Managing Solaris™ Operating Environment Upgrades with Live Upgrade 2.0

By John S. Howard - Enterprise Engineering

Sun BluePrints™ OnLine - February 2002

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.: 816-4351-10
Revision 1.0
Edition: February 2002

Managing Solaris™ Operating Environment Upgrades with Live Upgrade 2.0

Performing an upgrade of an operating system and the associated system software and unbundled products is one of the most time-consuming and error prone tasks facing system administrators. Further, most mission-critical or datacenter systems cannot afford to be taken down for any length of time to perform and test patches and to do software upgrades.

Live Upgrade (LU) provides a mechanism to upgrade and manage multiple on-disk copies of the Solaris™ Operating Environments (Solaris OEs). Using LU, you can upgrade an environment without taking the system down. LU provides a framework to upgrade and work within these multiple on-disk environments and reboots into the new Solaris OE after you complete the changes to the on-disk software images.

LU can also provide a safe “fall-back” environment to quickly recover from upgrade problems or failures. Additionally, you can use LU for patch testing and rollout, as well as sidegrades—the large scale reorganization of on-disk OEs.

This article will provide the following:

- An introduction to LU concepts and terminology
- An example Solaris OE upgrade
- Uses of LU beyond Solaris OE upgrades

This article is the first of a three part series; Part 2 will explain using LU 2.0 with a logical volume manager, such as VERITAS Volume Manager (VxVM) or Solstice DiskSuite™ software. Part 3 of this series will examine using LU with JumpStart™ technology and Web Start Flash.

Live Upgrade Framework and Terminology

LU 2.0 was introduced with the Solaris 8 10/01 OE (Update 6). On the Solaris 8 10/01 OE media, the LU 2.0 packages are located in the EA (Easy Access) directory of the CD marked “Solaris 8 Software 2 of 2.” With Solaris 8 01/02 OE (Update 7), LU 2.0 was moved from the EA area to the product area and is bundled with the OE packages.

LU 2.0 was also released as a Web Release (08/01) which is available at http://www.sun.com/solaris/liveupgrade. LU 2.0 works with, and can be installed on, all releases of the Solaris OE versions 2.6, 7, and 8. LU 2.0 is the first general-availability release of LU. The use of LU 2.0 is recommended over the use of LU 1.0, and LU 1.0 must never be used in a production environment or on a production server.

Upgrade Philosophy

To upgrade to a particular release of the Solaris OE, you must install the version of LU that is bundled with the release of the Solaris OE to which you want to upgrade. Then, use that version of LU to upgrade to the desired release of the Solaris OE.

For example, if you are running the Solaris 2.6 OE and you want to upgrade to the Solaris 8 10/01 OE, you would install LU from the Solaris 8 10/01 OE distribution onto the Solaris 2.6 OE system, and then use the Solaris 8 10/01 OE version of LU to upgrade the system to the Solaris 8 10/01 OE.

For the SPARC™ platform edition of the Solaris OE, you can use LU 2.0 to upgrade from Solaris 2.6 OE and later versions. For the Intel platform edition of the Solaris OE, you can use LU 2.0 to upgrade from Solaris 2.7 OE Intel platform edition and later versions. For both the SPARC and Intel platform editions, the minimum supported version of the Solaris OE to which you can upgrade is Solaris 8 01/01 OE (Update 3).

To upgrade to the Solaris 8 01/01 OE (Update 3), the Solaris 8 04/01 OE (Update 4), or the Solaris 8 07/01 OE (Update 5), install and use the LU 2.0 08/01 Web Release at http://www.sun.com/solaris/liveupgrade.

To upgrade to the Solaris 8 10/01 OE (Update 6), install and use the LU 2.0 10/01 OE located in the EA area of the Solaris 8 10/01 OE distribution.

To upgrade to the Solaris 8 01/02 OE (Update 7), install and use the LU 2.0 01/02 software that is integrated into the operating system package area.

Boot Environments

The concept of a boot environment (BE) is central to the operation and implementation of LU. A BE is a group of file systems and their associated mount points. LU uses the term “boot environment” instead of “boot disk” because a BE can be contained on one disk or can be spread over several disks. LU provides a command-line interface and a character-based user interface (CUI) to create, populate, manipulate, and activate BEs.

Note – The CUI has a few restrictions. The CUI is neither localized nor internationalized. Also, the existing CUI does not provide access to the full functionality of LU.

You can create BEs on separate disks or you can create them on the same disk; however, a single root (/) file system is the recommended layout for the Solaris OE.

The active BE is the one that is currently booted and active; all other defined BEs are considered inactive. Inactive BEs are also referred to as alternate boot environments (ABEs).

BEs can be completely self-contained, or they can share file systems. Only file systems that do not contain any OE-specific data and that must be available in any OE should be shared among BEs. For example, users’ home directories on the /export/home file system would be a good candidate to share among several BEs.

If you used multiple file systems for the Solaris OE, such as separate file systems for /kernel, /usr, /etc, /, etc., do not share these OE-specific file systems among BEs. In addition, do not split certain file systems from / (such as /kernel, /etc, /dev, or /devices). If you split them off onto a separate file system from /, the BE that is created may not be bootable.

Additionally, 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 to all BEs.

BEs created with LU may be backed up with the ufsdump or fssnap commands. Consult the man pages for information about the uses of these commands.

Implementing and Using Live Upgrade

To appreciate the value of using LU to upgrade a system, consider the common situation of having to upgrade a production server from the Solaris 7 OE to the Solaris 8 OE. Most likely, you could not take the server down to do the upgrade.

Additionally, the site change control procedures likely require that you provide a back-out plan to restore the initial Solaris 7 OE in the case of any unforeseen upgrade failures or software incompatibilities. Using LU, you can complete this upgrade while the Solaris 7 OE is up and “live” while maintaining the Solaris 7 OE as a fallback in case failure during the upgrade procedure.

In the following example, the latest Solaris 7 OE recommended patch cluster and the LU2.0 packages are installed on napa; then, the system is upgraded. The following tasks outline the process required to upgrade a system using LU:

1. Create and populate a new BE by cloning the current OE.
2. Upgrade the new BE.
3. Install (or upgrade) unbundled software, patching as necessary, in the new BE.
4. When you are ready to cut over to the new version of the OE, activate the new BE and reboot into the new BE.

Rather than using slice 7 of the boot disk for the /export file system, use this slice for the clone OE or as an ABE.

The following example uses LU to upgrade a system from the Solaris 7 OE to the Solaris 8 10/01 OE. In this example, napa is being upgraded to the Solaris 8 OE; napa currently runs the Solaris 7 OE, booted off of /dev/dsk/c1t8d0s0; the Solaris 8 10/01 OE will be installed on /dev/dsk/c0t0d0s0.

When using the LU command line interface, you must use the Solaris OE format command to partition the disk for the new ABE before the lucreate command executes. In the following example, the c0t0d0 disk was partitioned prior to executing the following LU commands. However, if changes to the partitioning are needed, you can implement them through the Slice submenu in the lucreate command CUI.

For simplicity, this example upgrades napa using a locally mounted Solaris 8 10/01 CD-ROM. The use of LU, in conjunction with a JumpStart installation or Web Start Flash, will be detailed in Part 3 of this LU Sun BluePrints series. Additionally, consult the *Solaris Live Upgrade 2.0 Guide* (available at http://docs.sun.com) or the luupgrade man page for information about using Web Start Flash with LU.

Note that napa is not taken down or made unavailable to users or applications at any time during this procedure. The only downtime is the time required to shutdown and reboot napa when cutting over to the Solaris 8 OE.

Step 1: Creating and Populating a New Boot Environment

The following example creates a new BE named “Solaris7-0399” for the current Solaris 7 OE, and an ABE named “Solaris8-1001” for upgrading to the Solaris 8 10/01 OE. Note that the BE for the Solaris 8 OE is initially populated with a copy or “clone” of the BE for the Solaris 7 OE. You can schedule the copying of BEs during a time when the system is in a non-peak usage period.

```
napa# cat /etc/release
            Solaris 7 3/99 s998s_u1SunServer_10 SPARC
  Copyright 1999 Sun Microsystems, Inc.  All Rights Reserved.
                 Assembled 26 January 1999
napa# pkginfo |grep SUNWlu
application SUNWlur        Live Upgrade 2.0 10/01 (root)
application SUNWluu        Live Upgrade 2.0 10/01 (usr)
napa# lucreate -c "Solaris7-0399" \
> -m /:/dev/dsk/c0t0d0s0:ufs \
> -n "Solaris8-1001"
Please wait while your system configuration is determined.
Determining what file systems should be in the new BE.
Searching /dev for possible BE filesystem devices
Please wait while the configuration files are updated.
Please wait. Configuration validation in progress...
************************************************************
Beginning process of creating Boot Environment <Solaris8-1001>.
No more user interaction is required until this process is
complete.
************************************************************
Setting BE <Solaris8-1001> state to Not Complete.
Creating file systems on BE <Solaris8-1001>.
( continued on following page )
```

```
Creating <ufs> file system on </dev/dsk/c0t0d0s0>.
/dev/rdsk/c0t0d0s0:     12584484 sectors in 4356 cylinders of 27
tracks, 107 sectors
        6144.8MB in 137 cyl groups (32 c/g, 45.14MB/g, 5632 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
32, 92592, 185152, 277712, 370272, 462832, 555392, 647952, 740512,
833072,
 925632, 1018192, 1110752, 1203312, 1295872, 1388432, 1480992,
1573552,1666112, 1758672, 1851232, 1943792, 2036352, 2128912,
2221472, 2314032, 2406592, 2499152, 2591712, 2684272, 2776832,
2869392, 2958368, 3050928, 3143488, 3236048, 3328608, 3421168,
3513728, 3606288, 3698848, 3791408, 3883968, 3976528, 4069088,
4161648, 4254208, 4346768, 4439328, 4531888, 4624448, 4717008,
4809568, 4902128, 4994688, 5087248, 5179808, 5272368, 5364928,
5457488, 5550048, 5642608, 5735168, 5827728, 5916704, 6009264,
6101824, 6194384, 6286944, 6379504, 6472064, 6564624, 6657184,
6749744, 6842304, 6934864, 7027424, 7119984, 7212544, 7305104,
7397664, 7490224, 7582784, 7675344, 7767904, 7860464, 7953024,
8045584, 8138144, 8230704, 8323264, 8415824, 8508384, 8600944,
8693504, 8786064, 8875040, 8967600, 9060160, 9152720, 9245280,
9337840, 9430400, 9522960, 9615520, 9708080, 9800640, 9893200,
9985760, 10078320, 10170880, 10263440, 10356000, 10448560,
10541120, 10633680, 10726240, 10818800, 10911360, 11003920,
11096480, 11189040, 11281600, 11374160, 11466720, 11559280,
11651840, 11744400, 11833376, 11925936, 12018496, 12111056,
12203616, 12296176, 12388736, 12481296, 12573856,
Mounting file systems for BE <Solaris8-1001>.
Calculating required sizes of file systems for BE <Solaris8-1001>.
( continued on following page )
```

```
Populating file systems on BE <Solaris8-1001>.
Copying file system contents to BE <Solaris8-1001>.
INFORMATION: Setting asynchronous flag on ABE <Solaris8-1001>
mount point </.alt.3606/> file system type <ufs>.
Copying of file system / directory </> is in progress...
Copying of file system / directory </> completed successfully.
Creating compare database for file system </>.
Updating compare database on other BEs.
Updating compare database on BE <Solaris8-1001>.
Compare databases updated on all BEs.
Making Boot Environment <Solaris8-1001> bootable.
Making the ABE bootable.
Updating ABE's /etc/vfstab file.
The update of the vfstab file on the ABE succeeded.
Updating ABE's /etc/mnttab file.
The update of the mnttab file on the ABE succeeded.
Updating ABE's /etc/dumpadm.conf file.
The update of the dumpadm.conf file on the ABE succeeded.
Updating partition ID tag on boot environment <Solaris8-1001>
device </dev/rdsk/c0t0d0s2> to be root slice.
Updating boot loader for <SUNW,Ultra-60> on boot environment
<Solaris8-1001> device </dev/dsk/c0t0d0s0> to match OS release.
Making the ABE <Solaris8-1001> bootable succeeded.
Setting BE <Solaris8-1001> state to Complete.
Creation of Boot Environment <Solaris8-1001> successful.
Creation of Boot Environment <Solaris8-1001> successful.
```

The location of / for the Solaris8-1001 BE was specified by the -m option on the lucreate command. It is also important to note that the location of the primary swap device (in this case, /dev/dsk/c1t8d0s1) was not changed.

Step 2: Upgrading the Alternate Boot Environment

After creating and populating the ABE, it is upgraded as shown in the following example.

```
napa# luupgrade -u -n "Solaris8-1001" \
> -s /cdrom/sol_8_701_sparc/s0
Validating the contents of the media </cdrom/sol_8_701_sparc/s0>.
The media is a standard Solaris media.
The media contains an operating system upgrade image.
The media contains <Solaris> version <8>.
The media contains patches for the product.
Locating upgrade profile template to use.
Locating the operating system upgrade program.
Checking for existence of previously scheduled Live Upgrade
requests.
Creating upgrade profile for BE <Solaris8-1001>.
Updating ABE's /etc/vfstab file.
The update of the vfstab file on the ABE succeeded.
Determining packages to install or upgrade for BE <Solaris8-1001>.
Performing the operating system upgrade of the BE <Solaris8-1001>.
CAUTION: Interrupting this process may leave the boot environment
unstable or unbootable.
The operating system upgrade completed.
Adding operating system patches to the BE <Solaris8-1001>.
The operating system patch installation completed.
INFORMATION: </var/sadm/system/logs/upgrade_log> contains a log of
the upgrade operation.
INFORMATION: </var/sadm/system/data/upgrade_cleanup> contains a
log of cleanup operations required.
INFORMATION: Please review the above listed files on BE <Solaris8-
1001> to determine if any additional cleanup work is required, or
installers on additional volumes of the media being upgraded to
need to be run, before activating the BE.
The Live Upgrade of the BE <Solaris8-1001> is completed.
```

Step 3: Adding Software and Patches to the Alternate Boot Environment

After upgrading the ABE, any necessary unbundled software or patches can be installed to the Solaris8-1001 BE. The -p and -P options of the luupgrade command can be used to add or remove, respectively, software packages to an ABE. The -t and -T options of the luupgrade command can be used to add or remove, respectively, patches to an ABE.

Consult the *Solaris Live Upgrade 2.0 Guide* (available at http://docs.sun.com) or the luupgrade man page for details about the usage of these options.

Step 4: Activating the New Solaris Operating Environment

After all of the other steps involved in upgrading the OE in the Solaris8-1001 ABE have been performed, the ABE is activated by rebooting at a convenient time as shown in the following example.

```
napa# luactivate "Solaris8-1001"
A Live Upgrade Sync operation will be performed on startup of boot
environment <Solaris8-1001>.

**********************************************************

The target boot environment 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 boot environment by
typing:

    setenv boot-device rootdisk

3. Boot to the original boot environment by typing:

     boot

**********************************************************
Activation of boot environment <Solaris8-1001> successful.
```

```
napa# init 6
napa#
INIT: New run level: 0
The system is coming down.  Please wait.
System services are now being stopped.
Print services stopped.
Oct  2 15:35:02 napa syslogd: going down on signal 15
Live Upgrade: Deactivating current boot environment <Solaris7-
0399>.
Live Upgrade: Executing Stop procedures for boot environment
<Solaris7-0399>.
Live Upgrade: Current boot environment is <Solaris7-0399>.
Live Upgrade: New boot environment will be <Solaris8-1001>.
Live Upgrade: Activating boot environment <Solaris8-1001>.
Live Upgrade: Updating partition ID tag on boot environment
<Solaris8-1001> device </dev/rdsk/c0t0d0s2> to be root slice.
fmthard:  New volume table of contents now in place.
Live Upgrade: Updating boot loader for <SUNW,Ultra-60> on boot
environment
<Solaris8-1001> device </dev/rdsk/c0t0d0s0> to match OS release.
Live Upgrade: The boot device for boot environment <Solaris8-1001>
will be </dev/dsk/c0t0d0s0>.
Live Upgrade: Changing primary boot device to boot environment
<Solaris8-1001>.
Live Upgrade: The current boot environment <Solaris7-0399> boots
from device <rootdisk rootmirror rootmirror2>.
Live Upgrade: The new boot environment <Solaris8-1001> boots from
device <disk:a rootdisk>.
Live Upgrade: Activation of boot environment <Solaris8-1001>
completed.
The system is down.

syncing file systems... done
Resetting ...
```

```
screen not found.
Can't open input device.
Keyboard not present.  Using ttya for input and output.

Sun Ultra 60 UPA/PCI (2 X UltraSPARC-II 450MHz), No Keyboard
OpenBoot 3.27, 2048 MB memory installed, Serial #13100131.
Ethernet address 8:0:20:c8:7:11, Host ID: 80c80711.

Initializing Memory
Boot device: disk:a  File and args:
SunOS Release 5.8 Version Generic_108528-11 64-bit
Copyright 1983-2001 Sun Microsystems, Inc.  All rights reserved.
configuring IPv4 interfaces: hme0.
Hostname: napa
Configuring /dev and /devices
Configuring the /dev directory (compatibility devices)
The system is coming up.  Please wait.
checking ufs filesystems
/dev/rdsk/c1t8d0s7: is clean.
Live Upgrade: Synchronizing new boot environment.
Live Upgrade: Previous boot environment was <Solaris7>.
Live Upgrade: Current boot environment is now <Solaris8-1001>.
Configuring network interface addresses: hme0.
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 napa
syslog service starting.
Print services started.
volume management starting.

The system is ready.

napa console login: root
Password:
Last login: Tue Oct  2 14:01:40 on console
Oct  2 15:38:46 napa login: ROOT LOGIN /dev/console
Sun Microsystems Inc.   SunOS 5.8       Generic February 2000
napa#
napa# cat /etc/release
             Solaris 8 10/01 s28s_u6wos_08a SPARC
  Copyright 2001 Sun Microsystems, Inc.  All Rights Reserved.
                 Assembled 12 September 2001
```

After the Solaris8-1001 BE was activated, init was used to shut the system down. To properly activate the BE, use only init or shutdown when switching between BEs. Note that the information provided at the completion of the luactivate command provides the procedure required to fallback to the original Solaris 7 OE image.

As previously mentioned, even though the Solaris8-1001 BE has been activated, the primary swap device (in this case, /dev/rdsk/c1t8d0s1) has not changed.

The luactivate command automatically performs synchronization between the new BE and the previous BE. This synchronization is usually needed only the first time a BE is activated. If you prefer to synchronize the BEs manually, specify the -s option to the luactivate command. Manual synchronization instructs that all files listed in /etc/lu/synclist are to be copied to the BE specified by the -n option of luactivate.

Caution – Manual synchronization of BEs (the -s option of the luactivate command) should be done with caution and should only be used between BEs that are the same version of Solaris OE.

To understand what can happen if you synchronize BEs on different versions of the Solaris OE, consider a system with two BEs; a S8 BE with the Solaris 8 OE installed and a S2.6 BE with the Solaris 2.6 OE installed. Because it is possible that the files listed for synchronization in /etc/lu/synclist may be specific to the Solaris 8 OE (for example, the files for configuring or implementing features found only in the Solaris 8 OE or later versions such as IPv6), these files may not function or may misconfigure the facilities of the Solaris 2.6 OE BE. As a result, if the S8 BE is active and manual synchonization is used, the S2.6 BE may encounter problems during boot or may be misconfigured upon completion of the boot process.

Beyond Upgrades

The ability to create multiple BEs and populate them with live OE data provides greater flexibility for reacting to changing system or user needs with minimal downtime.

Performing Sidegrades

LU enables you to perform “sidegrades” (the large scale reorganization of the OE) with minimal impact to the user. This section details methods for using LU to perform sidegrades.

Over the course of time, the on-disk data of systems and OEs tend towards a state of greater disorder, as workarounds and special cases are implemented, and then never rearchitected to the site standard. Workarounds and special cases are usually left in place because the downtime to resolve them is not available. Using LU, you can reinforce a site standard for BEs on systems that have suffered at the hands of entropy and workarounds.

For example, consider a system installed with an undersized root (/) file system. If / is sized such that it is large enough for the initial installation of the OE, however, over the course of time, several patches are installed, the disk space requirements of the patches (and the space needed to save previous versions of the files) may cause / to become 100 percent full. To alleviate space constraints on /, it is common practice to move the contents of /var/sadm to a different file system (for example, /opt2/var/sadm), and then either create a symbolic link from /var/sadm to /opt2/var/sadm or mount /opt2/var/sadm on /var/sadm.

LU can be used to consolidate these seperate file systems back onto a single file system. Using the lucreate command, clone the current BE with a / that is large enough for future patch needs. Then, using the luactivate command, select the new BE and reboot when it is convenient.

The following example demonstrates the use of a sidegrade to consolidate a 2GB / (c0t0d0s0) and a 512MB /var (c0t0d0s3) onto a single 6GB / (c0t1d0s0).

```
barossa# df -k
Filesystem            kbytes    used   avail capacity  Mounted on
/dev/dsk/c0t0d0s0    2056211  821519 1173006    42%    /
/proc                      0       0       0     0%    /proc
fd                         0       0       0     0%    /dev/fd
mnttab                     0       0       0     0%    /etc/mnttab
/dev/dsk/c0t0d0s3     494235    7667  437145     2%    /var
swap                 2489328      16 2489312     1%    /var/run
swap                 2489328      16 2489312     1%    /tmp
barossa# lucreate -c "Solaris8-1001" \
> -m /:/dev/dsk/c0t1d0s0:ufs \
> -n "Solaris8-1001_Prime"
Please wait while your system configuration is determined.
No name for Current BE.
Current BE is named <Solaris8-1001>.
Creating initial configuration for primary BE <Solaris8-1001>.
PBE configuration successful: PBE name <Solaris8-1001> PBE Boot
Device </dev/dsk/c0t0d0s0>.
Determining what file systems should be in the new BE.
Searching /dev for possible BE filesystem devices
Please wait while the configuration files are updated.
Please wait. Configuration validation in progress...
**********************************************************************
Beginning process of creating Boot Environment <Solaris8-
1001_Prime>.
No more user interaction is required until this process is
complete.
**********************************************************************
Setting BE <Solaris8-1001_Prime> state to Not Complete.
Creating file systems on BE <Solaris8-1001_Prime>.
Creating <ufs> file system on </dev/dsk/c0t1d0s0>.
/dev/rdsk/c0t1d0s0:     12585752 sectors in 2671 cylinders of 19
tracks, 248 sectors
6145.4MB in 122 cyl groups (22 c/g, 50.62MB/g, 6208 i/g)
super-block backups (for fsck -F ufs -o b=#) at:

( continued on following page )
```

```
32, 103952, 207872, 311792, 415712, 519632, 623552, 727472,
831392, 935312, 1039232, 1143152, 1247072, 1350992, 1454912,
1558832, 1662752, 1766672, 1870592, 1974512, 2078432, 2182352,
2286272, 2390192, 2494112, 2598032, 2701952, 2805872, 2909792,
3013712, 3117632, 3221552, 3317280, 3421200, 3525120, 3629040,
3732960, 3836880, 3940800, 4044720, 4148640, 4252560, 4356480,
4460400, 4564320, 4668240, 4772160, 4876080, 4980000, 5083920,
5187840, 5291760, 5395680, 5499600, 5603520, 5707440, 5811360,
5915280, 6019200, 6123120, 6227040, 6330960, 6434880, 6538800,
6634528, 6738448, 6842368, 6946288, 7050208, 7154128, 7258048,
7361968, 7465888, 7569808, 7673728, 7777648, 7881568, 7985488,
8089408, 8193328, 8297248, 8401168, 8505088, 8609008, 8712928,
8816848, 8920768, 9024688, 9128608, 9232528, 9336448, 9440368,
9544288, 9648208, 9752128, 9856048, 9951776, 10055696, 10159616,
10263536, 10367456, 10471376, 10575296, 10679216, 10783136,
10887056, 10990976, 11094896, 11198816, 11302736, 11406656,
11510576, 11614496, 11718416, 11822336, 11926256, 12030176,
12134096, 12238016, 12341936, 12445856, 12549776,
Mounting file systems for BE <Solaris8-1001_Prime>.
Calculating required sizes of file systems for BE <Solaris8-
1001_Prime>.
Populating file systems on BE <Solaris8-1001_Prime>.
Copying file system contents to BE <Solaris8-1001_Prime>.
INFORMATION: Setting asynchronous flag on ABE <Solaris8-
1001_Prime> mount point </.alt.4241/> file system type <ufs>.
Copying of file system / directory </var> is in progress...
Copying of file system / directory </var> completed successfully.
Copying of file system / directory </> is in progress...
Copying of file system / directory </> completed successfully.
Creating compare database for file system </var>.
Creating compare database for file system </>.
Updating compare database on other BEs.
Updating compare database on BE <Solaris8-1001_Prime>.
Compare databases updated on all BEs.
Making Boot Environment <Solaris8-1001_Prime> bootable.
Making the ABE bootable.
Updating ABE’s /etc/vfstab file.
The update of the vfstab file on the ABE succeeded.
Updating ABE’s /etc/mnttab file.
The update of the mnttab file on the ABE succeeded.
Updating ABE’s /etc/dumpadm.conf file.
The update of the dumpadm.conf file on the ABE succeeded.
Making the ABE <Solaris8-1001_Prime> bootable succeeded.
Setting BE <Solaris8-1001_Prime> state to Complete.
Creation of Boot Environment <Solaris8-1001_Prime> successful.
Creation of Boot Environment <Solaris8-1001_Prime> successful.
```
(continued on following page)

```
barossa# luactivate Solaris8-1001_Prime
******************************************************************
The target boot environment 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. Boot the machine to Single User mode using a different boot
device (like the Solaris Install CD or Network). Examples:

At the PROM monitor (ok prompt):

     For boot to Solaris CD:  boot cdrom -s

     For boot to network:     boot net -s

3. Mount the Current boot environment root slice to some directory
(like /mnt). You can use the following command to mount:

     mount -Fufs /dev/dsk/c0t0d0s0 /mnt

4. Run <luactivate> utility with out any arguments from the current
boot environment root slice, as shown below:

     /mnt/sbin/luactivate

5. luactivate, activates the previous working boot environment and
indicates the result.

6. Exit Single User mode and reboot the machine.
******************************************************************

Activation of boot environment <Solaris8-1001_Prime> successful.
barossa# lustatus
BE_name              Complete  Active  ActiveOnReboot  CopyStatus
----------------------------------------------------------------
Solaris8-1001        yes       yes     no              -
Solaris8-1001_Prime  yes       no      yes             -
barossa# init 6
```

(continued on following page)

```
INIT: New run level: 6
The system is coming down.  Please wait.
System services are now being stopped.
Print services stopped.
Dec 20 16:14:40 barossa syslogd: going down on signal 15
Live Upgrade: Deactivating current boot environment <Solaris8-
1001>.
Live Upgrade: Executing Stop procedures for boot environment
<Solaris8-1001>.
Live Upgrade: Current boot environment is <Solaris8-1001>.
Live Upgrade: New boot environment will be <Solaris8-1001_Prime>.
Live Upgrade: Activating boot environment <Solaris8-1001_Prime>.
Live Upgrade: Updating partition ID tag on boot environment
<Solaris8-1001_Prime> device </dev/rdsk/c0t1d0s2> to be root
slice.
fmthard:  New volume table of contents now in place.
Live Upgrade: Updating boot loader for <SUNW,UltraSPARC-IIi-
cEngine> on boot environment <Solaris8-1001_Prime> device </dev/
rdsk/c0t1d0s0> to match OS release.
Live Upgrade: The boot device for boot environment <Solaris8-
1001_Prime> will be </dev/dsk/c0t1d0s0>.
Live Upgrade: Changing primary boot device to boot environment
<Solaris8-1001_Prime>.
Live Upgrade: The current boot environment <Solaris8-1001> boots
from device <disk net>.
Live Upgrade: The new boot environment <Solaris8-1001_Prime> boots
from device <disk1:a disk:a>.
Live Upgrade: Activation of boot environment <Solaris8-1001_Prime>
completed.
The system is down.

INIT: failed write of utmpx entry:"s6"
INIT: failed write of utmpx entry:"rb"
syncing file systems... done
rebooting...

Resetting ...

Netra t1 (UltraSPARC-IIi 360MHz), No Keyboard
OpenBoot 3.10.24 ME, 512 MB memory installed, Serial #11659611.
Ethernet address 8:0:20:8:16:73, Host ID: 80b21673.
```
(continued on following page)

```
Initializing Memory
Executing last command: boot
Boot device: disk1:a File and args:
SunOS Release 5.8 Version Generic_108528-11 64-bit
Copyright 1983-2001 Sun Microsystems, Inc.  All rights reserved.
configuring IPv4 interfaces: hme0.
Hostname: barossa
Configuring /dev and /devices
Configuring the /dev directory (compatibility devices)
The system is coming up.  Please wait.
Live Upgrade: Synchronizing new boot environment.
Live Upgrade: Previous boot environment was <Solaris8-1001>.
Live Upgrade: Current boot environment is now <Solaris8-
1001_Prime>.
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 barossa
syslog service starting.
Print services started.
volume management starting.

The system is ready.

barossa console login: root
Password:
Last login: Thu Dec 20 15:49:17 on console
Dec 20 16:22:10 barossa login: ROOT LOGIN /dev/console
Sun Microsystems Inc.   SunOS 5.8       Generic February 2000

Enterprise Engineering Lab JumpStart
OS installation date: Thursday December 20 15:45:14 PST 2001

barossa# df -k
Filesystem            kbytes    used   avail capacity  Mounted on
/dev/dsk/c0t1d0s0    6196234  832525 5301747    14%    /
/proc                      0       0       0     0%    /proc
fd                         0       0       0     0%    /dev/fd
mnttab                     0       0       0     0%    /etc/mnttab
swap                 2493096      16 2493080     1%    /var/run
swap                 2493096      16 2493080     1%    /tmp
```

After switching over to the Solaris8-1001_Prime BE and validating its functionality, the Solaris8-1001 BE may be deleted (using the ludelete command), as follows.

```
barossa# lustatus
BE_name              Complete  Active  ActiveOnReboot  CopyStatus
-------------------------------------------------------------------
Solaris8-1001        yes       no      no              -
Solaris8-1001_Prime  yes       yes     yes             -
barossa# ludelete Solaris8-1001
Determining the devices to be marked free.
Updating BE configuration database.
Updating compare database.
Updating compare database on other BEs.
Updating BE configuration database on all BEs.
Boot environment <Solaris8-1001> deleted.
barossa# lustatus
BE_name               Complete  Active  ActiveOnReboot  CopyStatus
-------------------------------------------------------------------
Solaris8-1001_Prime   yes       yes     yes              -
```

The disk space used by the Solaris8-1001 BE may now be reclaimed.

Summary

This article introduced and reviewed LU, and provided techniques and best practices for LU usage. LU is a valuable tool that provides a mechanism for upgrading the Solaris OE with minimal downtime. Further, LU enables large scale changes to the Solaris OE 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.

The next article in this series will explain methods for using LU 2.0 with a logical volume manager, such as VxVM or Solstice DiskSuite software.

Acknowledgements

The author would like to thank Gary M. Gere and Mike Arwood for their technical comments and suggestions on this article.

Author's Bio: John S. Howard

John S. Howard is a Senior Staff Engineer with Sun's Enterprise Engineering group. He has over 19 years experience in software engineering and systems administration on a diversity of platforms. John is currently working on projects for enhancing system availability and serviceability.

John is the author of numerous technical papers and co-author of the books JumpStart Technology: Effective Use in the Solaris Operating Environment and Boot Disk Management: A Guide for the Solaris Operating Environment.