postdeployment directory.
- To affect only a local clone system, provide the path to the script that is stored on the clone system with the local_customization keyword in the JumpStart profile.

EXAMPLE 3–4 Postdeployment Script

```
#!/bin/sh
$FLASH_DIR/TestApplication/clone_reconfiguration
$FLASH_DIR/TestApplication/restore_data $FLASH_DIR/flash
```

Creating a Reboot Script

The reboot script is kept in the archive and runs after the system is rebooted. The script does any final configuration after system reconfiguration.

After you install the flash archive on a clone system, some host-specific files are deleted and are re-created for the clone machine. The installation program uses the sys-unconfig(1M) command and the sysidtool(1M) programs to delete and re-create host-specific network configuration files. The files that are re-created include, for example, /etc/hosts, /etc/defaultrouter, and /etc/defaultdomain. You can use the reboot script for any final reconfiguration.

Once you create the reboot script, store the script in the /etc/flash/reboot directory.

EXAMPLE 3–5   Creating a Reboot Script

```
#!/bin/sh
$FLASH_DIR/TestApplication/finalize_license
```

Creating a Flash Archive

You can create an archive with an initial installation that overwrites all the files on the clone system, or you can create a differential archive that only overwrites the changes that are specified. For an overview of a differential archive, see "Planning to Create the Flash Archive Differential Archive for an Update" on page 26.

⚠️ **Caution** – A flash archive cannot be properly created when a non-global zone is installed. The Flash Archive feature is not compatible with the Oracle Solaris Zones partitioning technology. If you create a flash archive, the resulting archive is not installed properly when the archive is deployed under these conditions:

- The archive is created in a non-global zone
- The archive is created in a global zone that has non-global zones installed

▼ How to Create a Flash Archive for an Initial Installation

After you install the master system, create a flash archive to use to install other systems.

1   Become superuser or assume an equivalent role.

Roles contain authorizations and privileged commands. For more information about roles, see "Configuring RBAC (Task Map)" in *System Administration Guide: Security Services*.

2   Boot the master system and run it in as inactive a state as possible.

When possible, run the system in single-user mode. If that is not possible, shut down any applications that you want to archive and any applications that require extensive operating system resources.

You can create a flash archive while the master system is running in multiuser mode, single-user mode, or while booted from one of the following:

- Oracle Solaris Operating System DVD.

  Note – Starting with the Oracle Solaris 10 9/10 release, only a DVD is provided. Oracle Solaris Software CDs are no longer provided.

- Oracle Solaris Software - 1 CD.

■ An image of the Oracle Solaris Software. If you are using CD media, the image could include the Oracle Solaris Languages CDs if needed.

3   To create the archive, use the `flarcreate` command.

`# flarcreate -n` *name options path/filename*

name   The name that you give the archive. The *name* you specify is the value of the `content_name` keyword.

options   For a description of options, see "flar Command" on page 69.

path   The path to the directory in which you want to save the archive file. If you do not specify a path, `flarcreate` saves the archive file in the current directory.

filename   The name of the archive file.

■ If the archive creation is successful, the `flarcreate` command returns an exit code of 0.

■ If the archive creation fails, the `flarcreate` command returns a nonzero exit code.

4   Make a copy of the archive and save it.

The copy can be used in the future to update a clone system with a differential archive.

Creating a Flash Archive (Examples)

File systems can be copied exactly or can be customized by excluding some directories or files. You can achieve the same results by using different options. Use the options that best suit your environment.

The file systems in the following examples have been greatly simplified for clarification. Rather than use file system names such as /var, /usr, or /opt, the master system file structure for these examples is the following:

```
/aaa/bbb/ccc/ddd
/aaa/bbb/fff
/aaa/eee
/ggg
```

Caution – Use the `flarcreate` file-exclusion options with caution. If you exclude some directories, others that you were unaware of might be left in the archive, such as system configuration files. The system would then be inconsistent and the installation would not work. Excluding directories and files is best used with data that can easily be removed without disrupting the system, such as large data files.

Examples of Creating a Flash Archive

EXAMPLE 3–6 Creating a Duplicate Archive

In this example, the archive is named `archive1`. This archive is copied exactly from the master system and then compressed. The archive is an exact duplicate of the master system and is stored in `archive1.flar`.

```
# flarcreate -n archive1 -c archive1.flar
```

To check the file structure of the archive:

```
# flar info -l archive1.flarlost+found
export
export/home
export/home/lost+found
var
var/sadm
var/sadm/install
var/sadm/install/admin
var/sadm/install/admin/default
var/sadm/install/logs
var/sadm/install/contents
var/sadm/install/.lockfile
var/sadm/install/.pkg.lock
var/sadm/pkg
var/sadm/pkg/sunwocfd
var/sadm/pkg/sunwocfd/install
var/sadm/pkg/sunwocfd/install/copyright
var/sadm/pkg/sunwocfd/save
var/sadm/pkg/sunwocfd/save/pspool
var/sadm/pkg/sunwocfd/save/pspool/SUNWocfd
.....
.....
    usr/bin/sparcv7
usr/bin/sparcv7/savecore
usr/bin/sparcv7/gcore
....
....
usr/lib/diff3prog
usr/lib/madv.so.1
usr/lib/mpss.so.1
usr/lib/cpu
usr/lib/cpu/sparcv8plus
....
....
devices/pseudo/udp6@0:udp6
devices/pseudo/udp@0:udp
devices/pseudo/tcp@0:tcp
devices/pseudo/iwscn@0:iwscn
devices/pseudo/wc@0:wscons
devices/pseudo/tcp6@0:tcp6
devices/pseudo/sctp6@0:sctp6
var/fm/fmd/ckpt
var/fm/fmd/rsrc
```

EXAMPLE 3–6   Creating a Duplicate Archive   *(Continued)*

```
kernel/drv/st.conf
kernel/drv/st.conf
kernel/drv/st.conf
kernel/drv/st.conf
#
```

EXAMPLE 3–7   Creating an Archive That Contains Large Files

In this example, some individual files are greater than 4 GB. The default archiving utility, cpio, cannot handle these large files. The -L pax copy method is used to create an archive that contains large individual files. The archive is named archive1. This archive is copied exactly from the master system and then compressed. The archive is an exact duplicate of the master system and is stored in archive1.flar.

```
# flarcreate -l pax -n archive1 -c archive1.flar
```

To check the file structure of the archive, type the following command:

```
# flar info -l archive1.flar
aaa
aaa/bbb
aaa/bbb/ccc
aaa/bbb/ccc/ddd
aaa/bbb/fff
aaa/eee
aaa/eee
ggg
```

EXAMPLE 3–8   Creating an Archive From an Alternate Root (/) File System

In this example, the archive is named archive4. This archive is copied exactly from the master system and then compressed. The archive is an exact duplicate of the master system and is stored in archive4.flar. The -R option is used to create the archive from another directory tree.

```
# flarcreate -n archive4 -c -R /x/yy/zz archive4.flar
```

EXAMPLE 3–9   Creating an Archive and Adding a Description

In this example, the archive is named archive3. This archive is copied exactly from the master system and then compressed. Options add descriptions to the archive-identification section, which can help you to identify the archive later. For information about keywords, their values, and formats, see "Flash Archive Keywords" on page 65.

```
# flarcreate -n archive3 -i 20000131221409 -m pumbaa \
 -e "Solaris 8 Print Server" -a "Mighty Matt" -U "Internal Finance" \
 -T server archive3.flar
```

EXAMPLE 3–9 Creating an Archive and Adding a Description *(Continued)*

After the archive is created, you can access the archive identification section that contains the detailed description. The following example shows an archive identification section.

```
section_begin=identification
            files_archived_method=cpio
            files_compressed_method=compress
            files_archived_size=259323342
            files_unarchived_size=591238111
            creation_date=20000131221409
            creation_master=pumbaa
            content_name=Finance Print Server
            content_type=server
            content_description=Solaris 8 Print Server
            content_author=Mighty Matt
            content_architectures=sun4u
            creation_node=pumbaa
            creation_hardware_class=sun4u
            creation_platform=SUNW,Sun-Fire
            creation_processor=sparc
            creation_release=5.9
            creation_os_name=SunOS
            creation_os_version=s81_49
            x-department=Internal Finance
```

Examples of Creating a Flash Archive and Customizing Files

EXAMPLE 3–10 Excluding and Including Files and Directories

In this example, the archive is named archive2. This archive is copied from the master system but is not an exact copy. The content under the /aaa directory is excluded but the content in /aaa/bbb/ccc remains.

```
# flarcreate -n archive2 -x /aaa -y /aaa/bbb/ccc  archive2.flar
```

To check the file structure of the archive, type the following command. The excluded directories that include copied files appear, but only the files that were restored contain data.

```
# flar info -l aaa
aaa
aaa/bbb/ccc
aaa/bbb/ccc/ddd
aaa/bbb
ggg
```

EXAMPLE 3–11 Excluding and Including Files and Directories by Using Lists

In this example, the archive is named archive5. This archive is copied from the master system but is not an exact copy.

EXAMPLE 3–11 Excluding and Including Files and Directories by Using Lists   *(Continued)*

The exclude file contains the following list:

/aaa

The include file contains the following list:

/aaa/bbb/ccc

The content under the /aaa directory is excluded, but the content in /aaa/bbb/ccc remains.

```
# flarcreate -n archive5 -X exclude -f include  archive5.flar
```

To check about the file structure of the archive, type the following command. The excluded directories that include copied files appear, but only the files that were restored contain data.

```
# flar info -l archive5.flar
aaa
aaa/bbb/ccc
aaa/bbb/ccc/ddd
aaa/bbb
ggg
```

EXAMPLE 3–12 Excluding Files and Directories by Using a List and Restoring a Directory

You can combine options -x, -y, -X and -f. In this example, options -X and -y are combined. The archive is named archive5. This archive is copied from the master system but is not an exact copy.

The exclude file contains the following list:

/aaa

The -Y option restores the /aaa/bbb/ccc directory. The following command produces the archive.

```
# flarcreate -n archive5 -X exclude -y /aaa/bbb/ccc  archive5.flar
```

To check the file structure of the archive, type the following command. The excluded directories that include copied files appear, but only the files that were restored contain data.

```
# flar info -l archive5.flar
aaa
aaa/bbb
aaa/bbb/ccc
aaa/bbb/ccc/ddd
ggg
```

EXAMPLE 3–13   Excluding and Including Files and Directories by Using a List With the -z Option

In this example, the archive is named archive3. It is copied from the master system but is not an exact copy. The files and directories to be selected are included in the filter1 file. Within the files, the directories are marked with a minus (-) or a plus (+) to indicate which files to exclude and restore. In this example, the directory /aaa is excluded with a minus and the subdirectory /aaa/bbb/ccc is restored with a plus. The filter1 file contains the following list:

```
- /aaa
+ /aaa/bbb/ccc
```

The following command produces the archive.

```
# flarcreate -n archive3 -z filter1 archive3.flar
```

To check the file structure of the archive, type the following command. The excluded directories that include copied files appear, but only the files that were restored contain data.

```
# flar info -l archive3.flar
aaa
aaa/bbb
aaa/bbb/ccc
aaa/bbb/ccc/ddd
ggg
```

Creating a Flash Archive Differential Archive With an Updated Master Image

Before creating a differential archive, you need two images to compare: an unchanged master image and an updated master image. One image is the unchanged master image that has been kept unchanged. This image was stored and needs to be accessed. The second image is the unchanged master image that is updated with minor changes. The root (/) file system is the default for the new image, but you can access this image if it has been stored elsewhere. After you have the two images, you can create a differential archive, which contains only the differences between the two images. The differential archive can then be installed on clones that were installed previously with the unchanged master image.

▼ How to Create a Flash Archive Differential Archive With an Updated Master Image

Before You Begin   A copy of the unchanged master image must be kept protected from changes and available for mounting later.

1 Prepare the master system with changes. The master system should be running a duplicate of the original archive.

2 Update the unchanged master image with any of the following changes.

- Delete packages.
- Add packages or patches.
- Modify configuration files.
- Add support for peripheral devices on the clone system.

3 (Optional) Create custom scripts.

See "Creating Customization Scripts" on page 33.

4 Become superuser or assume an equivalent role.

Roles contain authorizations and privileged commands. For more information about roles, see "Configuring RBAC (Task Map)" in *System Administration Guide: Security Services*.

5 Deliver the unchanged master image in a mount point.

If the unchanged master image is stored on an inactive boot environment, retrieve it by using the lumount command.

```
# lumount BE-name mountpoint
```

BE-name   Specifies the boot environment name where the unchanged master image is stored

mountpoint   Specifies a root (/) file system where the image is stored

In the following example, the inactive boot environment is named unchanged_master1. The mount point is the directory /a on the master system.

```
# lumount  unchanged_master1 /a
```

If the image is stored on a clone, mount the clone by using NFS.

a. On the master system, share the clone's root (/) file system and give the master root permissions on the clone system.

```
# share -F nfs -o rw,root=master-system "/"
```

master-system is the name of the master system.

b. On the master system, mount the clone.

```
# mount -F nfs clone-system:/ master-dir
```

clone-system   Specifies the name of the system to be mounted

master-dir   Specifies the directory where the unchanged master image is stored

If you saved the image with the ufsdump command, use the ufsrestore command to retrieve a copy. For information about how to use these commands, see Chapter 23, "UFS Backup and Restore Commands (Reference)," in *System Administration Guide: Devices and File Systems*.

6   Create the differential archive.

```
# flarcreate -n archive-name -A unchanged-master-image-dir \
  options path/filename
```

archive-name	Specifies the name that you give the archive. The *archive-name* you specify is the value of the content_name keyword. The name is listed in the archive-identification section.
-A *unchanged-master-image-dir*	Creates a differential archive by comparing a new system image with the image that is specified by the *unchanged-master-image-dir* argument. By default, the new system image is root (/). You can change the default with the -R option. *unchanged-master-image-dir* is a directory where the unchanged system image is stored or mounted through UFS, NFS, or the lumount command.
	You can include and exclude some files by using the options for contents selection. For a list of options, see "flar Command" on page 69.
options	For a description of options, see "flar Command" on page 69.
path	Specifies the path to the directory in which you want to save the archive file. If you do not specify a path, flarcreate saves the archive file in the current directory.
filename	Specifies the name of the archive file.

- If the differential archive creation is successful, the flarcreate command returns an exit code of 0.
- If the differential archive creation fails, the flarcreate command returns a nonzero exit code.

For procedures about installing an archive, see "To Prepare to Install a Flash Archive With a JumpStart Installation" in *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*.

Example 3–14   Creating a Differential Archive With the New Master Image on the Master System

In this example, the directory for unchanged master image is named unchanged_master1. The new master image that contains changes is the root (/) directory. The new master image is compared to the unchanged master image and the resulting differential archive is then

compressed. The differential archive is stored in `diffarchive1.flar` file. The archive contains files that are to be deleted, changed, or added when installed.

```
# flarcreate -n diffarchive1 -A /a/unchanged_master1 -c diffarchive1.flar
```

Example 3–15 Creating a Differential Archive With the Images Stored on an Inactive Boot Environment

In this example, the unchanged master image, `unchanged_master1`, is stored on an inactive boot environment and is accessed by mounting the boot environment. The new master image is the root (`/`) directory. The new master image is compared to the unchanged master and the resulting differential archive is then compressed. The archive is stored in `diffarchive4.flar`. The archive contains files that are to be deleted, changed, or added when installed.

```
# lumount unchanged_master1 /a
# flarcreate -n diffarchive4 -A /a -c  diffarchive4.flar
```

▼ How to Create a Flash Archive Differential Archive by Using Live Upgrade

To manage system updates, you can use Live Upgrade to copy the OS, which creates a new boot environment. This copy can be compared to the master system that has been updated with minor changes. The resulting Flash Archive differential archive can then be installed on clone systems.

For more information about Live Upgrade, see Chapter 2, "Live Upgrade (Overview)," in *Oracle Solaris 10 1/13 Installation Guide: Live Upgrade and Upgrade Planning*.

1 **From the unchanged master system, create a new boot environment by using the `lucreate` command.**

This new boot environment is an exact copy of the master system and can be used to create the differential archive.

2 **Become superuser or assume an equivalent role.**

Roles contain authorizations and privileged commands. For more information about roles, see "Configuring RBAC (Task Map)" in *System Administration Guide: Security Services*.

3 **Check the status of the two boot environments.**

```
# lustatus copy_BE
boot environment   Is         Active   Active    Can      Copy
Name               Complete   Now      OnReboot  Delete   Status
------------------------------------------------------------------
master_BE          yes        yes      yes       no       -
```

```
copy_BE          yes      no      no      yes      -
```

4 **Update the master image with any of the following changes if necessary:**

- Delete packages
- Add packages or patches
- Modify configuration files
- Add support for peripheral devices on the clone system

5 **(Optional) Create custom scripts.**

See "Creating Customization Scripts" on page 33.

6 **Create the differential archive.**

a. **Mount the newly created boot environment.**

   # lumount *BE-name* **/a**

b. **Create the differential archive by comparing the master system to the boot environment.**

   # flarcreate -n *archive-name* **-A** *new-BE-dir*\ *options path/filename*

archive-name	Specifies the name that you give the archive.
-A *new-BE-dir*	Creates a differential archive by comparing a new system image with the image that is specified by the *new-BE-dir* argument.
options	For a list of options, see "flar Command" on page 69.
path	Specifies the path to the directory in which you want to save the archive file. If you do not specify a path, flarcreate saves the archive file in the current directory.
filename	Specifies the name of the archive file.

c. **Unmount the new boot environment.**

   # luumount *BE-copy*

   The flarcreate command returns an exit code.

   - If the creation is successful, an exit code of 0 is returned.
   - If a failure occurs, a nonzero exit code is returned.

7 **Install the Flash Archive differential archive by using a JumpStart profile.**

The clone systems that are to be installed must be a duplicate of the original master system or the installation fails.

The following example profile installs a differential archive, test.diff, on the device c1t1d0s0.

```
JumpStart profile
-----------------------
```

```
install_type  flash_update
archive_location http server /rw/test.diff
root_device c1t1d0s0
```

Example 3–16   Creating a Differential Archive by Using Live Upgrade

In this example, master_BE is the name of the current boot environment. copy_BE is the name of the new boot environment. The file systems root (/) and /usr are placed on s0 and s3. The lustatus command reports that the new boot environment copy is complete. The SUNWMAN package is added to the master system. After the master system is updated by adding the SUNWMAN package, the flarcreate command creates a differential archive by comparing the changed master and the unchanged new boot environment.

```
# lucreate -c master_BE  -m /:/dev/dsk/c0t1d0s0:ufs \
-m /usr:/dev/dsk/c0t1d0s3:ufs -n copy_BE
# lustatus
# pkgadd SUNWman
# lumount copy_BE /a
# flarcreate -n test.diff -c -A /a /net/server/export/test.diff
# luumount copy_BE
```

Install the differential archive on clone systems. For procedures describing how to install an archive, see "To Prepare to Install a Flash Archive With a JumpStart Installation" in *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*.

The following image shows the creation of the new boot environment by using the lucreate command.

Original Master System
2 Physical Disks

c0t0d0   c0t1d0

root (/)

/swap

Master System After
New Boot Environment Creation

c0t0d0   c0t1d0

root (/)   — Copy →   root (/)

/swap   ← Share →   /swap
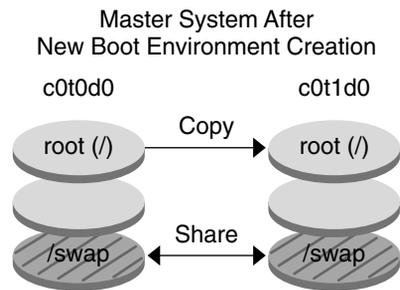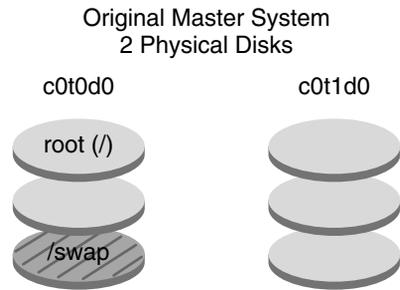
Command: # lucreate
            -m /:/dev/dsk/c0t1d0s0:ufs \
            -n second_disk

◆ ◆ ◆ **C H A P T E R 4**

4

Installing and Administering Flash Archives
(Tasks)

This chapter provides step-by-step procedures for installing a flash archive by using the Oracle
Solaris installation program. This chapter also provides references to procedures for installing a
flash archive when using other installation programs. Also, step-by-step procedures for
administering an archive are provided.

⚠ **Caution** – When installing the Oracle Solaris OS with a flash archive, the archive and the
installation media must contain identical operating system versions and Kernel Update (KU)
patch. For example, if the archive is a Oracle Solaris 10 operating system and you are using
DVD media, then you must use Oracle Solaris 10 DVD media to install the archive. If the
operating systems versions and Kernel Update (KU) patch levels do not match, the installation
on the target system might fail.

About Flash Archive Installation

You can use any of the Oracle Solaris installation methods to install a flash archive for an initial
installation. You must use JumpStart or Live Upgrade to install a Flash Archive differential
archive.

- Oracle Solaris installation program – Covered in this chapter.
- Live Upgrade – See "Installing Flash Archives on a Boot Environment" in *Oracle
  Solaris 10 1/13 Installation Guide: Live Upgrade and Upgrade Planning*.
- JumpStart installation program – See "Creating a Profile" in *Oracle Solaris 10 1/13
  Installation Guide: JumpStart Installations* and "To Prepare to Install a Flash Archive With a
  JumpStart Installation" in *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*.
- WAN boot installation method – See Chapter 10, "WAN Boot (Overview)," in *Oracle
  Solaris 10 1/13 Installation Guide: Network-Based Installations*.

You must use either JumpStart or Live Upgrade to install an update with a Flash Archive
differential archive.

For limitations when creating or installing a flash archive, see Table 2–1.

Installing a Flash Archive With the Oracle Solaris Installation Program

To use the Oracle Solaris installation program to install a flash archive, use the following procedure.

▼ How to Install a Flash Archive

1 **Begin the Oracle Solaris installation program and proceed through the panels until you reach the Specify Media panel.**

For the step-by-step procedures, see either of the following procedures:

- SPARC: "Performing an Installation or Upgrade for UFS File Systems With the Oracle Solaris Installation Program" in *Oracle Solaris 10 1/13 Installation Guide: Basic Installations*
- x86: "Performing an Installation or Upgrade for UFS File Systems With the Oracle Solaris Installation Program" in *Oracle Solaris 10 1/13 Installation Guide: Basic Installations*

2 **Specify the media you are using to install.**

The following table describes the processes for the media choices.

Media Selected	Prompt
DVD or CD	Insert the disc where the flash archive is located.
Network File System	Specify the path to the network file system where the flash archive is located. You can also specify the archive file name.
HTTP	Specify the URL and proxy information that is needed to access the flash archive.
FTP	Specify the FTP server and the path to the flash archive. Specify the user and password information that is required to access the FTP server. Specify any proxy information that is needed to access the FTP server.
Local tape	Specify the local tape device and the position on the tape where the flash archive is located.

If you selected to install an archive from a DVD, CD, or from an NFS server, the Select Flash Archives panel is displayed.

a. **Select one or more flash archives to install.**

The Flash Archives Summary panel is displayed.

b. Confirm the selected archives and click Next.

The Additional Flash Archives panel is displayed.

c. To install an additional flash archive, specify the media where the other archive is located. If you do not want to install additional archives, select None.

3 Click Next to continue the installation.

Follow the steps from one of the following procedures to complete the installation:

- SPARC: "Performing an Installation or Upgrade for UFS File Systems With the Oracle Solaris Installation Program" in *Oracle Solaris 10 1/13 Installation Guide: Basic Installations*
- x86: "Performing an Installation or Upgrade for UFS File Systems With the Oracle Solaris Installation Program" in *Oracle Solaris 10 1/13 Installation Guide: Basic Installations*

Administering Flash Archives

The `flar` command enables you to administer archives. You can split an archive into sections. Those sections can be modified, added to, or deleted, and then merged to create an archive. You can also obtain information about the archive.

⚠️ **Caution –** Do not modify the Archive Files section or you compromise the integrity of the archive.

Splitting a Flash Archive

You can split an archive into sections, which enables you to modify some sections, add new sections, or delete sections. After you have modified the sections, you need to merge the sections to create an new archive. For example, you might want to add a user-defined section or modify the archive identification section. Do not modify the Archive Files section or you compromise the integrity of the archive.

The `flar split` command splits a flash archive into sections. The `flar` command copies each section into a separate file in the current directory or specified directory. The files are named after the sections, for example, the archive cookie is saved in a file that is named `cookie`. You can specify that the `flar split` command save only one section. The syntax of the command is as follows:

`flar split` [`-d` *dir*] [`-u` *section*] [`-f` archive] [`-S` *section*] [`-t` [`-p` *posn*] [`-b` *blocksize*]] *filename*

`-d` *dir*	Retrieves the sections to copy from *dir*, rather than from the current directory.

-u *section*	■ If you use this option, `flar` copies the cookie, identification, archive, and *section* sections. You can specify a single section name or a space-separated list of section names.
	■ If you do *not* use this option, `flar` copies the cookie, identification, and archive sections only.
-f archive	Extracts the archive section into a directory that is named `archive`, rather than placing it in a file with the name `archive`.
-S *section*	Only copies the section that is named *section* from the archive. This section is user defined.

EXAMPLE 4–1   Splitting an Archive

In the following example, `archive1.flar` is split into three files:

■ `cookie` – The first line of the archive, which identifies the version of the archive format. Do not change this identifier.

■ `identification` – A copy of the archive identification section with all keyword-value pairs.

■ `archive` – The archive itself. This file can be compressed.

```
# flar split archive1.flar
```

After the archive is split, you can modify the archive identification section or add a user-defined section. The sections can then be merged to re-create the archive.

Merging Flash Archives

After you have split an archive into sections, you can combine the sections to create a new archive.

The `flar combine` command creates a flash archive from individual sections. The following table describes how the `flar` command handles combining sections.

Conditions	Description
Minimum number of files	Each section is assumed to be in a separate file, the names of which are the section names. These three files must be present:
	■ Archive cookie (`cookie`)
	■ Archive identification (`identification`)
	■ Archive files (`archive`)

Conditions	Description
Archive copy method	If archive is a directory, the contents are archived before including the directory in the combined archive by using the cpio copy utility.
	■ cpio is the default copy method. Individual file sizes cannot be greater than 4 GB.
	■ pax is the copy method to handle large individual files. The flarcreate command with the -L pax option uses the pax utility to create an archive without limitations on individual file sizes. Individual file sizes can be greater than 4 GB.
Compressing an archive	If the archive identification section specifies to compress the archive, flar compresses the contents of the newly combined archive.
Validation	No validation is performed on any of the sections. In particular, no fields in the archive identification section are validated or updated.

The syntax for the flar combine command is as follows:

flar combine [-d *dir*] [-u *section*] [-t [-p *posn*] [-b *blocksize*]] *filename*

-d *dir*   Retrieves the sections to combine from *dir*, rather than from the current directory.

-u *section*

■ If you use this option, flar copies the cookie, identification, archive, and *section* sections. You can specify a single section name or a space-separated list of section names.

■ If you do *not* use this option, flar copies the cookie, identification, and archive sections only.

EXAMPLE 4–2   Merging a Flash Archive

In this example, an archive cookie section, an archive identification section, and an archive files section are combined to become a complete archive. The archive is named newarchive.flar.

```
# flar combine newarchive.flar
```

EXAMPLE 4–3   Merging a Flash Archive and Adding a User-Defined Section

In this example, an archive cookie section, an archive identification section, an archive files section, and a user-defined section are combined to become a complete archive. The archive is named newarchive.flar. The user-defined section content is in the file that is named user-defined, which is in the current directory.

```
# flar combine -u user_defined newarchive.flar
```

Extracting Information From an Archive

Use the flar info command to obtain information about archives you have already created. The syntax of the command is as follows:

flar info [-l] [-k *keyword*] [-t [-p *posn*] [-b *blocksize*]] *filename*

-k *keyword*   Returns only the value of the keyword *keyword*.

-l   Lists all the files in the archive section.

EXAMPLE 4–4   Listing Files in an Archive Section

In this example, the command checks the file structure of the archive named archive3.flar.

```
# flar info -l archive3.flar
aaa
aaa/bbb
aaa/bbb/ccc
aaa/bbb/ccc/ddd
aaa/eee
```

5

Creating and Using a Disaster Recovery Image

You can use the procedures in this chapter to create a flash archive recovery image that can be used to restore a system to "factory fresh" condition.

Recovery Image Procedures

The following procedures provide the simplest instructions for creating a Flash Archive (FLAR) image that can be loaded onto the target system in order to recover from a failed disk drive.

▼ How to Create and Save a FLAR Image

Before You Begin   To perform these procedures, you need access to the following:

- Either the initial boot media, such as the installation CD or DVD, or a netinstall service
- Off-system storage for the FLAR image

1   **Record the partition table of the disk drive that the image is for.**

This step assumes that the replacement disk drive will be the same size and partitioned identically to the original drive.

Use one of the following two methods to obtain information about the partition table on the disk drive.

- **As superuser (root), use the `format` command to print the partition table for the drive that the FLAR image will be taken from.**

    ```
    # format
    ```

    The format command provides the names of the partitions.

    a.   Select the boot drive from the list.

    The first disk in the list is usually the boot drive.

    b. Use the `partition` subcommand to access the Partition menu.

    c. Use the `quit` subcommand to exit from the Partition menu.

    d. Use the `quit` subcommand to exit from the Format application.

    For further information, see the `format(1M)` man page.

■ **As superuser (root), use the `prtvtoc` command to generate the partition information.**

```
# prtvtoc /dev/dsk/c0t0d0s0
```

The `prtvtoc` command provides the size of the partitions by the number of cylinders for each partition.

Save the information to a safe location. You will use this information during the restoration of the system image during recovery.

2 Ensure there is adequate space for the FLAR image where it will be created.

The FLAR archive will require up to 15 GB of space without compression.

```
# df -h /tmp
```

Note – If you do not have sufficient space in /tmp, try a different filesystem, such as /export, instead. If you use the alternate file system, substitute its name for /tmp in the following steps.

3 Shut down and reboot the system into single-user mode.

For instructions on shutting down a system, see "How to Shut Down a Stand-Alone System" in *Oracle Solaris Administration: Basic Administration*.

■ **For SPARC systems, boot as follows:**

```
>OK boot -s
```

For further information, see "How to Boot a System to Run Level S (Single-User Level)" in *Oracle Solaris Administration: Basic Administration*.

■ **For x86 systems, boot using the instructions at "How to Boot a System to Run Level S (Single-User Level)" in** *Oracle Solaris Administration: Basic Administration*.

4 Create the FLAR archive.

In this example, the FLAR image will be stored to a directory under /tmp named FLAR_recovery. The FLAR image will be named newsystem_recovery.flar.

```
# mkdir /FLAR_recovery
# flarcreate -n my_recovery_image -x /FLAR_recovery \
/FLAR_recovery/newsystem_recovery.flar
```

In this example:

- -n my_recovery_image implants a name into the FLAR image. The name should be something unique and meaningful to better identify it as the FLAR image for the system.

- -x /FLAR_recovery causes the /FLAR_recovery directory and its contents to be excluded from the FLAR image because it will not be needed in the recovery image.

Note – By default, the flarcreate command ignores items that are located in swap partitions.

- /FLAR_recovery/newsystem_recovery.flar is the path and file name of the FLAR image. The file name should be something unique and meaningful to better identify it as the FLAR image for the system.

5 Save the FLAR image to a secure off-system location.

The FLAR image must be saved to a local storage device that is not the boot device, or to a remote location across NFS. The storage device, or remote location, must be accessible to the system at recovery time.

For example:

```
# cp /FLAR_recovery/newsystem_recovery.flar \
 /net/my-safe-machine/FLAR_image
```

▼ How to Recover the System Image From a FLAR Image

The recovery process begins as a normal installation using whichever install method you choose. Instead of installing from the boot method, the installer is used to install from the FLAR image.

1 Begin the boot process using one of the following options:

- **Use the initial boot media (installation CD/DVD).**

- **Use the netinstall service.**

   ok> boot net

2 Supply the network, date and time, and password information for the system.

3 When the Specify Media screen appears, select Network File System.

4   In the Specify Network File System Path screen, provide the path to the off-system location of the FLAR image.

For example:

`/net/my-safe-machine/FLAR_image/newsystem_recovery.flar`

The Flash Archive Summary screen is displayed.

5   Verify the file information.

- If the file information is correct, click Next.

- If the file information is incorrect, click Deselect All Archives, and you will be able to provide the correct information.

6   Specify the FLAR image location.

From the example, the location would be the following.

`my-safe-machine:/FLAR_image/newsystem_recovery.flar`

The Disk Selection screen is displayed.

7   Select the disk where the FLAR image is to be installed.

8   Choose not to preserve existing data.

The File System and Disk Layout screen is displayed.

9   Select Customize to edit the disk slices to input the values of the disk partition table from the original disk.

The partition table corresponds to each slice on the disk. Partition 0 from the partition table maps to Slice 0 (`s0`) on the hard drive.

- The slice sizes can be viewed in Cylinders to better match the output from the partition table. Select Cyl in the Partition form to view the form by cylinders.

- Do not change the size of Slice 2. It must span the entire disk regardless of the space being allocated.

- To get the Start and Size values for the Partition form, use the partition information that you recorded earlier when you ran the `prtvtoc` command. To get the value for Start in the Partition form, divide the First Sector value by the Sectors/Cylinder value , both found in the `prtvtoc` command output. The Size value in the Partition form is found by dividing the Sector Count by the Sectors/Cylinder value, information also provided by the `prtvtoc` command output.

- If the replacement disk has more storage space than the original disk, then it can be partitioned to use the space available. However, at least as much space for each partition must be allocated as was allocated on the original disk.

After the system reboots, the recovery is now complete.

10   (Optional) Rebuilding the Device Trees

These recovery instructions assume that none of the hardware components have been added, removed or moved between the time that the recovery image was created and the time that a recovery is performed. If, however, a system has been recovered after hardware has been changed, the device trees (`/dev` and `/devices`) might need to be updated. This update can be done using either a reconfiguration reboot of the system or by using the `devfsadm` command.

To rebuild the device trees, as a root-level user, use the `devfsadm` command as follows.

```
# devfsadm -C
```

Additional Resources

- *Using Flash Archive in the Solaris Operating System for Disaster Recovery*
- "Installing a ZFS Root File System (Oracle Solaris Flash Archive Installation)" in *Oracle Solaris ZFS Administration Guide*
- *Oracle Solaris 10 1/13 Installation Guide: Network-Based Installations*
- *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*

6

Flash Archive (Reference)

This chapter provides a description of Flash Archive sections, keywords, and keyword values. Also, the chapter describes the `flar` command options.

For limitations when creating or installing a flash archive, see Table 2–1.

Flash Archive Section Descriptions

Each flash archive is grouped into sections. Some sections are generated by the Flash Archive software and need no input from you. Some sections require input or optionally allow you to add information. The following table describes each section.

TABLE 6–1 Flash Archive Sections

Section Name	Description	Required by Archive?	Requires Input From User?
Archive cookie	The first section contains a cookie that identifies the file as a flash archive. The deployment code uses the cookie for identification and validation purposes. The cookie must be present for an archive to be valid.	Yes	No

TABLE 6–1 Flash Archive Sections   *(Continued)*

Section Name	Description	Required by Archive?	Requires Input From User?
Archive identification	The second section contains keywords with values that provide identification information about the archive. The software generates some information such as the following: ■ The archive ID number ■ The method of archival such as `cpio` ■ The creation date by default You are required to specify a name for the flash archive. Other information that you can specify about the archive includes the following: ■ The author of the archive ■ The date that the archive was created ■ The name of the master system that you used to create the archive For a list of keywords that describe the archive, see "Keywords for the Archive Identification Section" on page 65.	Yes	Content is generated by both user and the software
Manifest	A section of a flash archive that is used to validate a clone system. The manifest section lists the files on a system to be retained, added to, or deleted from the clone system. The installation fails if the files do not match the expected file set. This section is informational only. The section lists the files in an internal format and cannot be used for scripting. You can exclude this section by creating the differential archive with the `flarcreate -M` option. Because no validation of the archive occurs, excluding this section is not recommended.	No	No
Predeployment, Postdeployment, Reboot	This section contains internal information that the Flash Archive software uses before and after installing an OS image. Any customization scripts that you have provided are stored in this section.	Yes	No
Summary	This section contains messages about the archive creation and records the activities of predeployment scripts.	Yes	Content is generated by both user and the software
User-defined	This section follows the archive-identification section. The archive can contain zero or more user-defined sections. These sections are not processed by the archive extraction code. These sections are retrieved separately and can be used for content descriptions.	No	Yes
Archive files	The archive files section contains the files that have been gathered from the master system in binary data. This section begins with `section_begin=archive`, but it does not have an ending section boundary.	Yes	No

Flash Archive Keywords

Flash Archive keywords are like JumpStart keywords. They define elements of the installation. Each keyword is a command that controls one aspect of how the Flash Archive software installs the software on a clone system.

Use the following guidelines to format keywords and values:

- Keywords and values are separated by a single equal sign with only one pair per line
- Keywords are case insensitive
- Individual lines can be any length

General Keywords

Each Flash Archive section is defined by the `section_begin` and `section_end` keywords. For example, the archive files section includes a `section_begin` keyword, though with a different value. User-defined archive sections are delimited by `section_begin` and `section_end` keywords, with values appropriate to each section. The values for the `section_begin` and `section_end` keywords are described in the following table.

TABLE 6–2   Values for section_begin and section_end Keywords

Archive Section	Value for `section_begin` and `section_end` keywords
Archive cookie	`cookie` – This section is not delimited by the `section-begin` and `section-end` keywords.
Archive identification	`identification`
User-defined sections	*section-name* – An example of a `section-name` keyword is `X-user_section_1`.
Archive files	`archive`

Keywords for the Archive Identification Section

The following tables describe the keywords for use in the archive identification section and the values you can define for them. The required general keywords are described in Table 6–2.

The following keywords, used in the archive-identification section, describe the contents of the archive files section.

TABLE 6–3   Archive Identification Section Keywords: Contents of Archive Files Section

Keywords	Value Definitions	Value	Required
archive_id	This keyword uniquely describes the contents of the archive. This value is used by the installation software only to validate the contents of the archive during archive installation. If the keyword is not present, no integrity check is performed. For example, the archive_id keyword might be FlAsH-ARcHive-2.0.	Text	No
files_archived_method	This keyword describes the archive method that is used in the files section. ■ If this keyword is not present, the files section is assumed to be in cpio format with ASCII headers. This format is the cpio -c option. ■ If this keyword is present, it has one of the following values: ■ cpio – The archive format in the files section is cpio with ASCII headers. ■ pax – The archive format in the files section is pax with extended tar interchange format. The pax utility enables archiving and extracting files that are greater than 4 GB. If the files_compressed_method is present, the compression method is applied to the archive file that is created by the archive method.	Text	No
files_archived_size	This keyword value is the size of the archived files section in bytes.	Numeric	No
files_compress_method	This keyword describes the compression algorithm that is used on the files section. ■ If the keyword is present, it can have one of the following values: ■ none – The archive file section is not compressed. ■ compress – The file section is compressed by using the compress command. ■ If this keyword is not present, the archive files section is assumed to be uncompressed. The compression method that is indicated by this keyword is applied to the archive file created by the archive method indicated by the files_archived_method keyword.	Text	No
files_unarchived_size	This keyword defines the cumulative size in bytes of the extracted archive. The value is used for file-system size verification.	Numeric	No

The following keywords provide descriptive information about the entire archive. These keywords are generally used to assist you in archive selection and to aid in archive management. These keywords are all optional and are used to help you to distinguish between individual archives. You use options for the flarcreate command to include these keywords. For details, see Example 3–9.

TABLE 6–4 Archive Identification Section Keywords: User Describes the Archive

Keywords	Value Definitions	Value	Required
creation_date	This keyword value is a textual timestamp that represents the time that you created the archive. ■ You can use the flarcreate command with the -i option to create the date. ■ If you do not specify a creation date with the flarcreate command, the default date is set in Greenwich mean time (GMT). ■ The value must be in ISO-8601 complete basic calendar format without the time designator (ISO-8601,§5.4.1(a)). The format is *CCYYMMDDhhmmss*. For example, 20100131221409 represents January 31, 2010, 10:14:09 p.m.	Text	No
creation_master	This keyword value is the name of the master system you used to create the archive. You can use the flarcreate -m option to create this value. If you do not specify a value, the value is taken from the uname -n command.	Text	No
content_name	This keyword identifies the archive. The value is generated from the flarcreate -n option. Follow these guidelines when you create this value: ■ The descriptive name can be no longer than 256 characters. ■ The description should contain the function and purpose of the archive.	Text	Yes
content_type	This keyword value specifies a category for the archive. You use the flarcreate -T option to generate the value.	Text	No
content_description	The keyword value describes the contents of the archive. The value of this keyword has no length limit. You use the flarcreate -E option to create this value.	Text	No
content_author	This keyword value identifies the creator of the archive. You use the flarcreate -a option to create this value. Suggested values include the full name of the creator and the creator's email address.	Text	No

TABLE 6–4 Archive Identification Section Keywords: User Describes the Archive *(Continued)*

Keywords	Value Definitions	Value	Required
content_architectures	This keyword value is a comma-separated list of the kernel architectures that the archive supports. ■ If the keyword is present, the installation software validates the kernel architecture of the clone system against the list of architectures that the archive supports. The installation fails if the archive does not support the kernel architecture of the clone system. ■ If the keyword is not present, the installation software does not validate the architecture of the clone system.	Text list	No

The following keywords also describe the entire archive. By default, the values are filled in by uname when the flash archive is created. If you create a flash archive in which the root directory is not /, the archive software inserts the string UNKNOWN for the keywords. The exceptions are the creation_node, creation_release, and creation_os_name keywords.

■ For creation_node, the software uses the contents of the nodename file.

■ For creation_release and creation_os_name, the software attempts to use the contents of root directory /var/sadm/system/admin/INST_RELEASE. If the software is unsuccessful in reading this file, it assigns the value UNKNOWN.

Regardless of their sources, you cannot override the values of these keywords.

TABLE 6–5 Archive Identification Section Keywords: Software Describes the Archive

Keyword	Return Value Source
creation_node	uname -n
creation_hardware_class	uname -m
creation_platform	uname -i
creation_processor	uname -p
creation_release	uname -r
creation_os_name	uname -s
creation_os_version	uname -v

User-Defined Section Keywords

In addition to the keywords that are defined by the flash archive, you can define other keywords. The flash archive ignores user-defined keywords, but you can provide scripts or programs that process the archive identification section and use user-defined keywords. Use the following format when creating user-defined keywords:

- Begin the keyword name with an X.
- Create the keyword with any characters other than linefeeds, equal signs, and null characters.
- Suggested naming conventions for user-defined keywords include the underscore-delimited descriptive method used for the predefined keywords. Another convention is a federated convention similar to the naming of Java packages.

For example, X-department is a valid name for a user-defined keyword.

For an example of using options to include user-defined keywords in the archive identification section, see Example 3–9.

Flash Archive `flar` Command

Use the Flash Archive `flar` command to create a flash archive and administer the archive.

`flar` Command

You can use the `flar` command to perform the following actions:

- `flarcreate` creates an archive
- `flar combine` merges two archives
- `flar split` breaks an archive into sections
- `flar info` checks the structure of an archive

Use the `flarcreate` command to create a flash archive from a master system. You can use this command when the master system is running in multiuser mode or single-user mode. You can also use `flarcreate` when the master system is booted from the following media.

- Oracle Solaris Operating System DVD

Note – Starting with the Oracle Solaris 10 9/10 release, only a DVD is provided. Oracle Solaris Software CDs are no longer provided.

- Oracle Solaris Software - 1 CD

- A Oracle Solaris network installation image of the DVD or CDs.

The master system should be in as stable a state as possible when you create a flash archive.

Note – You can create a flash archive by using either of these command options:

- As two words: `flar` with the `create` subcommand
- As one word: `flarcreate`

The syntax of the command is as follows:

`flarcreate -n` *archive-name* `[-R` *root*`] [-A` *unchanged-master-image-dir*`]`
`[-H][-I][-M][[-S]-c][-t [-p` *posn*`] [-b` *blocksize*`]][-i` *date*`][-u` *section* `...][-m`
master`][-f [`*list-filename*`| -] [-F][-a` *author*`][-e` *descr*`|-E` *descr-file*`] [-L pax] [-T`
type`][-U` *key=val* `...][-x` *exclude-dir/filename*`] [-y` *include-dir/filename*`] [-z`
list-filename`] [-X` *list-filename*`]`*path/filename*

`flar combine [-d` *dir*`] [-u` *section...*`] [-t [-p` *posn*`]` *path/filename*

`flar split [-d` *dir*`] [-u` *section...*`] [-f] [-S` *section*`] [-t [-p` *posn*`]` *path/filename*

`flar info [-l] [-k` *keyword*`] [-t [-p` *posn*`]` *path/filename*

In these command lines, *path* is the directory in which you want the archive file to be saved.
filename is the name of the archive file. If you do not specify a path, `flarcreate` saves the
archive file in the current directory.

TABLE 6–6   Command-Line Options for the `flar` Command

Option	Description
Required Options	
-n *archive-name*	The value of this flag is the name of the archive. The *archive-name* you specify is the value of the `content_name` keyword.
Option for Compression	
-c	Compresses the archive by using `compress(1)`.
Options for Directories and Sizes	
-R *root*	Creates the archive from the file system tree that begins at the file system specified by *root*. If you do not specify this option, `flarcreate` creates an archive from a file system that begins at the root (`/`) file system.
-S	Omits sizing information in the archive.
-H	Does not generate the hash identifier.

TABLE 6–6 Command-Line Options for the `flar` Command   *(Continued)*

Option	Description
Options for Creating a Differential Archive	
-A *unchanged-master-image-dir*	Creates a differential archive by comparing a new system image with the image that is specified by the *unchanged-master-image-dir* argument. By default, the new system image is root (`/`). You can change the default with the -R option. *unchanged-master-image-dir* is a directory where the unchanged master system image is stored or mounted through UFS, NFS, or `lumount`. You can modify the effects of file selection for a differential archive by using the options for contents selection described in the next section of the table.
-M	Excludes the manifest file. When you use this option, no validation occurs on the differential archive. When creating a differential archive, `flarcreate` creates a long list of the files in the system that are unchanged, are changed, and are to be deleted from the archive. This list is stored in the manifest section of the archive. When the differential archive is deployed, the software uses this list to perform a file-by-file check, ensuring the integrity of the clone system. Use of this option avoids such a check and saves the space that is used by the manifest section in a differential archive. However, you must consider the savings in time and disk space against the loss of an integrity check upon installation. Because no validation occurs, avoid using this option.
Options for Contents Selection	
Caution – Use the `flarcreate` file-exclusion options with caution. If you exclude some directories, others that you were unaware of might be left in the archive, such as system configuration files. The system would then be inconsistent and the installation would not work. Excluding directories and files is best used with data that can easily be removed without disrupting the system, such as large data files.	
-y *include-dir/filename*	Adds to the archive those files and directories that are specified on the command line. This option is used when you have excluded a directory, but want to restore individual subdirectories or files. *include-dir/filename* is the name of the subdirectory or file to be included.
-f *list-filename*	Adds files and directories from a list to the archive. *list-filename* is the full path to a file that contains a list. The contents of the file are added to the file list unless -F is specified. - The *list-filename* file must contain one file per line. - If you specify a file system with -R *root*, the path to each file must be relative to the alternate `root` directory or an absolute path. - If *filename* is "-", `flarcreate` reads standard input as the list of files. When you use the value "-", the archive size is not calculated.
-F	Uses only the files in -f *list-filename* to create the archive. This option makes the -f *list-filename* the absolute list, rather than a list that is appended to the normal file list.

TABLE 6–6   Command-Line Options for the `flar` Command   *(Continued)*

Option	Description
-x *exclude-dir/filename*	Excludes files and directories from the archive. These files and directories are specified at the command line. You can use multiple instances of this option to exclude more than one file or directory.
	exclude-dir/filename is the name of the directory or file to be excluded.
-X *list-filename*	Excludes a list of files or directories from the archive.
	list-filename is the full path to a file that contains the list. - The *list-filename* file must contain one file per line. - If you specify a file system with -R *root*, the path to each file must be relative to the alternate `root` directory or an absolute path. - If *list-filename* is "-", `flarcreate` reads standard input as the list of files. When you use the value "-", the archive size is not calculated.
-z *list-filename*	Excludes or includes a list of files or directories from the archive. Each file or directory in the list is noted with a plus "+" or minus "-". A plus indicates an included file or directory and the minus indicates an excluded file or directory.
	list-filename is the full path to a file that contains the list. - The *list-filename* file must contain one file per line. - If you specify a file system with -R *root*, the path to each file must be relative to the alternate `root` directory or an absolute path.
-I	Override the integrity check. To prevent you from excluding important system files from an archive, `flarcreate` runs an integrity check. This check examines all files that are registered in a system package database and stops archive creation if any of them are excluded. Use of this option overrides this integrity check. Therefore, avoid the use of the -I option.
Options for Splitting and Merging Archives	
-d *dir*	Retrieves the sections to copy from *dir*, rather than from the current directory.
-u *section*	- If you use this option, `flar` copies the cookie, identification, archive, and *section* sections. You can specify a single section name or a space-separated list of section names. - If you do *not* use this option, `flar` copies the cookie, identification, and archive sections only.
-f *archive*	Extracts the archive section into a directory that is named `archive`, rather than placing it in a file with the name `archive`. Used for splitting an archive.
-S *section*	Only copies the section that is named *section* from the archive. This section is user defined. Used for splitting an archive.
Option Used To Copy Files (Archive)	

TABLE 6–6 Command-Line Options for the `flar` Command *(Continued)*

Option	Description
`-L pax`	The `cpio` utility is the default copy method. If you have large individual files, the `-L pax` option uses the pax utility to create an archive without limitations on individual file sizes. Individual file sizes can be greater than 4 GB.
Options Used With User-Defined Sections	
`-u` *section*	Includes *section* as a user-defined section. To include more than one user-defined section, *section* must be a space-separated list of section names.
`-d` *dir*	Retrieves the section file that is specified with `-u` from *dir*.
Options Used With Tape Archives	
`-t`	Creates an archive on a tape device. The *filename* argument is the name of the tape device.
`-p` *posn*	Use only with the `-t` option. Specifies the position on the tape device for `flarcreate` to store the archive. If you do not use this option, `flarcreate` places the archive at the current position of the tape.
`-b` *blocksize*	Specifies the block size `flarcreate` uses when creating the archive. If you do not specify a block size, `flarcreate` uses the default block size of 64 KB.
Options for Archive Identification	
These keywords and values appear in the archive identification section of the archive.	
`-U` *key=val*	Includes user-defined keywords and values in the archive identification section.
`-i` *date*	Uses *date* as the value for the `creation_date` keyword. If you do not specify a date, `flarcreate` uses the current system time and date.
`-m` *master*	Uses *master* as the name of the master system on which you created the archive. *master* is the value for the `creation_master` keyword. If you do not specify *master*, `flarcreate` uses the system name that is reported by `uname -n`.
`-e` *descr*	Uses *descr* for the value of the `content_description` keyword. You cannot use this option when you use the `-E` option.
`-E` *descr-file*	Retrieves the value for the `content_description` keyword from the *descr-file* file. You cannot use this option when you use the `-e` option.
`-a` *author*	Uses *author* as the author name in the archive identification section. *author* is the value for the `content_author` keyword. If you do not specify an author, `flarcreate` does not include the `content_author` keyword in the archive identification section.

TABLE 6–6 Command-Line Options for the `flar` Command   *(Continued)*

Option	Description
-T *type*	Uses *type* as the value for the `content_type` keyword. *type* is user defined. If you do not specify a type, `flarcreate` does not include the `content_type` keyword.

Glossary

archive   A file that contains a collection of files that were copied from a master system. The file also contains identification information about the archive, such as a name and the date that you created the archive. After you install an archive on a system, the system contains the exact configuration of the master system.

An archive could be a differential archive, which is a flash archive that contains only the differences between two system images, an unchanged master image and an updated master image. The differential archive contains files to be retained, modified, or deleted from the clone system. A differential update changes only the files specified and is restricted to systems that contain software consistent with the unchanged master image.

begin script   A user-defined Bourne shell script, specified within the `rules` file, that performs tasks before the Oracle Solaris software is installed on the system. You can use begin scripts only with JumpStart installations.

boot   To load the system software into memory and start it.

boot archive   **x86 only**: A boot archive is a collection of critical files that is used to boot the Oracle Solaris OS. These files are needed during system startup before the root (`/`) file system is mounted. Two boot archives are maintained on a system:

- The boot archive that is used to boot the Oracle Solaris OS on a system. This boot archive is sometimes called the primary boot archive.

- The boot archive that is used for recovery when the primary boot archive is damaged. This boot archive starts the system without mounting the root (`/`) file system. On the GRUB menu, this boot archive is called failsafe. The archive's essential purpose is to regenerate the primary boot archive, which is usually used to boot the system.

boot environment   A collection of mandatory file systems (disk slices and mount points) that are critical to the operation of the Oracle Solaris OS. These disk slices might be on the same disk or distributed across multiple disks.

The active boot environment is the one that is currently booted. Exactly one active boot environment can be booted. An inactive boot environment is not currently booted, but can be in a state of waiting for activation on the next reboot.

clone system   A system that you install by using a flash archive. The clone system has the same installation configuration as the master system.

differential archive   A flash archive that contains only the differences between two system images, an unchanged master image and an updated master image. The differential archive contains files to be retained, modified, or deleted from the clone system. A differential update changes only the files that are specified and is restricted to systems that contain software consistent with the unchanged master image.

End User Oracle Solaris software Group	A software group that contains the Core Software Group plus the recommended software for an end user, including the Common Desktop Environment (CDE) and DeskSet software.
Entire Oracle Solaris software Group	A software group that contains the entire Solaris release.
Entire Oracle Solaris software Group Plus OEM Support	A software group that contains the entire Oracle Solaris release plus additional hardware support for OEMs. This software group is recommended when installing Oracle Solaris software on SPARC based servers. In order for a Flash archive to be installed on different system types, the Entire Plus OEM distribution needs to be installed on the master system.
/etc directory	A directory that contains critical system configuration files and maintenance commands.
/export file system	A file system on an OS server that is shared with other systems on a network. For example, the /export file system can contain the root (/) file system and swap space for diskless clients and the home directories for users on the network. Diskless clients rely on the /export file system on an OS server to boot and run.
failsafe boot archive	**x86 only**: A boot archive that is used for recovery when the primary boot archive is damaged. This boot archive starts the system without mounting the root (/) file system. This boot archive is called failsafe on the GRUB menu. The archive's essential purpose is to regenerate the primary boot archive, which is usually used to boot the system. See *boot archive*.
file server	A server that provides the software and file storage for systems on a network.
file system	In the SunOS operating system, a tree-structured network of files and directories that you can access.
finish script	A user-defined Bourne shell script, specified within the rules file, that performs tasks after the Oracle Solaris software is installed on the system but before the system reboots. You use finish scripts with JumpStart installations.
Flash archive	An Oracle Solaris installation feature that enables you to create an archive of the files on a system, called the *master system*. You can then use the archive to install other systems, making the other systems identical in their configuration to the master system. See also *archive.*
format	To put data into a structure or divide a disk into sectors for receiving data.
initial installation	An installation that overwrites the currently running software or initializes a blank disk.
	An initial installation of the Oracle Solaris OS overwrites the system's disk or disks with the new version of the Oracle Solaris OS. If your system is not running the Oracle Solaris OS, you must perform an initial installation. If your system is running an upgradable version of the Oracle Solaris OS, an initial installation overwrites the disk and does not preserve the OS or local modifications.
install server	A server that provides the Oracle Solaris DVD or CD images from which other systems on a network can install Oracle Solaris (also called a *media server*). You can create an install server by copying the Oracle Solaris DVD or CD images to the server's hard disk.
JumpStart	A type of installation in which the Oracle Solaris software is automatically installed on a system that is based on a user-defined profile. You can create customized profiles for different types of users and systems.

JumpStart installation	A type of installation in which the Oracle Solaris software is automatically installed on a system by using the factory-installed JumpStart software.
Live Upgrade	An upgrade method that enables a duplicate boot environment to be upgraded while the active boot environment is still running, thus eliminating downtime of the production environment.
manifest section	A section of a Flash Archive that is used to validate a clone system. The manifest section lists the files on a system to be retained, added to, or deleted from the clone system. This section is informational only. The section lists the files in an internal format and cannot be used for scripting.
master system	A system that you use to create a Flash Archive archive. The system configuration is saved in the archive.
mount	The process of accessing a directory from a disk that is attached to a machine that is making the mount request or a remote disk on a network. To mount a file system, you need a mount point on the local system and the name of the file system to be mounted (for example, /usr).
mount point	A workstation directory to which you mount a file system that exists on a remote machine.
network installation	A way to install software over the network from a system with a CD-ROM or DVD-ROM drive to a system without a CD-ROM or DVD-ROM drive. Network installations require a *name server* and an *install server*.
non-global zone	A virtualized operating system environment created within a single instance of the Oracle Solaris Operating System. One or more applications can run in a non-global zone without interacting with the rest of the system. Non-global zones are also called zones. See also *Oracle Solaris Zones* and *global zone*.
/opt file system	A file system that contains the mount points for third-party and unbundled software.
Oracle Solaris DVD or CD images	The Oracle Solaris software that is installed on a system, which you can access on the Oracle Solaris DVDs or CDs or an install server's hard disk to which you have copied the Oracle Solaris DVD or CD images.
Oracle Solaris installation program	A graphical user interface (GUI) or command-line interface (CLI) installation program that uses wizard panels to guide you step-by-step through installing the Oracle Solaris software and third-party software.
profile	A text file that defines how to install the Oracle Solaris software when using the custom JumpStart method. For example, a profile defines which software group to install. Every rule specifies a profile that defines how a system is to be installed when the rule is matched. You usually create a different profile for every rule. However, the same profile can be used in more than one rule. See also *rules file*.
root	The top level of a hierarchy of items. Root is the one item from which all other items are descended. See *root directory* or *root (/) file system*.
root (/) file system	The top-level file system from which all other file systems stem. The root (/) file system is the base on which all other file systems are mounted, and is never unmounted. The root (/) file system contains the directories and files critical for system operation, such as the kernel, device drivers, and the programs that are used to start (boot) a system.
root directory	The top-level directory from which all other directories stem.
server	A network device that manages resources and supplies services to a client.

slice	The unit into which the disk space is divided by the software.
software group	A logical grouping of the Oracle Solaris software (clusters and packages). During a Oracle Solaris installation, you can install one of the following software groups: Core, End User Oracle Solaris Software, Developer Oracle Solaris software, or Entire Oracle Solaris software, and for SPARC systems only, Entire Oracle Solaris software Group Plus OEM Support.
superuser	A special user who has privileges to perform all administrative tasks on the system. The superuser has the ability to read and write to any file, run all programs, and send kill signals to any process.
swap space	A slice or file that temporarily holds the contents of a memory area till it can be reloaded in memory. Also called the /swap or swap volume.
unmount	The process of removing access to a directory on a disk that is attached to a machine or to a remote disk on a network.
update	An installation, or to perform an installation, on a system that changes software that is of the same type. Unlike an upgrade, an update might downgrade the system. Unlike an initial installation, software of the same type that is being installed must be present before an update can occur.
upgrade	An installation that merges files with existing files and preserves modifications where possible.
	An upgrade of the Oracle Solaris OS merges the new version of the Oracle Solaris OS with the existing files on the system's disk or disks. An upgrade saves as many modifications as possible that you have made to the previous version of the Oracle Solaris OS.
upgrade option	An option that is presented by the Oracle Solaris installation program. The upgrade procedure merges the new version of Oracle Solaris with existing files on your disk or disks. An upgrade also saves as many local modifications as possible since the last time Oracle Solaris was installed.
/usr file system	A file system on a standalone system or server that contains many of the standard UNIX programs. Sharing the large /usr file system with a server rather than maintaining a local copy minimizes the overall disk space that is required to install and run the Oracle Solaris software on a system.
/var file system	A file system or directory (on standalone systems) that contains system files that are likely to change or grow over the life of the system. These files include system logs, vi files, mail files, and UUCP files.
ZFS	A file system using storage pools to manage physical storage.
zone	See *non-global zone*

Index