

Best Practices for Migrating to the Solaris™ 9 Operating System

A Case Study at Sun — Upgrading 1500 Servers to the Solaris 9 Operating System
December 2003



Table of Contents

| | |
|---|-----------|
| Executive Overview | 1 |
| A Successful Strategy | 1 |
| Scope of This Document | 2 |
| Introduction | 3 |
| The Challenge | 3 |
| A Traditional Approach Would Not Accomplish the Goal | 3 |
| Sun's Strategy and Process | 5 |
| Sun's Migration Strategy | 5 |
| Benefits and Risks of Sun's Strategy | 6 |
| Key Technologies in Sun's Strategy and Implementation | 6 |
| Solaris Flash Technology | 7 |
| Solaris Live Upgrade Technology | 7 |
| A Four-Stage Process | 7 |
| Stage 1 — Taking an Inventory of the Data Center Environment | 8 |
| Stage 2— Creating a Master Solaris 9 OS Image | 8 |
| Stage 3— Migrating to the New OS | 9 |
| Stage 4— Re-establishing Disk Redundancy and Removing the Fallback Boot . . . | 11 |
| Program Management and Problem Resolution | 12 |
| Documentation and Tools Available for Download | 13 |
| Achieving Results | 15 |
| Delivering Business Benefits | 15 |
| Minimizing Application Change | 16 |
| Conclusion | 17 |
| For More Information | 17 |

Chapter 1

Executive Overview

With businesses facing shrinking IT budgets and the need to stretch resources further, the idea of spending money on updating the operating systems of data center servers is often overlooked. Budgets are tight and only a small fraction of the IT projects that make it to the drawing board ever get approved and executed. What's often overlooked in an IT infrastructure project, however, is that the benefits can go well beyond the cost savings that are typically used to justify it.

Sun's recent project to migrate 1500 servers to the Solaris™ 9 Operating System (OS) was justified based on simplifying data center operations, reducing costs, and improving service levels. Sun recognized that the cost of managing its internal IT infrastructure could be significantly reduced by taking advantage of some of the new technologies available in the latest version of its operating system, the Solaris 9 OS. With all of Sun's servers running the Solaris 9 OS, Sun's data center infrastructure could provide better security and more flexible provisioning of resources. The servers could then be more fully utilized and user service levels could be maintained more easily during spikes in user demand.

In early 2003, a Sun management initiative was put forth to migrate Sun's internal servers to the Solaris 9 OS within a six-month time period. Because of Sun's size, and the fact that it operates multiple data centers around the world, this project to upgrade 1500 servers to the Solaris 9 OS seemed daunting at first. The project team had limited resources, a tight six-month time schedule, and an unknown number of applications that were not "supported" on Solaris 9 OS.

A Successful Strategy

Faced with this challenge, the Sun IT organization developed a new and innovative approach to system upgrades. The project team created a cost-effective method to rapidly upgrade systems

with minimal downtime. They were also able to limit the need for application testing to save costs without compromising user quality of service.

Sun's project was based on a few key assumptions that turned out to be a good basis for the project strategy. The key assumptions built into the project strategy were:

- Most applications would just work on the new OS and extensive testing would not be necessary
- The cost of additional hardware for test configurations could be avoided without incurring too much extra risk
- Investing engineering time up front on developing automated programs to simplify the migration process would pay off in terms of big gains in efficiency

Sun IT estimates that it could have spent well over \$100M in testing, had it used traditional testing for all of the applications and servers that were migrated. This would have made the project nearly impossible to justify. By testing only mission-critical applications and providing a fallback environment (dual boot environment) so that users could trust the process, Sun IT was able to avoid the high cost of installing additional test configurations and the engineering effort that would have been required for testing.

The use of automation in Sun's process provided big gains in efficiency. Not only was it much faster to migrate each server using automated programs, but this process automation also resulted in less variation of the OS configurations that were deployed. This simplified data center environment now has the potential to reduce the cost of Sun's ongoing data center operations.

Sun's strategy can be summarized as follows:

- Utilize Solaris Live Upgrade and Flash Archive features to improve efficiency and minimize downtime
- Provide dual boot capability on servers to help minimize risk of application failure
- Avoid, where possible, the need for system administrator input during the migration process to improve efficiency and help increase standardization of server configurations

Within a six-month time period, the Sun IT project team completed the migration of nearly 1500 servers to the Solaris 9 OS with only 43 application defects identified during that time period. Approximately one thousand applications were migrated and greater than 99 percent of them worked without modifications to the code or to the server environment, a testimony to the binary compatibility of the Solaris OS.

Scope of This Document

This document provides a case study of Sun's internal project to migrate its servers to the Solaris 9 OS. It also serves as a best practices guide to organizations that want to capitalize on the benefits of the Solaris 9 OS, but have lacked a methodology to quickly and effectively migrate their existing application workload to the new operating system. Sun's methodology is explained in detail, including Sun's assumptions, the strategies used to minimize risk, and classification of defects found. The document also provides links to additional information such as how to download project documentation and the programs used by Sun to automate its process.

The intent of this paper is to empower businesses to leverage Sun's migration process to implement their own smooth and trouble-free migration to the Solaris 9 OS.

Chapter 2

Introduction

The Challenge

With more than 1500 servers running business systems and applications in data centers across the globe, it was important for Sun to standardize on a common Solaris Operating System configuration. Sun wanted to bring all of its data centers up to the current OS release to help simplify data center operations and to begin taking advantage of Solaris 9 OS technologies designed to drive down data center costs and improve services levels.

Many of the systems to be upgraded were four to seven years old, running OS versions that were several releases behind, and containing applications that may not have been updated since the systems were first installed. In fact, some of the applications were no longer supported by their vendors regardless of what operating system they were running on. Others were certified on a prior release of the Solaris OS, but were not yet certified to run on Solaris 9 OS. The migration process would therefore need a quick and easy way to bring back the old application environment in order to make a support call to a third-party software vendor.

A Traditional Approach Would Not Accomplish the Goal

Sun's IT staff recognized that traditional methods of upgrading these 1500 servers would be prohibitively expensive in terms of system administrator time and downtime for applications. A standard OS upgrade could take up to several hours per server and application testing could be an even greater challenge. To add to the complexity of the challenge, Sun IT staff had been given only six months in which to complete the migration of these servers to the Solaris 9 OS.

A partial inventory of 530 servers was used to determine how many applications would need to be migrated to the new operating system. Sun used automated programs to look for executable

files on these servers and found more than 23,000 unique executable files, excluding system files. Because larger applications often have many executable files, the actual number of applications was estimated to be only one thousand. However, it was clear that even a thousand applications would be time consuming and difficult to qualify. Tracking them down and testing them to confirm that each application would run unchanged on the new OS would be cost-prohibitive. The team needed a process that could scale to upgrade 1500 servers and migrate an unknown number of applications.

In summary, the methodology to be used would need the following characteristics:

- *Quick implementation* — The six-month deadline must be met
- *Minimal application downtime* — Servers must be upgraded without major disruption to users
- *Scalable process* — OS upgrades on 1500 servers could not take several hours each
- *Dependable fallback mechanism* — Users needed a way to quickly get to an environment that was supported by their third-party software vendor
- *Minimal application testing* — Applications must be assumed to work in the new environment
- *Effective program management* — The process must be well managed using a process that simplifies isolation of potential failures and enables rapid determination of root causes

Sun's approach to addressing these requirements is summarized in Table 2-1.

Table 2-1: Sun's approach to addressing key project requirements

| Project Requirement | Sun's Approach |
|---|--|
| Quick implementation and scalable process | Deploy a <i>standard OS configuration</i> for all application servers in the data center using Solaris Flash technology and <i>automated programs</i> for a seamless process that would help minimize system administrator effort and create a consistent environment throughout the data center. |
| Minimal application downtime | Use Solaris Live Upgrade technology to install and configure the Solaris 9 OS on an <i>alternate boot device so that the old system could continue to operate</i> during the upgrade process. Using this approach, application downtime is often reduced to the time required for a reboot. |
| Dependable fallback mechanism | Deploy a <i>dual boot environment</i> during the fallback period so that the server could be easily booted with either the Solaris 9 OS or the previously installed OS version. This would enable application software problems to be easily reproduced on the previous OS platform in case a support call must be placed to an Independent Software Vendor (ISV). |
| Minimal application testing | Trust the binary compatibility capacity of the Solaris 9 OS to enable applications to run unchanged in the new environment and <i>limit testing to mission-critical applications</i> . Other applications would be assumed to work unless a problem was reported. |
| Effective program management | Create a <i>dedicated project team assigned to manage incidents and track down root causes of errors</i> reported by users. User errors or process problems could be addressed immediately by this team without going through the OS product support team, resulting in faster response. |

Sun's migration process was designed to reduce the manual effort required for server installation and patching by automating as many of the steps as possible. Automation would not only accelerate the project implementation schedule, but also would standardize on a single OS configuration, thereby simplifying system administration and potentially improving service levels.

Chapter 3

Sun's Strategy and Process

Sun's Migration Strategy

Because Sun has multiple data center locations and varied support personnel administering systems, it was important to standardize on a common Solaris Operating System configuration to simplify data center operations. To achieve this, Sun defined a process to analyze, build and archive a master Solaris 9 OS image that could be used as a starting point for all server upgrades.

Sun's migration process was based on a strategy that involved using a second boot drive on each server so that the server could be booted either from its old environment, or from the new environment using the Solaris 9 OS (see Figure 3-1). Most servers that were being upgraded started with a disk configuration that included a mirrored system disk for higher availability. Disk mirroring would be temporarily turned off so that the mirrored disk could be converted to an alternate boot disk that would contain a boot environment for the Solaris 9 OS. The migration process assumed that the mirrored disk was available and would break the mirror to prepare the server for migration. Then it installed the Solaris 9 OS on the disk that was formerly the mirror disk. The automated programs that performed this task also allowed for manual input from the system administrator to select an appropriate disk for the new OS in cases where disk mirroring was not in place.

The migration process included a fallback period in which servers would maintain dual boot disks while running from the new Solaris 9 OS boot disk. This would allow for a quick return to the previously installed OS environment if a problem was encountered. The fallback period was ended at different stages depending on the server's purpose. Production servers maintained a fallback period for 15 to 30 days, whereas test systems would maintain a fallback period until all software had been certified on the Solaris 9 OS. This would allow test servers to reproduce on the old OS,

any problems that might be encountered on a production server running the Solaris 9 OS. By reproducing the problem on the supported OS, users could then get support from third-party software vendors whose software had not yet been certified on the Solaris 9 OS.

When the dual boot environment was no longer needed, disk mirroring would be turned back on, and the old boot environment would be overwritten to add redundancy by mirroring the Solaris 9 OS boot disk.

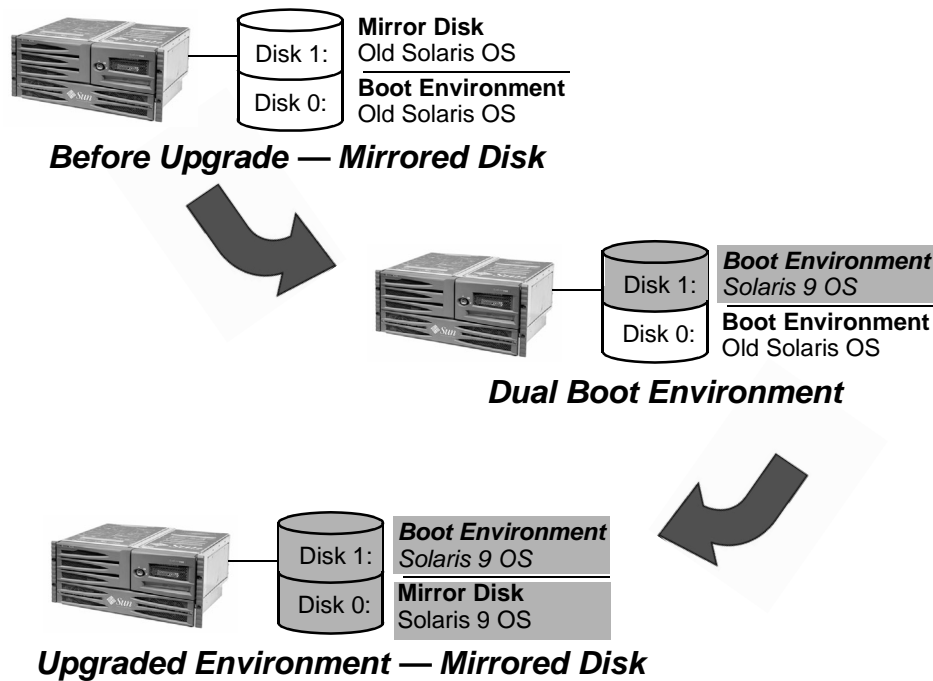


Figure 3-1: A dual boot environment enabled users to boot either OS version, providing a fallback to the old OS if necessary.

Benefits and Risks of Sun's Strategy

The major benefit of this approach was that it circumvented the need to run a separate test server. Secondly, it provided users with a fallback position, so that if they were to have a problem during the fallback period, they could quickly revert to their old environment. This meant that if their third-party software application were not yet qualified on Solaris 9 OS, they could make a quick transition to a supported environment by booting the previously installed release of the OS. Then they could place a support call to their software vendor.

The accepted risk of this approach was that it left the server without OS disk redundancy during the test period. Given that this approach prevented the need to install additional disks or run a separate test server, it was deemed a good trade-off to simply turn off disk mirroring during the test period. If spare disk space were available on the server, the administrator could also choose a different disk for the Solaris 9 OS boot disk rather than turning off the disk mirroring for the current OS, thus alleviating this risk.

Key Technologies in Sun's Strategy and Implementation

Sun's strategy for the Solaris 9 migration project was dependent on two key technologies, Solaris Flash and Solaris Live Upgrade. These technologies and their benefits are described briefly in the following subsections.

Solaris™ Flash Technology

Sun's in-house experience with Solaris Flash technology had shown that it could significantly reduce installation time when setting up a new Sun server or modifying an existing server. Internal tests conducted at Sun showed that a complete Web server could be set up using the network install feature of Solaris Flash software in under three minutes.

Solaris Flash technology would be used in this project to install a common Solaris 9 OS configuration throughout the data center. This method of replicating a single Solaris 9 OS image can reduce configuration complexity and improve deployment scalability. This process is ideal for migrating a large number of servers and can provide the following major benefits:

- Greatly speeds up the process of installation
- Creates efficiency and avoids duplication of efforts in configuring software on each server
- Minimizes the risk of errors in the installation by deploying a proven and tested software stack
- Simplifies overall system management by helping to ensure servers are identically configured

For more information on Solaris Flash technology, visit <http://www.sun.com/software/solaris/webstartflash/>.

Solaris Live Upgrade Technology

Solaris Live Upgrade technology can also be used with Solaris Flash technology to simplify the task of upgrading servers and help minimize downtime. Solaris Live Upgrade software promotes greater availability by providing a mechanism to upgrade and manage multiple on-disk instances of the Solaris Operating Environment — allowing OS upgrades to take place while the system continues to operate.

With Solaris Live Upgrade software, the updates are made to an alternate boot environment while the system continues to run using the active boot environment. The original system configuration remains functional, and in operation, until the alternate boot environment is activated and the server is rebooted. The updated software stack goes into effect immediately upon system reboot.

The primary benefits of Solaris Live Upgrade technology are:

- Reduces the downtime necessary in an operating system upgrade to that of a simple reboot
- Provides a fallback position, allowing the previous boot environment to remain available in case of an urgent need to revert to the prior software stack.

More information on Solaris Live Upgrade software is available at <http://www.sun.com/software/solaris/liveupgrade/>.

A Four-Stage Process

Sun's upgrade process can be divided into four primary stages that are described in the succeeding sections. The primary stages are:

- Taking an inventory of the data center environment to define the needs of the OS configuration
- *Creating a master* Solaris 9 OS configuration using Solaris Flash technology
- *Migrating each server* to the new Solaris 9 OS configuration
- *Re-establishing disk redundancy* and removing the fallback boot environment

Stage 1 — Taking an Inventory of the Data Center Environment

Sun's first step was to take an inventory of existing servers in the data center to help determine the requirements for master server configurations. Sun's data centers included a mix of Sun servers of different architecture types (e.g. Sun4u). Multiple master servers would therefore be needed because the OS installation options selected had to reflect the underlying architecture. Servers of the same architecture could use the same OS image even if they were different model servers. For example, a low-end server such as the Sun Enterprise™ 250 Server would use the same OS image as the much larger Sun Fire™ 15K server because they were both based on Sun4u architecture, built on the UltraSPARC® central processing unit.

In addition to architecture type of the target servers, other things to consider in taking the inventory included:

- Types of hardware interface, storage, and peripherals that are attached to servers
- Other types of system software such as a third-party volume manager that is used on some servers
- Common infrastructure software modules required (Sun IT could build the required client modules into the standard OS configuration.)
- OS components that are not needed on any servers and can therefore be removed from the master system
- Current required patches
- Minimum required OS disk partition size to utilize standardized OS disk layout

Most of this information was collected in an automated fashion by running Sun™ Explorer Data Collector on each server and then analyzing the system information contained in the output. After collecting and analyzing the data about the current server environment, the necessary OS requirements were known. The final output of this first stage was a documented configuration for the new OS, and a list of target server architectures on which it was to be deployed.

Tip – Sun Explorer Data Collector is a data collection and diagnostic tool that is available for download through the SunSolve Web site at <http://sunsolve.sun.com/>.

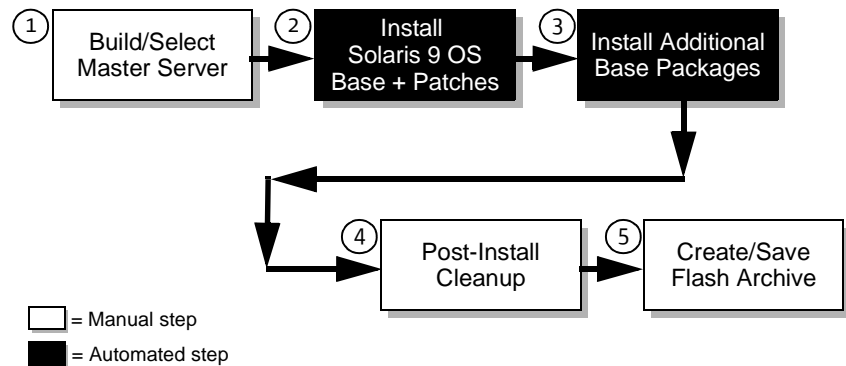
The next step involved creating a master OS image for each architecture type (e.g. Sun4u) represented in the data center.

Stage 2— Creating a Master Solaris 9 OS Image

Given Sun's desire to deploy a single common configuration of the OS throughout its data centers, the first step was to install and configure the OS on a master server. This server would contain the standard "base" Solaris 9 OS with appropriate packages and patches installed, allowing that same configuration to then be replicated on other servers.

Based on the inventory of the data center and the documented needs for the OS configurations, Sun IT selected a master server on which to build the standard OS configuration. Figure 3-2 shows the steps to configuring this master server and then creating and saving a Solaris Flash archive image for replication. The first step was simply preparing the hardware system for an OS installation and configuring the system to meet the minimum requirements for the installation.

Figure 3-2: Creation of a master OS image using Solaris Flash technology



The installation of the Solaris 9 OS and additional base packages (steps 2 and 3) was an automated process once Solaris JumpStart™ software profiles were defined, whereas other steps required the manual effort of a system administrator. The post-install cleanup (step 4) was also a manual process. In step 4, the system administrator deleted all unwanted OS components and installed the required patches.

Once the master server was configured, the Solaris 9 OS image could be saved using Solaris Flash feature of the Solaris 9 OS (step 5) to capture a snapshot image of the complete server configuration including all installed applications, OS patches, etc. In this instance, the master server contained no user applications, so the Solaris Flash archive image captured only the desired configuration of the Solaris 9 OS. Localization of target servers (adding additional Solaris packages beyond those configured in the master server) could be performed on individual servers after installing the master Solaris Flash image. The process for installing a Solaris Flash image on a target server is explained in more detail in the following section.

The end result of stage 2 was a system image that could be quickly and easily replicated on other servers to deploy a common OS configuration throughout the data center.

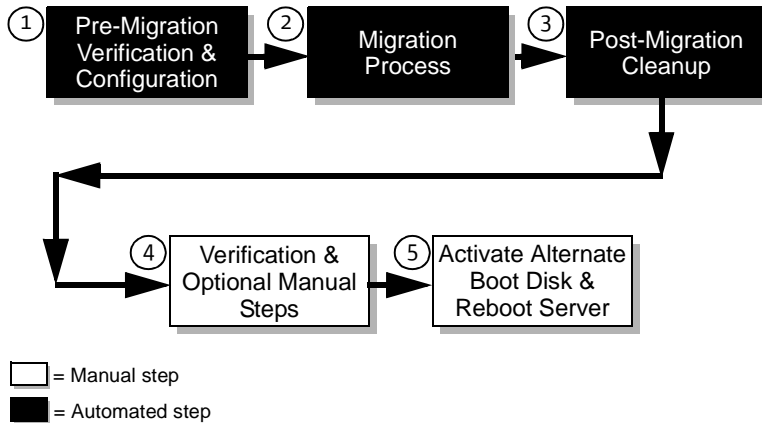
Stage 3— Migrating to the New OS

In order to complete the migration of 1500 servers in the desired six-month time period, Sun needed a highly efficient process that used a high degree of automation so that it would also result in a consistent server architecture. Using the process defined below, Sun was able to reduce the time required for migrating a server to less than an hour compared to traditional upgrade times of four to eight hours.

Sun's migration development team developed automated programs that would allow a system administrator to complete each of the major steps of the migration process by simply typing in a single command to initiate the program. All of the default configuration options were handled within these programs. This helped drive efficiency and speed as well as creating a consistent OS configuration throughout the data center. The major steps in the migration process are shown in Figure 3-3. Note that the first three boxes represent steps that are automated, leaving only two minor steps at the end that required operator interaction to complete the step.

The first step, called pre-migration, would bring the server being migrated up to the minimum requirements for the Solaris 9 migration process. If the server could not be brought up to the minimum requirements through software updates, the program would abort the process and require the administrator to restart the program after resolving the indicated issue.

Figure 3-3: Migrating a server to the Solaris 9 OS



The first step for the automated pre-migration process initiated the following actions:

- Verify that current OS level meets requirement (SunOS 5.6, 5.7, or 5.8)
- Validate minimum requirements for memory and OS disk space per the Solaris 9 Installation Guide and in-house standards
- Check for installation of required packages and patches for the migration process (e.g. Solaris Live Upgrade)
- Install packages and patches as necessary
- Break mirror of OS disk (The administrator can bypass this step if another disk is used for the Solaris 9 OS boot disk.)
- Partition Solaris 9 OS disk to recommended sizes

The second step, called the migration process, involved setting up the Solaris 9 OS disk partition by installing the Solaris Flash archive image containing the standard OS configuration. Again this step was automated and could be executed by a single command. It performed the following functions:

- Create file system and boot environments on target disk partition
- Install pre-built standardized Solaris OS image on alternate boot disk
- Create Solaris 9 OS required mail user
- Create mount points for file systems
- Synchronize alternate boot environment with devices and file system (Solaris Live Upgrade synchronizes some additional system files when the boot environment is activated and the server rebooted)

The third step, post-migration cleanup, checked for inconsistencies between the old and the new boot environment and prepared a validation report so that a system administrator could easily determine if any additional steps needed to be taken before activating this new boot environment. The compact size of the validation report made it easy to identify the required manual actions (step 4).

The final step was to manually activate the alternate boot environment (Solaris 9 OS) and reboot the server. Solaris Live Upgrade technology allowed the server to continue operating during the installation process, but this final step required some downtime. However, the downtime was generally limited to only the time required for the reboot of the server. Because

this step would impact the user community, this final step of the migration needed to be scheduled in advance so as to minimize the impact to users.

Stage 4— Re-establishing Disk Redundancy and Removing the Fallback Boot Environment

Given that Solaris 9 OS guarantees binary compatibility for applications that run on previous versions of Solaris, it was decided that it was a reasonable risk to assume that non-mission critical applications would just work. Sun performed testing on mission-critical applications, and other applications were simply migrated to the new environment and assumed to work unless a problem was reported.

Non-Mission-Critical Applications

Applications that were not considered mission-critical, were set up in a dual boot environment as explained at the beginning of this chapter, allowing the previously installed OS to be quickly and easily reasserted if necessary. Rather than performing full testing of these applications, Sun IT simply observed their behavior during the fallback period. The dual boot environment was considered no longer necessary when all applications on a given server ran without incident for a fallback period of 15 to 30 days.

One of the biggest obstacles to this approach was convincing users that their applications would work without full testing. To help overcome this obstacle, the Sun project team kept good records about application failure rates so that after a small percentage of applications had been migrated, there was good data to support the project team's claims that most user applications would just work on the Solaris 9 OS, and that full testing would be too expensive and unnecessary.

Mission-Critical Applications

Mission-critical applications were tested using their existing test processes. For example, Sun's Enterprise Resource Planning (ERP) system had a test configuration in place and had test procedures that were designed for testing new releases of the ERP software before they would be put into production. A development configuration was also in use for the development of software updates or enhancements to the system.

Sun's process for migrating each of these configurations to the Solaris 9 OS involved maintaining the development system on the old OS until the third-party software applications were actually certified for use on the Solaris 9 OS. The existing test configuration would be the first configuration to use the Solaris 9 OS and would be migrated to a dual boot environment with the Solaris 9 OS disk established as the active boot disk, and the old OS still available on the inactive boot disk. This configuration would be run for a period of 15 to 30 days, and if it ran without incident, the production system would have disk mirroring turned on again to add redundancy for the Solaris 9 OS disk.

The production configuration would also be maintained in a dual boot environment for a short period to make sure that it would have a quick fallback environment if a problem were encountered. After the production configuration had run on the Solaris 9 OS for a period of 15 to 30 days without incident, its disk mirroring would then be turned on again, removing the ability to immediately fallback to the old OS on the production server.

The process for migrating each of the configurations of a mission-critical application environment to the Solaris 9 OS is summarized in Table 3-1.

Table 3-1: Migration stages for development, test, and production configurations of mission-critical applications

| | Existing OS | First 15-30 Day Period | Next 15-30 Day Period | Steady State Until Software Gets Certified | After Certification on Solaris 9 OS |
|---------------------------|-------------|------------------------|-----------------------|--|-------------------------------------|
| Development System | Old OS | Old OS | Old OS | Old OS | Solaris 9 OS |
| Test System | Old OS | Dual boot ^a | Dual boot | Dual boot | Solaris 9 OS |
| Production System | Old OS | Old OS | Dual boot | Solaris 9 OS | Solaris 9 OS |

a. In dual boot mode, Solaris 9 OS is the default active OS. The old OS remains on the inactive boot drive where it can be quickly converted to the active OS if necessary.

The primary benefit of this approach is that it utilizes the existing processes for testing the application so that no new test configurations are required. The test configuration also remains available to be quickly rebooted to the old OS environment even after the production system has been migrated to the Solaris 9 OS. With this approach, there is a fallback position to reproduce a production problem on the old OS environment until third-party applications are fully supported or certified on the Solaris 9 OS.

Program Management and Problem Resolution

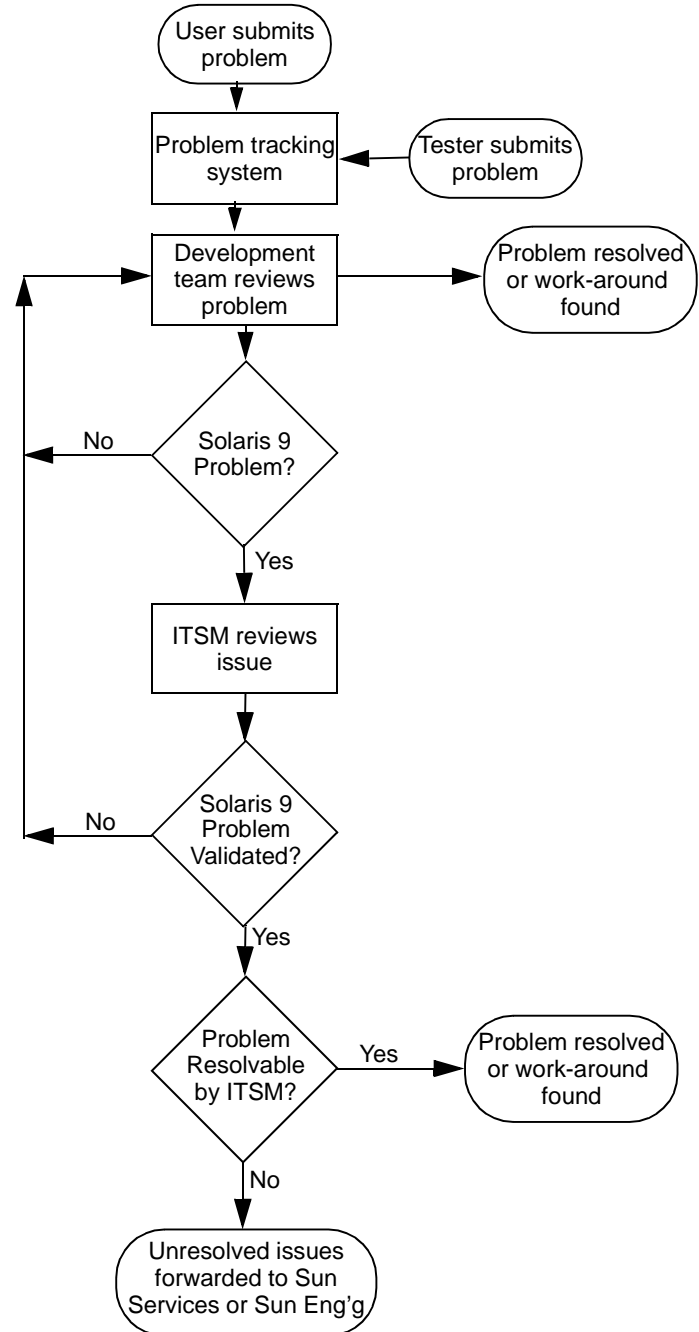
A project of this scale requires some forethought on how to manage logistics such as scheduling servers to be migrated, notifying users of planned outages, responding to user inquiries, and resolving reported software problems or process errors.

An independent project team was established and given responsibility for program management and user support during the migration process. The team members were experienced with Solaris system administration, application development, and database administration, and they acted as the first line of support for all problems that were reported during the Solaris 9 migration. They reviewed each submission to ensure all information was documented in the problem report, and determined if the problem was indeed related to the Solaris 9 OS migration. The team considered a problem to be related to the Solaris 9 OS migration if an application that worked under the old OS did not work with the Solaris 9 OS for any reason. If there was a need to recompile the code or change the environment by editing configuration files in order to make the application work on the Solaris 9 OS, it was considered to be a Solaris 9 migration problem. This rather broad definition of migration problems meant that the migration team was responsible for directly resolving most problem reports submitted by users. Existing Sun processes were leveraged for tracking and managing the problem reports.

Sun's IT Service Management (ITSM) organization became the line of support for all software escalations that were determined to be Solaris 9 related and not addressable by migration team. The ITSM team would verify the problem, and then attempt to resolve it. The Sun Services organization provided a third line of support for problems that could not be resolved by ITSM. If a problem was determined to be product-related and not previously documented, a Sun Services representative would also contact Solaris Engineering as another line of support.

The process for managing incidents reported by users is shown in Figure 3-4.

Figure 3-4: Incident management process for Solaris migration program



Documentation and Tools Available for Download

A key element of Sun's Solaris 9 migration strategy was the use of automated tools that would limit the interaction required by the system administrator. In addition to using Solaris Flash technology, Sun's deployment process included the use of automated programs for installation of the Solaris Flash image, for pre-migration setup, and for post-migration configuration steps. These automated programs saved a lot of time and resources during the installation process and also helped ensure consistency in the installations. Installation questions such as default location of the boot disk were answered the same way for each server unless the system administrator

installing that server chose to intentionally bypass the default answers in the automated programs.

The set of automated programs were designed such that each major step could be executed using a single command to run the master program for that step. Tables 3-2 and 3-3 describe the major tool sets and the project documentation that have been made available for download at <http://www.sun.com/bigadmin/content/sol9upgrade/>.

Table 3-2: Solaris 9 migration toolkit runtime components

| Component Name | Description |
|------------------------------|--|
| Run_S9_migrations_process.sh | Wrapper program which directs invocation and logging of the three main steps |
| s9_premigration.sh | Main program for Pre-Migration Validation programs |
| s9_premigration_prepare.sh | Validate the target system and record values |
| s9_parfacts.sh | Collect system configuration facts |
| s9_premigration_install.sh | Install patches and packages |
| s9_osdisk.sh | Perform disk maintenance (break-mirror, auto-partitioning) |
| s9_migration.sh | Install new boot environment and configure system files |
| s9_postmigration.sh | Validate target install disk and install the new boot environment |
| s9_mvpkg.sh | Transfer package definition directories between boot environments |
| cleanup_to_restart.sh | Restart process from pre-migration step |
| Other components | Library or include programs and other required data and configuration files |

Table 3-3: Solaris 9 migration project documentation

| Document Name | Description |
|--|--|
| Solaris 9 Migration Process Summary | Overview of the Solaris 9 Migration process methodology |
| Solaris 9 Migration Concepts | A detailed review of the migration process and technologies as well as functional descriptions of the automated tools used by Sun IT |
| Solaris 9 Migration Process Guide | Detailed step-by-step instructions for executing the Sun IT Solaris 9 Migration process |
| FAQ on the Solaris 9 Migration Process | Answers to frequently asked questions |

Chapter 4

Achieving Results

Sun's program was an overwhelming success. Nearly 1500 servers were migrated to the Solaris 9 OS in less than six months with a total of only 43 defects identified during that time period. Greater than 99 percent of the applications that were migrated worked without modifications and no problems were reported that were the result of actual Solaris Application Binary Interface (ABI) breakage. Not only was the project successful in terms of application compatibility, but it also met Sun's goals to perform the migration in a way that delivered maximum business benefit.

Delivering Business Benefits

The business case for Sun's migration project was justified based on simplifying data center operations and taking advantage of new technologies in the Solaris 9 OS to improve security, increase service levels, and reduce the ongoing cost of data center operations. The project was designed to deliver quick results while maintaining high standards for user service levels and building an environment that would simplify data center operations. The project delivered three primary business benefits:

- *User applications were largely unaffected* — Server downtime was limited to the time required for a server reboot and very few user applications experienced an incident that related to running the application on the Solaris 9 OS.
- *Business risk was minimized* — The use of a dual boot environment reduced the risk that an application failure would result in an extended outage by providing an easy way to revert to the previous configuration.
- *High efficiency helped minimize project cost and schedule* — Sun IT was able to reduce the time required for migrating a server to less than an hour compared to traditional upgrade times of four to eight hours.

Minimizing Application Change

All applications that were written to comply with the Solaris ABI worked without incident on the Solaris 9 OS. There were a total of 43 cases encountered where an incident occurred, such as finding a software bug or an application environment that required some modification in order to run on the Solaris 9 OS. More than half of the applications that required modification, required it for reasons that were documented in advance (22 of the 43 defects). For example, in some cases users had manually edited their autogenerated "sendmail.cf" files. These autogenerated files should never be modified directly, but instead changes should be made in the proper "cf" directory.

Table 4-1 shows the breakdown of the 43 defects according to five categories.

Table 4-1: Defects Identified in the Solaris 9 OS Migration Project

| Defect Category | Number of Defects | Defect Category Description |
|--|-------------------|--|
| Documented Changes | 12 | The application failed due to a documented change in the Solaris OS such as the need to modify a configuration file, a change in a utility's behavior, or an EOF removal from Solaris. |
| Unexpected Impact | 21 | A change in the Solaris OS that was not expected to affect any applications led to subtle interactions causing the application to fail. For example, increasing the Solaris default maximum number of open files per process induced failures in certain code paths. |
| Improper Application Code | 4 | The application was not written in a careful manner, but nevertheless worked on an earlier release of the Solaris OS. For example, the application may have been written with a dependency on something that was likely to change. |
| Incorrect Execution of Upgrade Process | 6 | Application failure occurred due to an incorrect upgrade procedure such as not applying required patches or updated kernel modules, or neglecting to carry forward special customizations. |
| TOTAL | 43 | |

Chapter 5

Conclusion

Sun's successful migration project demonstrates some of the key advantages of the Solaris Operating System. The sophisticated tools that are integrated into the operating environment, such as Solaris Flash and Solaris Live Upgrade technologies, can have a dramatic impact on the cost of operating a data center. Sun's migration project is also evidence that Sun's emphasis on binary compatibility of applications across releases of the Solaris OS has merit. This project demonstrated that applications that behave well in one release of the Solaris OS can be migrated to a later release without concern of introducing problems, and without having to do extensive testing. Hard data was collected, which shows that for Sun IT, more than 99 percent of its applications would run under the Solaris 9 OS without making any changes to either the server environment or the application.

For More Information

This white paper is part of Sun's commitment to make its own internal Solaris 9 adoption tools more useful to Sun customers. Additional information about these tools and technologies is available at the Web links listed in Table 5-1.

Table 5-1: Web links for additional information on Sun products and service offerings.

| Web Site URL | Description |
|---|---|
| http://www.sun.com/bigadmin/content/sol9upgrade/ | Solaris 9 migration project documentation and tools |
| http://www.sun.com/solaris | Solaris Operating System |
| http://www.sun.com/software/solaris/liveupgrade/ | Solaris Live Upgrade technology |
| http://www.sun.com/software/solaris/webstartflash/ | Solaris Flash Technology |

SUN™ Copyright 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054 U.S.A. All rights reserved.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and FAR 52.227-19.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

TRADEMARKS

Sun, Sun Microsystems, the Sun logo, Solaris, Solaris JumpStart, UltraSPARC, Sun Enterprise, and Sun Fire are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and or other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

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

THIS PUBLICATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS PUBLICATION COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THE PUBLICATION. SUN MICROSYSTEMS, INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS PUBLICATION AT ANY TIME.



Please
Recycle



Adobe PostScript

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 USA Phone 1-800-555-9786 or 1-800-555-9SUN Web sun.com



Sun Worldwide Sales Offices: Africa (North, West and Central) +33-13-067-4680, Argentina +5411-4317-5600, Australia +61-2-9844-5000, Austria +43-1-60563-0, Belgium +32-2-704-8000, Brazil +55-11-5187-2100, Canada +905-477-6745, Chile +56-2-3724500, Colombia +571-629-2323, Commonwealth of Independent States +7-502-935-8411, Czech Republic +420-2-3300-9311, Denmark +45 4556 5000, Egypt +202-570-9442, Estonia +372-6-308-900, Finland +358-9-525-561, France +33-134-03-00-00, Germany +49-89-46008-0, Greece +30-1-618-8111, Hungary +36-1-489-8900, Iceland +354-563-3010, India-Bangalore +91-80-2298989/2295454; New Delhi +91-11-6106000; Mumbai +91-22-697-8111, Ireland +353-1-8055-666, Israel +972-9-9710500, Italy +39-02-641511, Japan +81-3-5717-5000, Kazakhstan +7-3272-466774, Korea +822-2193-5114, Latvia +371-750-3700, Lithuania +370-729-8468, Luxembourg +352-49 11 33 1, Malaysia +603-21161888, Mexico +52-5-258-6100, The Netherlands +00-31-33-45-15-000, New Zealand-Auckland +64-9-976-6800; Wellington +64-4-462-0780, Norway +47 23 36 96 00, People's Republic of China-Beijing +86-10-6803-5588; Chengdu +86-28-619-9333; Guangzhou +86-20-8755-5900; Shanghai +86-21-6466-1228; Hong Kong +852-2202-6688, Poland +48-22-8747800, Portugal +351-21-4134000, Russia +7-502-935-8411, Singapore +65-6438-1888, Slovak Republic +421-2-4342-9485, South Africa +27 11 256-6300, Spain +34-91-596-9900, Sweden +46-8-631-10-00, Switzerland-German 41-1-908-90-00; French 41-22-999-0444, Taiwan +886-2-8732-9933, Thailand +662-344-6888, Turkey +90-212-335-22-00, United Arab Emirates +9714-3366333, United Kingdom +44-1-276-20444, United States +1-800-555-9SUN or +1-650-960-1300, Venezuela +58-2-905-3800