



Provisioning in Replicated, Mission-Critical Environments

*Jay Daliparthi, N1 and Management Solutions
Marketing*

James Falkner, Solaris Software

Sun BluePrints™ OnLine—March 2003



<http://www.sun.com/blueprints>

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95045 U.S.A.
650 960-1300

Part No. 817-2014-10
Revision 1.0, 2/18/03
Edition: March 2003

Copyright 2003 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, California 95045 U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the United States and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Sun BluePrints, Solaris Web Start, Solaris JumpStart, Solaris Flash, Solaris Live Upgrade, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the US and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

U.S. Government Rights—Commercial use. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, Californie 95045 Etats-Unis. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque enregistrée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company Ltd.

Sun, Sun Microsystems, le Sun logo, Sun BluePrints, Solaris Web Start, Solaris JumpStart, Solaris Flash, Solaris Live Upgrade, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REpondre A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



Please
Recycle



Adobe PostScript

Provisioning in Replicated, Mission-Critical Environments

Installing, upgrading, and maintaining operating environment software and unbundled products in a mission-critical environment can cause unwanted downtime if not managed correctly. This article introduces some of the latest technologies that enable rapid Solaris™ Operating Environment (OE) installation, software upgrades, and patch management in high availability environments. These tools and techniques enable system administrators to perform quick, consistent, controlled, and reproducible Solaris OE installations and upgrades.

This article addresses the following topics:

- “Solaris™ Flash and Solaris™ Live Upgrade Technology” on page 1 provides a brief overview of Solaris Flash and Live Upgrade.
- “Solaris Flash Software Overview” on page 2 provides an overview of using Solaris Flash software in mission-critical environments.
- “Solaris Live Upgrade Software Overview” on page 7 provides an overview of using Solaris Live Upgrade software and how this software can be used to install a Flash archive.

This article assumes that you have a basic understanding of the Solaris JumpStart™ and Solaris™ Web Start technology framework and services.

Solaris™ Flash and Solaris™ Live Upgrade Technology

Solaris Flash software provides a mechanism to create a Flash archive, which is a complete system snapshot of the Solaris OE and all other installed software from a reference system. This archive can then be replicated on multiple machines (clones).

Flash archives can be deployed by a network server or local media with custom Solaris JumpStart software or the Solaris Web Start software. John S. Howard (Howard and Noordergraaf, 2001) explains

Flash is a powerful tool for deploying complex server configurations. Complex system architecture only needs to be implemented and configured once and then a Flash archive created; all subsequent system installations from that Flash archive will be identical. For systems such as database servers, the Flash archive should be created before the system goes into production.

Solaris Live Upgrade software enables Solaris systems to perform uninterrupted while the system administrator installs a Flash archive or upgrades to a new Solaris 8 or 9 OE. Solaris Live Upgrade software allows creating, managing, upgrading, comparing, and activating multiple boot environments. Solaris Live Upgrade also supports fallback if an upgrade failure occurs. Solaris Live Upgrade software reduces system downtime and risk associated with an operating system upgrade or patch maintenance.

Solaris Flash Software Overview

The Solaris Flash software allows system administrators to create a snapshot image of a complete system including the Solaris OE, the applications, and the system configuration into a new Flash archive format. This reference system on which the archive is created is referred to as a master system. Using the Flash archive, administrators can replicate reference server configurations onto multiple clone, or client, machines.

The three steps used to create, archive, and deploy a Flash archive are described in the following sections.

- “Configuring the Master System” on page 2
- “Creating a Flash Archive” on page 4
- “Deploying a Flash Archive” on page 5

Configuring the Master System

The first step is to install a master machine with the configuration that you would like to have on your clone machines. For example, if you want to install ten cache servers, then first you need to install a master system and configure it to work as a cache server. To install a master system, you can use any of the Solaris installation

methods, such as a Solaris JumpStart software installation or interactive installation from a CD to install the Solaris OE. After installing the Solaris OE, any software can be added or deleted and configurations can be modified as necessary.

This first step is the most time-consuming, because it requires the administrator to install one or more software packages, and then configure the system to perform as it would in production. However, this step is the most critical—once the software load is defined and archived, it is considered to be stable and complete. It is important to spend enough resources to achieve this goal. Once a stable and complete system is configured, it need not be configured again. The modifications made in this step are archived as part of the Flash archive, and remain frozen until extracted at a later date.

FIGURE 1 provides a diagram of configuring a master system.

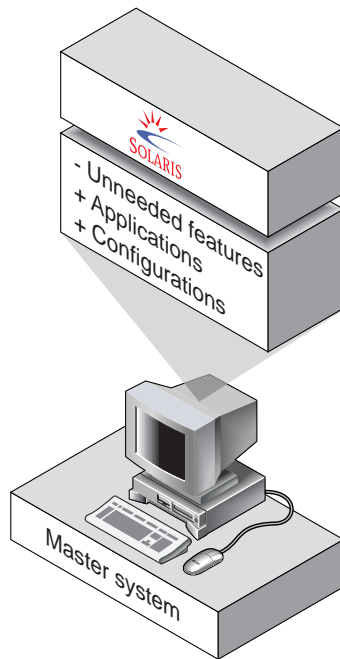


FIGURE 1 Configuring a Master System

When the archive is later extracted to multiple machines, the resource savings become apparent. For example, a horizontally-scaled web server is a typical application of Solaris Flash. In this scenario, a machine is installed with the Solaris OE. Next, various entries in `/etc/inetd.conf` are removed to harden the machine against any security breaches. Then a web server such as Apache is installed and configured to provide optimum performance and security. Finally, the machine is archived using the Solaris Flash, `flar(1M)` command (`flar create`). This archive can then be deployed across tens or even hundreds of machines to provide a quick,

virtually bulletproof farm of web servers. Each machine need not be individually configured; the required configuration is created “at the factory” when the master machine is configured.

Creating a Flash Archive

Once a master system has been configured to satisfaction, a snapshot is taken. This second step assumes the master system is complete and functional. In this process, the content of the master system is archived into a Flash archive. The Flash archive is created with the `flar(1M)` command. Note that the Flash archive copies all software and files on the master system unless certain files are specifically excluded. For example, if a flash archive is created from a web server, then that archive contains all the software and files installed on the master web server.

When the Flash archive is created with the `flar(1M)` command, the archival process includes not only the software and files on the system, but also includes any identification information (such as the size, date of creation, or MD5 integrity footprint) that can be used to later identify the archive.

The archival process can be time-consuming depending on the size of the system to be archived. However, this process is completely hands-off. The output of this process is the Flash archive which can then be stored on any file system or media such as an NFS, FTP, or HTTP server, tape, CD-ROM, or DVD. Typical sizes for a Flash archive containing the Solaris OE range from 100 Mbytes up to 1 Gbyte. Compression (`flar create -c`) can be used to minimize the resulting footprint. The Flash archives can then be installed over the network using custom Solaris JumpStart or the Solaris Web Start software for interactive installation.

FIGURE 2 provides a diagram of creating a Flash archive.

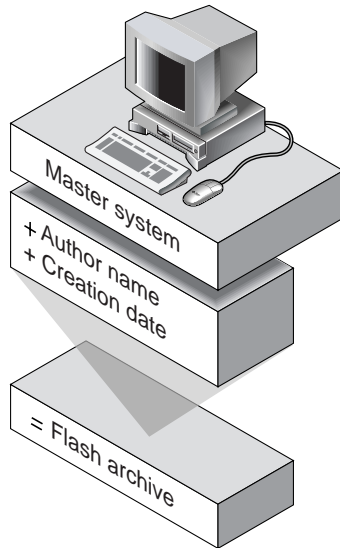


FIGURE 2 Creating a Flash Archive

There are important considerations when creating Flash archives. The master and clone machines preferably should have the same kernel architecture. For example, Flash archives created from a master server that has a `sun4u` architecture can only be used to install other servers with a `sun4u` architecture. However, post-install JumpStart scripts can be written for any necessary modifications to configuration files such as `/etc/nsswitch.conf`. These modifications allow for any differences in kernel architecture or device differences between the master machine and installation client. Separate Flash archives need to be created for both SPARC® and Intel architecture machines if you need to deploy on both platforms.

Deploying a Flash Archive

The final step is to deploy the archive. Several methods of deployment can be used. Traditional Solaris installation technologies can be used, such as Solaris interactive installation or custom JumpStart software, with the new `archive_location` keyword. In addition, Live Upgrade (described in “Solaris Live Upgrade Software Overview” on page 7) can also be used to deploy Flash archives.

FIGURE 3 provides a diagram of deploying a Flash archive.

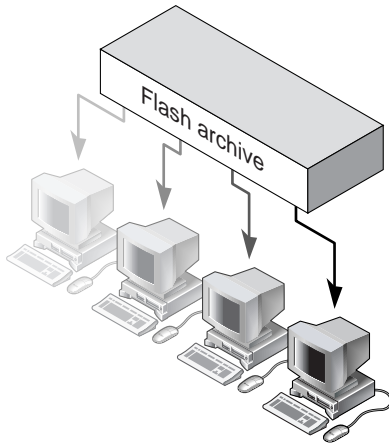


FIGURE 3 Deploying the Flash Archive

Since the majority of system configuration is complete, the performance of deploying a Flash archive is primarily limited by the available bandwidth between the location of the stored archive(s) and clone machines. The overhead associated with a traditional package-based install is eliminated using Flash archives. It is important to note that not everything on the clone machines is identical—for example, each clone machine will have a unique IP address. These machine-specific configurations are minimal and are accomplished with the same techniques found in traditional installations.

Installation with Solaris Flash is faster compared to traditional package-based installation, where each package is installed individually. John S. Howard (Howard, 2001) observes, “A traditional JumpStart software installation using packages may become time consuming as `pkgadd(1M)` needs to update the installed package database as well as relocate files.”

Solaris Flash technology can also be used to store existing server configurations in the Flash archive format as a backup. If a failure occurs, software can be reinstalled rapidly using Flash archives. Solaris Flash is also ideal for the replication and deployment of server farms. Solaris Flash technology provides the ability to have consistent, controlled, reliable, and reproducible Solaris OE installations in mission-critical operations such as those in data centers.

Solaris Flash technology is included in the Solaris 8 4/01 release. Additional information on Solaris Flash is also available from the *Solaris 8 Installation Guide* and in the November, 2001 Sun BluePrints™ OnLine article “Web Start Flash” by John S. Howard and Alex Noordergraaf.

Solaris Live Upgrade Software Overview

The process of performing an upgrade to the Solaris 8 or 9 OE and other system software or other routine patch maintenance can impact availability of systems for normal operations. Systems running mission-critical operations, such as those in data centers, cannot afford system down-time for patch application or software upgrades. Solaris Live Upgrade software allows upgrading Solaris 2.6, Solaris 7, 8, or Solaris 9 OE or general maintenance on an alternative boot environment (BE) without bringing the system down. Solaris Live Upgrade software also provides a mechanism to fallback to the previous state if a problem occurs with the software upgrade or patch application.

Solaris Live Upgrade software provides a framework to create, manage, manipulate, and activate multiple BEs on the same system. Live Upgrade software provides support for advanced file system operations such as splitting, merging, or sharing file systems between BEs. The software also allows migrating file systems from a storage device to another device. Live Upgrade works with mirrors and other disk layouts. Live Upgrade enables you to maintain multiple installations of software packages including the Solaris OE, which will share user data file systems. These inactive BEs can be used as a backup for fault tolerance.

Solaris Live Upgrade has integrated Solaris Flash technology. This functionality provides a mechanism to install the Flash archive on an inactive BE, while the active BE is fully functional and unaffected by the installation. When satisfied, the administrator can migrate to the other environment by rebooting to that environment.

Solaris Live Upgrade software can be used to manage patch updates. Patches can be applied to an inactive BE and, after they have been tested, be rolled out to the active BE. This process helps in reducing the system down-time. In addition, this process makes it less difficult to fallback to a previous state with a reboot. This process also helps in reducing service outages associated with the normal test and evaluation process required when introducing a new version of a patch to an operating environment.

Live Upgrade supports software migration and falling back to a previous active BE if a failure occurs. Live Upgrade also provides a mechanism to analyze the reasons for failure. Live Upgrade can also synchronize the differences between the active

and inactive BEs. This feature helps to prevent any losses while the system is in the upgrade or fallback process. FIGURE 4 illustrates the differentiation of active and inactive BEs.

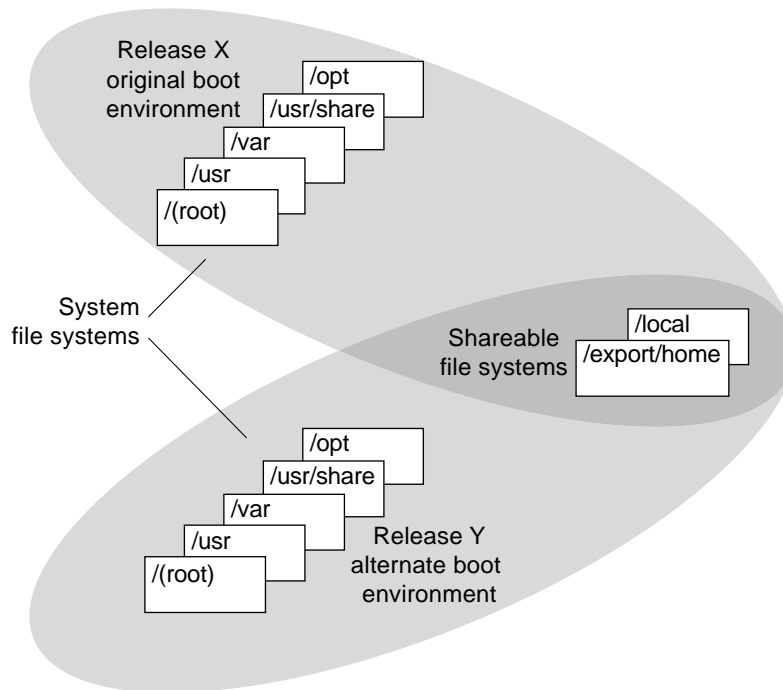


FIGURE 4 Active and Inactive (Alternate) Boot Environments

Boot Environment

A BE is a grouping of file systems and their associated mount points. These BEs can be created on the same disk or on separate disks; however, a single root “/” file system is the recommended layout for Solaris OE (Howard, 2000). The active BE is the BE that is currently booted, and all others are considered inactive or alternate BEs.

Solaris Live Upgrade Software Process

Solaris Live Upgrade software provides functionality for software upgrade, system maintenance, and installation of a Flash archive. Live Upgrade has the following operational steps:

1. Creating a BE
2. Upgrading an inactive BE
3. Installing Solaris Flash archive on inactive BE
4. Activating an inactive BE
5. Falling back to original BE

Creating a Boot Environment

Live Upgrade provides mechanisms to distinguish between critical file systems and shareable file systems. Critical file systems are those required by Solaris OE and have separate entries in the `vfstab` file of the active and inactive BEs. For example, root (`/`), `/usr`, `/var` or `/opt`. These file systems are always copied from the active to the inactive BE. Shareable file systems are user defined (for example, `/export`) that contain the same mount point and device entry in the `vfstab` file in both the active and inactive BEs (FIGURE 5). After defining an alternative BE using the `lucreate(1M)` command, critical file systems will be copied over. Refer to the *Solaris 9 Installation Guide* for detailed instructions.

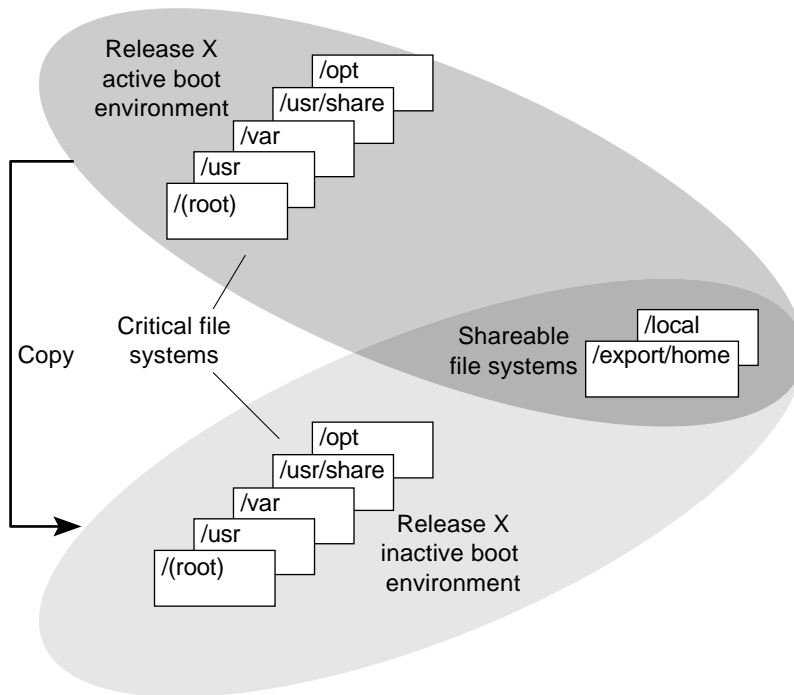


FIGURE 5 Creating an Inactive Boot Environment

Upgrading a Boot Environment

Once an inactive BE has been created, it remains unchanged and is identical to the active BE. Solaris Live Upgrade software provides a mechanism to perform a software upgrade or patch application on the inactive BE with the `luupgrade(1M)` command. The upgrade process will not affect any files in the active BE. FIGURE 6 illustrates an upgrade of an inactive BE.

Live Upgrade runs as a background process on the active BE. To minimize impact on the active BE, Live Upgrade provides a sophisticated resource manager to allow administrators to tailor Live Upgrade's usage of the active BE. This management interface can be used to further enhance the mission-critical response times of the active BE, while the inactive BE is upgraded or flashed. The constraints are controlled with the `/etc/default/lu` configuration file.

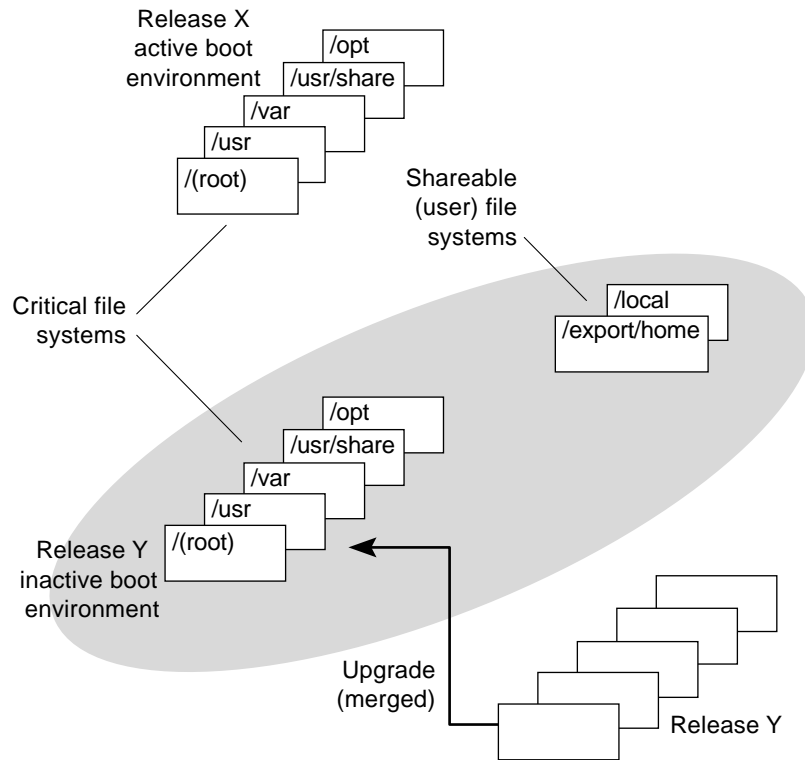


FIGURE 6 Upgrading an Inactive Boot Environment

Installing a Solaris Flash Archive on an Inactive Boot Environment

The latest release of Live Upgrade, bundled with Solaris 9 OE, provides a mechanism to install a Solaris Flash archive. For example, a Flash archive containing the Solaris 9 OE and other software can be installed on an inactive BE while a Solaris 2.6 OE is up and live on the active BE. When you install the Solaris Flash archive on a system, all the files in the archive are copied to that system and a new software version is installed without affecting the active BE. But, unlike an upgrade that merges files, installing a Solaris Flash archive overwrites the files as an initial installation would. FIGURE 7 illustrates an installation of a Solaris Flash archive on an inactive BE. Refer to the *Solaris 9 Advanced Installation Guide* for more details on installing a Flash archive with Live Upgrade.

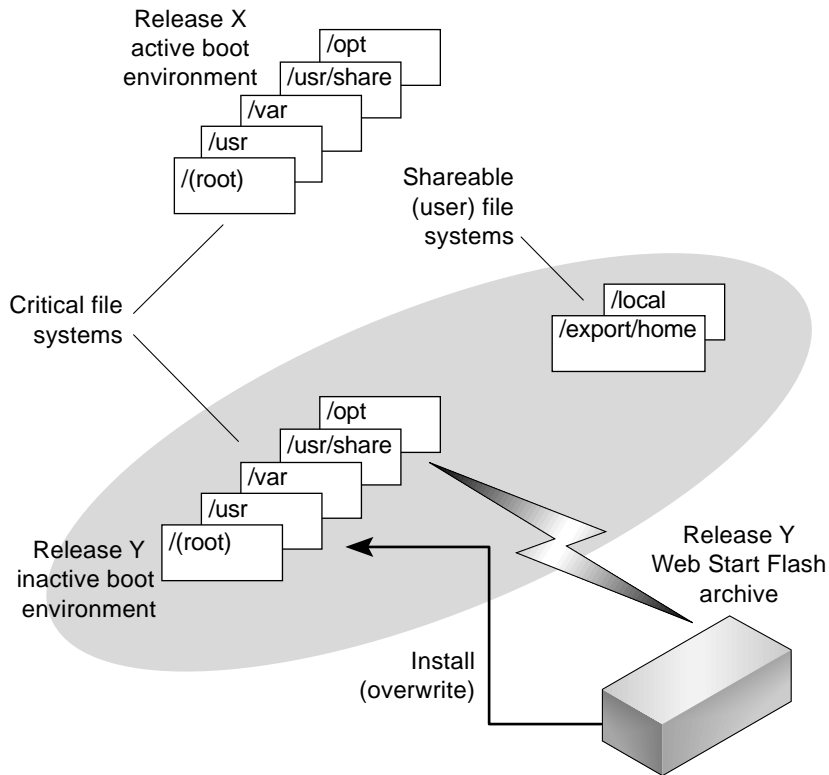


FIGURE 7 Installing a Solaris Flash Archive

Activating an Inactive Boot Environment

Solaris Live Upgrade software provides a mechanism to activate any inactive BE. Once the upgrade process is complete, the inactive BE can be activated by the `luactivate(1M)` command. Now the target BE will be activated and will become the active BE after reboot. Solaris Live Upgrade performs file synchronization when activating the inactive BE. FIGURE 8 illustrates a switch after activation of an inactive BE.

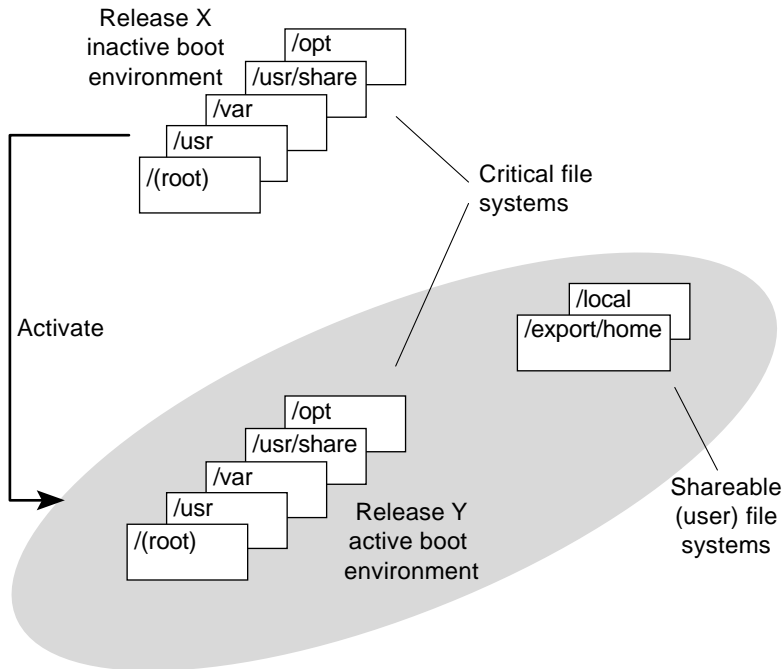


FIGURE 8 Activating an Alternative Boot Environment

FallBack Mechanism

Solaris Live Upgrade software provides a mechanism to fallback if a failure occurs or if you are not satisfied with the upgrade process. The software also provides functionality to immediately fallback to the original BE with an activation using the `luactivate(1M)` command and then rebooting. The upgraded BE will be saved and Live Upgrade provides functionality to analyze the failure. The process to boot from a media device is first to mount the root file system, then run the `luactivate` on the target fallback BE, and finally reboot. FIGURE 9 illustrates the switch that is made when you reboot to fallback. Refer to the *Solaris Advanced Installation Guide* for more details on the Live Upgrade fallback mechanism.

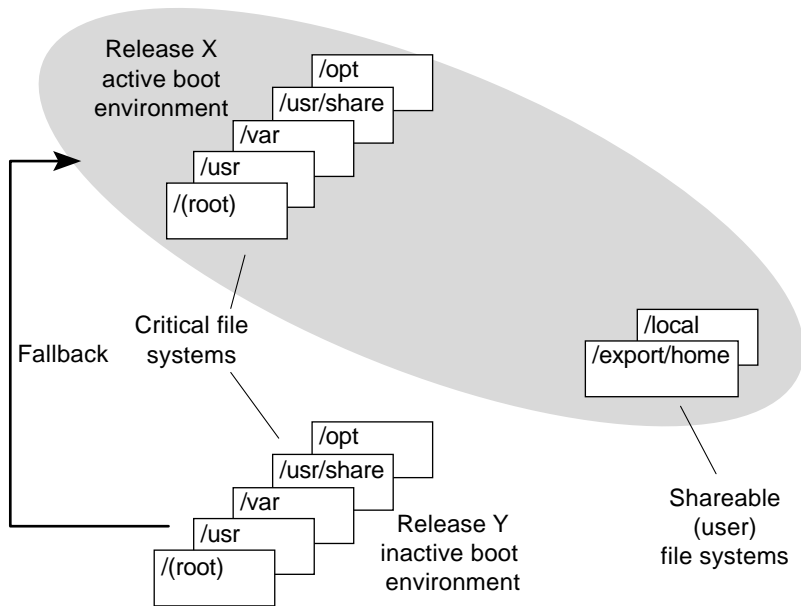


FIGURE 9 FallBack to the Original Boot Environment

About the Authors

Jay Daliparth

Jay Daliparth is a Senior Product Manager and Team Leader in N1 and Management Solutions Marketing. He is responsible for marketing and managing products in the Systems Management area, with a focus on Installation, Patch Management, Provisioning and Change Management. Jay Daliparth has been working at Sun since October 2000.

Prior to Sun, he worked in Oracle Corporation for three years as a Product Manager and as a Principal Consultant with a focus on business applications software. Before joining Oracle, he worked for seven years at the University of Massachusetts, Amherst as an Adjunct Asst. Professor/Sr. Research Fellow. Prior to that, he worked as an Asst. Professor at the A.P. Agricultural University, India. He holds a Doctorate

degree in Agricultural Sciences and has published many articles in various Journals. Dr. Daliparthy has many years of experience in programming, consulting, marketing, teaching, and research.

James Falkner

James Falkner has been at Sun for the past six years developing new software deployment models and provisioning technologies. As a Staff Engineer in the Solaris Provisioning group, he has worked on a number of products including Web Start Wizards, JumpStart, Flash, Live Upgrade, and Secure Packaging and Installation. He holds U.S. and European patents for techniques for Internet-based provisioning, and has published several articles and white papers for BigAdmin (Sun's system administrator portal), SuperG, and SysAdmin magazine. He can be reached at james.falkner@sun.com

References

Howard, John S. and Noordergraaf, Alex "Web Start Flash," Sun BluePrints OnLine, November 2001,

<http://www.sun.com/solutions/blueprints/browsedate.html#1101>

Solaris Live Upgrade 2.0 Guide, Sun Microsystems.

To access this book online, go to <http://docs.sun.com>

Related Resources

Howard, John S. "An Introduction to Live Upgrade," Sun BluePrints OnLine, July 2000,

<http://www.sun.com/solutions/blueprints/browsedate.html#0700>

Howard, John S. "Using Live Upgrade 2.0 With JumpStart™ Technology and Web Start Flash," *Sun BluePrints OnLine*, April 2002.

To access this article online, go to:

<http://www.sun.com/solutions/blueprints/browsedate.html#0402>

Howard, John S. and Noordergraaf, Alex *JumpStart Technology: Effective use in the Solaris Operating Environment*. ISBN 0-13-062154-4. Upper Saddle River: Prentice Hall, 2001.

To access this book online, go to:

<http://www.sun.com/solutions/blueprints/pubs.html>

Solaris 8 Installation Guide, Sun Microsystems.

To access this book online, go to <http://docs.sun.com>

Solaris 8 Advanced Installation Guide, Sun Microsystems.

To access this book online, go to <http://docs.sun.com>

Solaris 9 Installation Guide, Sun Microsystems.

To access this book online, go to <http://docs.sun.com>

Solaris 9 Advanced Installation Guide, Sun Microsystems.

To access this book online, go to <http://docs.sun.com>

Ordering Sun Documents

The SunDocsSM program provides more than 250 manuals from Sun Microsystems, Inc. If you live in the United States, Canada, Europe, or Japan, you can purchase documentation sets or individual manuals through this program.

Accessing Sun Documentation Online

The docs.sun.com web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. The URL is <http://docs.sun.com/>

To reference Sun BluePrints OnLine articles, visit the Sun BluePrints OnLine Web site at: <http://www.sun.com/blueprints/online.html>