



An Introduction to Live Upgrade

By John S. Howard - Enterprise Engineering

Sun BluePrints™ OnLine - July 2000



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

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303 USA
650 960-1300 fax 650 969-9131

Part No.: 806-6195-10
Revision 01, July 2000

Copyright 2000 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California 94303 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.

Sun, Sun Microsystems, the Sun logo, JumpStart, Solstice DiskSuite, SunOS, Ultra Enterprise, Sun BluePrints and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

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.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

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 2000 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303 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.

Sun, Sun Microsystems, le logo Sun, JumpStart, Solstice DiskSuite, SunOS, Ultra Enterprise, Sun BluePrints, et Solaris sont des marques de fabrique ou des marques déposées 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.

UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

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

An Introduction to Live Upgrade

The process of performing an upgrade of the Solaris™ Operating Environment and upgrading the associated system software unbundled products is one of the most time-consuming and error prone tasks facing system administrators. Further, most mission critical or datacenter systems can not afford to be taken down for a day or even an hour to perform software, patch and Operating Environment upgrades and then test those changes.

Live Upgrade (LU) provides a mechanism to manage and upgrade multiple on-disk Solaris Operating Environments allowing the upgrade of one environment without taking the system down. LU provides a framework to upgrade and work within these multiple on-disk environments, rebooting into the new Operating Environment after the changes to the on-disk software images have been completed.

LU can also provide a safe “fall-back” environment to quickly recover from upgrade problems or failures. Additionally, LU can be used for patch testing and rollout as well as “sidegrades” - the large scale re-organization of on-disk Operating Environments.

This article will provide an introduction and overview of Live Upgrade as well as providing techniques and best practices for the usage of Live Upgrade.

The LU Framework and Mechanism

An Early Access version of LU was provided with the Solaris 8 Operating Environment, however LU works with (and may be installed on) all releases of the Solaris 2.6, 7 and 8 Operating Environments.

Boot Environments

Central to the operation and implementation of LU is the concept of a Boot Environment. A Boot Environment (BE) is a grouping of filesystems and their associated mount points. LU uses the term boot environment instead of boot disk because a BE may be contained on one disk, or may be spread over several disks. LU provides a command line interface and a GUI to create, populate, manipulate and activate boot environments.

BEs may be created on separate disks or on the same disk, however a single root “/” filesystem is the recommended layout for the Solaris Operating Environment. For further information on the preferred method of laying out the root filesystem, and the reasoning behind that method, consult the numerous Sun BluePrints™ OnLine articles on boot disk layout.

The active BE is the BE that is currently booted and active, all other defined BEs are considered inactive. Inactive BEs are also referred to as Alternate Boot Environments (ABEs).

The Early Access version of LU does not allow the user to choose which filesystems are included in a BE. However, BEs may be completely self-contained or they may share filesystems. Only filesystems that do not contain any Operating Environment specific data and must be available in any Operating Environment, should be shared among BEs. For example, user’s home directories on `/export/home` would be a good candidate to be shared among several BEs.

Further, LU provides a mechanism to synchronize individual files among several BEs. This feature is especially useful for maintaining files such as `/etc/passwd` in one BE and then propagating changes out to all BEs.

LU and Volume Managers

LU is “volume manager aware” and is capable of working with disks managed by Solstice DiskSuite™ (SDS) or Veritas Volume Manager (VxVM). However, due to constraints imposed by VxVM and SDS, LU can not directly assign a VxVM volume or SDS metadvice for the root filesystem of a new BE. After a BE is activated and booted, the boot disk may be mirrored using the appropriate VxVM or SDS procedure for mirroring the root disk.

VxVM requires additional reboots during the boot disk encapsulation process and similarly, SDS requires additional reboots when performing the root mirroring process. The use of LU with VxVM or SDS does not add any additional downtime or reboots to either of these processes.

Implementing and Using LU

LU is included on the Solaris 8 Operating Environment media and may be installed with either the supplied installer or by `pkgadd`'ing the LU packages.

Consider the common situation of having to upgrade a production server from the Solaris 2.6 Operating Environment to the Solaris 8 Operating Environment. The server can not be taken down for the time required to do the upgrade and the site change control procedures require that a back-out plan be provided to restore the initial Solaris 2.6 Operating Environment in the case of any unforeseen upgrade failures or software incompatibilities. Using LU, this upgrade may be completed while the Solaris 2.6 Operating Environment is up and “live”.

An overview of the upgrade process using LU is as follows:

1. Create a new BE
2. Populate the new BE by cloning the current Operating Environment
3. Upgrade the new BE
4. Install (or upgrade) unbundled software into the new BE
5. Patch the new BE as necessary
6. When ready to cut over to the new version of the Operating Environment, activate the new BE and reboot into the new BE

Following is an example of using LU to upgrade a system from the Solaris 2.6 Operating Environment to the Solaris 8 Operating Environment. In this example, `racerox` is the system to be upgraded to the Solaris 8 Operating Environment and `blackmesa` is the system that is serving the Solaris 8 Operating Environment product media. `racerox` is currently running the Solaris 2.6 Operating Environment, booted off of `/dev/dsk/c0t0d0s0`, the Solaris 8 Operating Environment will be installed on `/dev/dsk/c0t1d0s0`.

The `c0t1d0` disk has been partitioned prior to executing the following LU commands. However, if changes to the partitioning are needed, they can be implemented via the “Slice” sub-menu in the `lucreate` command GUI.

Note that `racerox` is not taken down or made unavailable to users or application at anytime during this procedure. The only downtime required will be the time required to shutdown and reboot `racerox` when actually cutting over to the new Solaris 8 Operating Environment.

- Create a BE named “Solaris2.6” for the current Solaris 2.6 Operating Environment and a BE named “Solaris8” to upgrade to the Solaris 8 Operating Environment. Note that the Solaris8 BE is initially populated with a copy or “clone” of the Solaris2.6 BE. The copying of BEs can also be scheduled for a later time, such as when the system is in a non-peak usage period.

```

racerox# lucreate -c "Solaris2.6" -n "Solaris8"
Please wait while the configuration files are updated.
Please wait. Configuration validation in progress...
*****
Beginning process of creating Boot Environment: Solaris8
No more user interaction is required until this process is
complete.
*****
Creating file systems
/usr/lib/fs/ufs/mkfs /dev/rdisk/c0t1d0s0 3105360
Warning: inode blocks/cyl group (56) >= data blocks (37) in last
cylinder group. This implies 592 sector(s) cannot be allocated.
/dev/rdisk/c0t1d0s0:      3104768 sectors in 6064 cylinders of 16
tracks, 32 sectors
      1516.0MB in 379 cyl groups (16 c/g, 4.00MB/g, 1920 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
  32, 8256, 16480, 24704, 32928, 41152, 49376, 57600, 65824,
[ backup super-block inodes removed for brevity ]
Populating file systems
Copying file system contents to "Solaris8"
Copying of filesystem/directory is in progress for / ...
Copy of filesystem/directory is complete for /
Creating compare database for /

Updating compare database on other BEs...
      Solaris8..Done.
Making the Boot Environment bootable:  "Solaris8"
Updating ABE's /etc/vfstab....
Writing the new vtoc to disk...
Writing a loader matching the OS release to disk ....
Boot Environment creation complete.

```

The location of / for the Solaris8 BE was selected from the GUI menu. It is also important to note that the location of the primary swap device (in this case /dev/rdisk/c0t0d0s1) will not change.

- Upgrade the Solaris8 BE from the Solaris 2.6 Operating Environment to the Solaris 8 Operating Environment.

```
racerox# ls -la /net/blackmesa.west/JumpStart/Solaris_8
total 28
drwxr-xr-x  3 root    other      512 Apr 28 11:26 .
drwxrwxrwx  7 root    root       512 May  1 14:01 ..
-r--r--r--  1 root    root        84 Jan 21 08:46 .cdtoc
lrwxrwxrwx  1 root    root       30 Apr 28 11:26 .install_config
-> Solaris_8/Misc/.install_config
-rw-r--r--  1 root    root       51 Jan 21 08:46 .slicemapfile
-rw-r--r--  1 root    root       37 Jan 21 08:46 .volume.inf
-r--r--r--  1 root    root      6161 Jan 21 08:46 Copyright
drwxr-xr-x  8 root    root       512 May  2 17:10 Solaris_8
racerox# luupgrade Solaris8 \
/net/blackmesa.west/JumpStart/Solaris_8 yes yes
Processing default locales
  - Specifying default locale (en_US)
Processing profile
Loading local environment and services
Generating upgrade actions
  - Selecting locale (en_US)
Installing 64 bit Solaris packages
Space check complete.
Building upgrade script
Preparing system for Solaris upgrade
Upgrading Solaris: 100% completed

Installation log location
  - /a/var/sadm/system/logs/upgrade_log (before reboot)
  - /var/sadm/system/logs/upgrade_log (after reboot)
Please examine the file:
  - /a/var/sadm/system/data/upgrade_cleanup
It contains a list of actions that may need to be performed to
complete the upgrade. After this system is rebooted, this file can
be found at:
  - /var/sadm/system/data/upgrade_cleanup

Upgrade complete
Adding patches to /a
```

- Add any unbundled software or patches to the Solaris8 BE. In the following example, we add two patches to the Solaris8 BE.

```
racerox# lumount "Solaris8" /a
racerox# cd /net/blackmesa/JumpStart/Solaris_8/Solaris_8/Patches/
8_Recommended
racerox# patchadd -R /a \
-M /net/blackmesa/JumpStart/Solaris_8/Solaris_8/Patches/
8_Recommended \
patch_order
Checking installed patches...
Verifying sufficient filesystem capacity (dry run method)...
Installing patch packages...
Patch number 109137-01 has been successfully installed.
See /a/var/sadm/patch/109137-01/log for details
Patch packages installed:
    SUNWcsu

Checking installed patches...
Verifying sufficient filesystem capacity (dry run method)...
Installing patch packages...
Patch number 109320-01 has been successfully installed.
See /a/var/sadm/patch/109320-01/log for details
Patch packages installed:
    SUNWpcu
    SUNWpsu
racerox# luumount "Solaris8"
```

Note that we use of the `lumount` and `luumount` commands to mount and unmount all the filesystems from the Solaris8 BE, consult the `lumount` and `luumount` man pages for specific details on their usage.

The `luupgrade` command may also be used to upgrade unbundled software or to spawn a limited shell to manually install software within an inactive alternate BE. Consult the `luupgrade` man page for details on its usage.

- Cut over to the new Solaris 8 Operating Environment by activating the Solaris8 BE and rebooting when convenient.

```

racerx# luactivate "Solaris8"
*****
The target BE has been activated. It will be used when you reboot.
NOTE: You must use either init or shutdown when you reboot. If
you do not use one of these commands, the system will not boot
using the target BE.
*****
In case of a failure while booting to the target BE, the following
process needs to be followed to fallback to the currently working
boot environment.

1. Enter the PROM monitor (ok prompt).
2. Change the boot device back to the original BE by typing:

    setenv boot-device disk:a

3. Boot to the original BE by typing:

    boot

*****
racerx# init 6
racerx#
INIT: New run level: 6
The system is coming down. Please wait.
System services are now being stopped.
Print services stopped.
Stopping the syslog service.
Jun 26 12:20:51 racerx syslogd: going down on signal 15
Activating Boot Environment:Solaris8
Writing the new vtoc to disk...
fmthard: New volume table of contents now in place.
Writing a loader matching the OS release to disk ....
The boot device will be: /dev/dsk/c0t1d0s0
Changing the Primary Boot Device
The boot device will be disk1:a disk:a
Boot Environment activation complete.
Jun 26 12:21:26 snmpdx: received signal 15
The system is down.
syncing file systems... done
rebooting...
Resetting ...

```

```

Sun Ultra 1 UPA/SBus (UltraSPARC 167MHz), No Keyboard
OpenBoot 3.25, 128 MB memory installed, Serial #8191952.
Ethernet address 8:0:20:7c:ff:dd, Host ID: 807cfffdd.

Rebooting with command: boot
Boot device: disk1:a File and args:
SunOS Release 5.8 Version Generic 64-bit
Copyright 1983-2000 Sun Microsystems, Inc. All rights reserved.
configuring IPv4 interfaces: hme0.
Hostname: racerx
The system is coming up. Please wait.
starting rpc services: rpcbind done.
Setting netmask of hme0 to 255.255.255.0
Setting default IPv4 interface for multicast: add net 224.0/4:
gateway racerx
syslog service starting.
Print services started.
volume management starting.
The system is ready.

racerx console login: root
Password:
Last login: Mon Jun 26 09:23:36 on console
Jun 26 12:29:13 racerx login: ROOT LOGIN /dev/console
Sun Microsystems Inc. SunOS 5.8 Generic February 2000
racerx# uname -a
SunOS racerx 5.8 Generic sun4u sparc SUNW,Ultra-1
racerx# swap -l
swapfile dev swaplo blocks free
/dev/dsk/c0t0d0s1 32,1 16 1048784 1048784

```

It is important to note the information provided at the completion of the `luactivate` command. This provides the procedure required to fallback to the original Solaris 2.6 Operating Environment image.

Additionally, note that after the Solaris8 BE was activated `init` was used to shut the system down. For proper activation of the BE, it is required that only the `init` or `shutdown` be used when switching between BEs.

As previously mentioned, even though we have activated the Solaris8 BE, the primary swap device (in this case `/dev/rdisk/c0t0d0s1`) has not changed.

Beyond Upgrades

Sidegrades

The ability to create multiple boot environments and populate those boot environments with live Operating Environment data enables the system administrator greater flexibility to react to changing user needs with minimal downtime. LU enables the system administrator to perform “sidegrades”, the large scale reorganization of the Operating Environment with minimal impact to the user. The following section details using LU to perform such a sidegrade.

Migration to a site-standard boot environment

Over the course of time, the on-disk data of systems and Operating Environments tend towards a state of greater disorder. Work-arounds and special cases are implemented and then are never re-architected to the site standard. These work-arounds and special cases are usually left in place because the downtime to resolve them is not available. LU can be used to reinforce a site standard for boot environments onto systems that have suffered at the hands of entropy and work-arounds.

For example, consider a system that was originally installed with an undersized `/` filesystem. `/` was sized such that it was large enough for the initial installation of the Operating Environment, however over the course of time several patches were installed. The disk space requirements of these patches (and the space needed to save the previous versions of the files incase of backout) caused `/` to become 100% full. As a work-around to alleviate the space constraints on `/` the system administrator moved `/var/sadm` to another filesystem (`/opt2/var/sadm`) and then created a symbolic link from `/var/sadm` to `/opt2/var/sadm`.

Using `lucreate`, we can create a copy of the current BE into an alternate boot environment with a larger `/` (and large enough for future patch needs). Then using `luactivate` we select the new BE and reboot when convenient.

LU for patch testing and implementation

As seen in the previous example, LU provides a mechanism with which the system administrator can readily manipulate an inactive Operating Environment. This mechanism can be exploited to provide a software and patch testbed (along with a fallback procedure) for systems where a completely duplicated test environment is

unavailable or uneconomical. Consider an Ultra Enterprise™ 450 workgroup server running the Solaris 8 Operating Environment, this system may provide a necessary, but non-mission critical service. Patches and new software releases should be tested before installation on the Ultra Enterprise 450 however completely duplicating the Ultra Enterprise 450 for a test environment is not possible in this instance.

By utilizing LU, an alternate BE can be created with the Solaris 8 Operating Environment installed in it. Patches and software upgrades can then be applied to the alternate BE and the alternate BE then activated. If problems with the newly applied patches or software are encountered, the system administrator can quickly and easily fallback to the stable, known-good original BE.

Conclusion

This article has provided an introduction and overview of Live Upgrade as well as providing techniques and best practices for the usage of Live Upgrade. LU is a valuable tool providing a mechanism for upgrading the Solaris Operating Environment with minimal downtime. Further, LU enables the system administrator to make large scale changes to the Solaris Operating Environment with minimal impact to the user or applications. Most importantly, LU provides a safe and consistent fall-back environment in case an upgrade or software installation fails.

Author's Bio: John S. Howard

John S. Howard is currently a Staff Engineer in the Enterprise Engineering group at Sun Microsystems in San Diego, California. He has worked as a software engineer and systems administrator for the past 19 years. Prior to Enterprise Engineering, John worked in Enterprise Services as an Area System Support Engineer for five years. As an ASSE, he was responsible for developing and performing Reliability, Accessibility, and Serviceability (RAS) studies of customer datacenters and the development of proactive Enterprise RAS Services. Prior to Sun, John held engineering positions at: The Chicago Board of Trade Clearing Corporation, Datalogics Inc, and Rand McNally. Throughout his career he has developed: pagination and publishing software, loose-leaf publishing systems, extensive SGML systems development, database publishing systems, text editors and WYSIWIG systems, and device drivers.