



What's New in Solaris Express 11/03

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 817-4103-03
November 2003

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

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

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

Sun, Sun Microsystems, the Sun logo, docs.sun.com, AnswerBook, AnswerBook2, iPlanet, Java, J2EE, JavaServer Pages, JSP, Enterprise JavaBeans, EJB, Java Naming and Directory Interface, JumpStart, Sun Fire, StarOffice, Sun Blade, Sun Ray, Netra, Ultra, Solaris JumpStart, and Solaris are trademarks, registered trademarks, or service marks 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. Netscape and Netscape Navigator are trademarks or registered trademarks of Netscape Communications Corporation. PostScript is a trademark or registered trademark of Adobe Systems, Incorporated, which may be registered in certain jurisdictions.

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.

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions.

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

Copyright 2003 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

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

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

Sun, Sun Microsystems, le logo Sun, docs.sun.com, AnswerBook, AnswerBook2, iPlanet, Java, J2EE, JavaServer Pages, JSP, Enterprise JavaBeans, EJB, Java Naming and Directory Interface, JumpStart, Sun Fire, StarOffice, Sun Blade, Sun Ray, Netra, Ultra, Solaris JumpStart, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc. Netscape est une marque de Netscape Communications Corporation. Netscape Navigator est une marque de Netscape Communications Corporation. PostScript est une marque de fabrique d'Adobe Systems, Incorporated, laquelle pourrait être déposée dans certaines juridictions.

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.



031110@6671



Contents

Preface 9

1	What's New in Solaris Express	13
	Key Features in Solaris Express	13
	Features by Release Date	13
	New Features in Solaris Express 11/03	14
	New Features in Solaris Express 10/03	14
	New Features in Solaris Express 9/03	14
	System Administration Tools	15
	DTrace Dynamic Tracing Facility	15
	Solaris Product Registry Enhancements	15
	Top-Down Volume Creation in Solaris Volume Manager	16
	SPARC: 64-bit Package Changes for System Administrators	16
	NIS-to-LDAP Transition Service	16
	Signed Packages and Patches	17
	System V IPC Configuration	17
	Optional Interval for <code>netstat</code> Command	19
	GMT-offset Quoted Time Zones	19
	BIND 8.3.3	19
	Solaris Patch Update Feature	19
	SPARC: Multiterabyte Volume Support in Solaris Volume Manager	20
	Reconfiguration Coordination Manager Support in Solaris Volume Manager	20
	Solaris Dynamic Host Configuration Protocol (DHCP)	20
	Combine Data Address and Test Address for Single IP Network Multipathing Group	21
	Security Enhancements	21

Basic Audit and Reporting Tool	21
IPsec and the Solaris Cryptographic Framework	21
Solaris Cryptographic Framework for System Administrators	22
Remote Audit Log	22
Kerberos Enhancements	23
FTP Server Enhancements	23
FTP Client	24
Pluggable Authentication Module (PAM) Enhancements	24
Internet Key Exchange (IKE) Key Storage on Sun Crypto Accelerator 4000 Board	25
IKE Hardware Acceleration	25
ipseckey Enhancement	26
Credential Propagation Over Loopback Connections	26
Solaris Privileges	26
Auditing Header Token Contains Host Information	27
Auditing Enhancements	27
New Audit Token <code>path_attr</code>	28
Password History Checking	28
Enhanced <code>crypt()</code> Function	28
Password Management Feature in <code>pam_ldap</code>	28
Networking Enhancements	29
Added IPv6 Functionality During Solaris Installation	29
IPv6 Temporary Addresses	29
SPARC: Sun ONE Application Server Integration	30
<code>sendmail</code> version 8.12 Uses TCP Wrappers	31
<code>routeadm</code> Command	31
SPARC: TCP Multidata Transmit	31
<code>ifconfig router</code> Option	32
Internet Protocol Version 6 (IPv6) Default Address Selection	32
Sun ONE Message Queue	33
Disable NFS and automount Services	33
Internet Protocol Version 6 (IPv6) 6to4 Router	34
Packet Tunneling Over IPv6	34
Hosting Multiple Web Sites on a Single Solaris Machine	34
IP Quality of Service	35
User Selector for Internet Protocol Quality of Service (IPQoS)	35
Routing Information Protocol Version 2 (RIPv2)	35
Development Enhancements	36

Dynamic Tracing Facility	36
Atomic Operations	36
Solaris WBEM File Changes	36
Solaris Cryptographic Framework for Developers	36
SPARC: 64-bit Package Changes for Software Developers	37
SPNEGO Pseudo-Mechanism for GSS-API Applications	38
SPARC: Locality Groups	38
Solaris Privileges for Software Developers	38
Thread Stacks in pmap	39
New DOOR_REFUSE_DESC Flag	39
Stack Check APIs	39
Enhanced crypt () Function for Software Developers	39
New Flags for madvise () Function	40
Linkers and Libraries Updates	40
Memory Allocation With libumem	40
Smart Card Terminal Interfaces	41
Smartcard Middleware APIs	41
Language Support Enhancements	41
Locale Administrator	41
Code Conversions for Internationalized Domain Name Support	42
Standard Type Services Framework	42
Auto Encoding Finder	42
SunIM-based Korean Language Engine With Auxiliary Window Support	43
Common Transliteration-based Input Method for All Indian languages	43
Unicode Version 3.2 Support	43
Additional Keyboard Support	44
Wubi Input Method	44
Input Method Support for Indic	44
Seven Additional Indic Scripts for Support in Unicode Locales	44
HKSCS-2001 Support in Hong Kong Locales	45
Solaris Documentation Changes	45
<i>Solaris Dynamic Tracing Guide</i>	45
<i>Application Packaging Developer's Guide</i>	45
<i>Programming Interfaces Guide</i>	45
<i>System Administration Guide: Basic Administration</i>	46
<i>Solaris Tunable Parameters Reference Manual</i>	46
<i>System Administration Guide: Network Services</i>	46
<i>Solaris 10 Installation Guide</i>	46

<i>Solaris WBEM Developer's Guide</i>	47
System Performance Enhancements	47
New Architecture for Network Stacks	47
Improved System Performance	47
UFS Logging Performance Enhancements	47
SPARC: Memory Placement Optimization (MPO)	48
SPARC: Dynamic Intimate Shared Memory (DISM) Large-Page Support	48
Installation Enhancements	48
Replacement of Solaris Web Start Command-Line Interface	48
SPARC: 64-bit Package Changes	48
WAN Boot Installation Method	49
Creating RAID-1 Volumes (Mirrors) With Custom JumpStart	49
Solaris Live Upgrade 2.1	50
Custom JumpStart Installation Method Creates New Boot Environment	50
Solaris Flash Archives	51
Solaris Flash Differential Archives and Configuration Scripts	51
Customizing Content in a Solaris Flash Archive	51
Command-Line Interface Enhancements to the Solaris Product Registry	52
Reduced Networking Software Group	52
Modifying Disk Partition Tables by Using a Virtual Table of Contents (VTOC)	53
x86: Change in Default Boot Disk Partition Layout	53
Support for Lightweight Directory Access Protocol (LDAP) Version 2 Profiles	53
Device Management	54
USB 2.0 Features	54
USB 2.0 Devices	54
Solaris Support for USB Devices	55
USB Mass Storage Devices	55
USB Driver Enhancements	57
The EHCI and OHCI Drivers	57
Logical Unit Reset	58
File System Enhancements	58
NFS version 4	58
Enhancements to the NFS Client	59
SPARC: Multiterabyte UFS File Systems	59
Device File System (devfs)	61
SPARC: Multiterabyte Volume Support With EFI Disk Labels	61

New Configuration File for the Autofs Environment	62
System Resources Enhancements	62
Extended Accounting Subsystem	62
Physical Memory Control Using the Resource Capping Daemon	63
System V IPC Resource Controls	63
Desktop Features	64
GNOME 2.0 Desktop	64
X11 Windowing Features	65
Xscreensaver Program	65
X Event Interception Extension (XEvIE)	65
FreeType 2.1.x	65
Xserver Virtual Screen Enhancement	66
Xrender Extension	66
Web Browser Enhancements	66
Netscape 7.0 for the Solaris Release	66
Freeware Enhancements	67
Ghostscript 7.05	67
New Freeware Packages: libxml2 2.4.16 and libxslt 1.0.19	67
ANT 1.4.1 Freeware Package	68

Preface

The *What's New in Solaris Express 11/03* describes all features that are in the Solaris™ Express 11/03 release.

The Solaris Operating System runs on two platforms, SPARC® and x86. The information in this document pertains to both platforms unless otherwise specified in a special chapter, section, note, bulleted item, figure, table, or example.

Key features in the Solaris Express releases include “DTrace Dynamic Tracing Facility” on page 15 and “New Architecture for Network Stacks” on page 47.

Who Should Use This Book

This book provides introductory descriptions of the new Solaris Express features for users, developers, and system administrators who install and use the Solaris Express 11/03 Operating System.

Optional Feature Licensing

Certain optional features and products that are described in this document may require individual licensing for use. Please refer to the Solaris™ Binary Code License Agreement and Supplemental License Terms.

Related Books

For further information about the features that are summarized in this book, refer to the following Sun documentation:

- *Application Packaging Developer's Guide*
- *Programming Interfaces Guide*
- *Solaris 10 Installation Guide*
- *Solaris Dynamic Tracing Guide*
- *Solaris Tunable Parameters Reference Manual*
- *Solaris WBEM Developer's Guide*
- *System Administration Guide: Basic Administration*

- *System Administration Guide: Network Services*

Related Third-Party Web Site References

Third-party URLs are referenced in this document and provide additional, related information.

Note – Sun is not responsible for the availability of third-party Web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Accessing Sun Documentation Online

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

Ordering Sun Documentation

Sun Microsystems offers select product documentation in print. For a list of documents and how to order them, see “Buy printed documentation” at <http://docs.sun.com>.

Typographic Conventions

The following table describes the typographic changes used in this book.

TABLE P-1 Typographic Conventions

Typeface or Symbol	Meaning	Example
AaBbCc123	The names of commands, files, and directories; on-screen computer output	Edit your .login file. Use <code>ls -a</code> to list all files. <code>machine_name%</code> you have mail.
AaBbCc123	What you type, contrasted with on-screen computer output	<code>machine_name%</code> su Password:
<i>AaBbCc123</i>	Command-line placeholder: replace with a real name or value	To delete a file, type rm <i>filename</i> .

TABLE P-1 Typographic Conventions (Continued)

Typeface or Symbol	Meaning	Example
<i>AaBbCc123</i>	Book titles, new words, or terms, or words to be emphasized.	Read Chapter 6 in <i>User's Guide</i> . These are called <i>class</i> options. You must be <i>root</i> to do this.

Numerals in parentheses next to a command or function name, for example, `ioctl(2)`, indicate the reference manual section number where the manual page ("man page") for the command or function is located.

Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P-2 Shell Prompts

Shell	Prompt
C shell prompt	<code>machine_name%</code>
C shell superuser prompt	<code>machine_name#</code>
Bourne shell and Korn shell prompt	<code>\$</code>
Bourne shell and Korn shell superuser prompt	<code>#</code>

What's New in Solaris Express

This book summarizes all features in the Solaris™ Express releases that are new or have been enhanced since the Solaris 9 Operating System (Solaris OS) was originally distributed in May 2002. The descriptions in this book cover features that are new in the current Solaris Express 11/03 release and in the prior Solaris Express 9/03 and 10/03 releases. Descriptions are also included for all features that were originally introduced in the Solaris Express pilot program. Among these enhancements are Sun™ Open Net Environment (Sun ONE) features.

Most of the features that are described in this book run on both the SPARC platform and certain x86 platforms. Those descriptions that are specific to one platform are labeled as either a SPARC feature or an x86 feature in the description title.

Key Features in Solaris Express

Key features in the Solaris Express releases include the DTrace dynamic tracing facility and a new architecture for network stacks. DTrace is a comprehensive dynamic tracing facility that gives Solaris users, administrators, and developers a new level of observability into the kernel and user processes. For further information, see “DTrace Dynamic Tracing Facility” on page 15. Also of particular importance, the networking stack for TCP connections has been rearchitected to deliver extremely high performance while improving the scalability. For further information, see “New Architecture for Network Stacks” on page 47.

Features by Release Date

This section lists feature titles by Solaris Express release date. Each feature title links to its full description. The remaining sections of this book include the feature descriptions, presented by feature type.

All features that are not listed in this section were originally introduced as part of the Solaris Express pilot program.

New Features in Solaris Express 11/03

New features in the Solaris Express 11/03 release include the following:

- “DTrace Dynamic Tracing Facility” on page 15
- “Solaris Product Registry Enhancements” on page 15
- “Basic Audit and Reporting Tool” on page 21
- “Added IPv6 Functionality During Solaris Installation” on page 29
- “IPv6 Temporary Addresses” on page 29
- “Locale Administrator” on page 41

Documentation changes in the Solaris Express 11/03 release include the new “*Solaris Dynamic Tracing Guide*” on page 45 and revisions to the “*Application Packaging Developer’s Guide*” on page 45.

New Features in Solaris Express 10/03

New features in the Solaris Express 10/03 release include the following:

- “Atomic Operations” on page 36
- “Code Conversions for Internationalized Domain Name Support” on page 42
- “New Architecture for Network Stacks” on page 47

The Solaris Express 10/03 release also includes revisions to the “*Programming Interfaces Guide*” on page 45 and the “*System Administration Guide: Basic Administration*” on page 46.

New Features in Solaris Express 9/03

New features in the Solaris Express 9/03 release include the following:

- “Top-Down Volume Creation in Solaris Volume Manager” on page 16
- “sendmail version 8.12 Uses TCP Wrappers” on page 31
- “IPsec and the Solaris Cryptographic Framework” on page 21
- “routeadm Command” on page 31
- “Standard Type Services Framework” on page 42
- “Auto Encoding Finder” on page 42
- “Replacement of Solaris Web Start Command-Line Interface” on page 48
- “Solaris Support for USB Devices” on page 55
- “The EHCI and OHCI Drivers” on page 57

Feature descriptions that have been revised in the Solaris Express 9/03 release include the following:

- “SPARC: Sun ONE Application Server Integration” on page 30
- “Solaris WBEM File Changes” on page 36
- “*System Administration Guide: Basic Administration*” on page 46
- “USB 2.0 Features” on page 54
- “USB 2.0 Devices” on page 54
- “USB Mass Storage Devices” on page 55
- “USB Driver Enhancements” on page 57

System Administration Tools

This section describes all system administration enhancements that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases. The DTrace dynamic tracing facility and the Solaris Product Registry are new features in the current Solaris Express release. The DTrace feature is of particular importance in Solaris Express.

DTrace Dynamic Tracing Facility

DTrace is a new feature of key importance in the Solaris Express 11/03 release.

DTrace is a comprehensive dynamic tracing facility that gives Solaris users, administrators, and developers a new level of observability into the kernel and user processes.

DTrace helps you understand your system by permitting you to dynamically instrument the operating system kernel and user processes to record data that you specify at locations of interest, called *probes*. Probes are like small programmable sensors scattered all over your Solaris system. More than 30,000 probes are included in this release. Each probe can be associated with custom programs written in the new D programming language. This association enables you to access system data by using ANSI C types and expressions and easily capture stack traces, record timestamps, build histograms, and more.

All of DTrace's instrumentation is entirely dynamic and available for use on your production system. When DTrace is off, you have absolutely no performance cost, and the performance impact of any tracing is limited to only those probes and actions that you enable. Moreover, DTrace is entirely safe: you can never damage the running system because DTrace has security, complete safety, and error checking at the core of its design. These features enable you to use DTrace with confidence on your running system whenever you need the facility to help investigate a problem.

The *Solaris Dynamic Tracing Guide* and a set of man pages are provided to help you learn DTrace. The guide includes a complete feature reference and examples to help you get started. You can also join the DTrace forum on BigAdmin (see <http://www.sun.com/bigadmin/content/dtrace>) to learn more about DTrace, download example programs, and ask questions.

Solaris Product Registry Enhancements

The feature has been enhanced in the Solaris Express 11/03 release.

The improved Solaris Product Registry upgrades the product registry and the package databases to enhance the performance of package and patch operations. This version of the Solaris Product Registry introduces the following new functionality:

- The `pkgadm` command includes two new subcommands: `upgrade` and `revert`. The old product registry and package database format that prior Solaris releases contained might be incompatible with the improved Solaris Product Registry database. The `upgrade` and `revert` subcommands enable you to switch between

the old and the new databases in cases where you need to maintain database compatibility between different Solaris releases.

See the `pkgadm(1M)` man page for further information.

- By using role-based access control (RBAC), nonroot users can install and uninstall packages and patches on file systems where the users have appropriate access. The modifications are stored with the user's installed data, not in the system install database.
- The `pkgchk` command provides a new option to assist you in mapping files to packages. To map files to packages, use the `pkgchk -P` option instead of this command: `grep <pattern> /var/sadm/install/contents`. The `grep` command is no longer valid because the `/var/sadm/install/contents` file no longer contains a live image of the package database.

The `-P` option enables you to use a partial path. Use this option with the `-l` option to list the information about the files that contain the partial path.

See the `pkgchk(1M)` man page for further information.

Top-Down Volume Creation in Solaris Volume Manager

This feature is new in the Solaris Express 9/03 release.

The top-down volume creation feature of Solaris Volume Manager provides a new command that enables system administrators to quickly and easily create quality of service-based Solaris Volume Manager configurations. Rather than manually partitioning disks, creating stripes, and assembling mirrors, the `metassist` command manages the details and provides functional logical volumes. The volumes are based on the criteria that are specified on the command line or in referenced configuration files.

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

SPARC: 64-bit Package Changes for System Administrators

This feature was introduced in the Solaris Express pilot program.

In Solaris software, packaging has been simplified with most 32-bit components and 64-bit components being delivered in a single package. For further information, see "SPARC: 64-bit Package Changes" on page 48.

NIS-to-LDAP Transition Service

This feature was introduced in the Solaris Express pilot program.

The NIS-to-LDAP transition service can be used to enable a network transition from using NIS to using LDAP as the primary naming service. By using this transition service, administrators can utilize the bundled Sun ONE Directory Server, which works with LDAP naming service clients.

Signed Packages and Patches

This feature was introduced in the Solaris Express pilot program.

The Solaris software enables you to securely download Solaris packages and patches that include a digital signature by using the updated `pkgadd` and `patchadd` commands. A package or a patch with a valid digital signature ensures that the package or patch has not been modified after the signature was applied to the package or patch.

In previous Solaris releases, you could only add signed patches to your system if you used the Solaris patch management tools with PatchPro 2.1.

Additional software management features in this Solaris release include the following:

- You can add a digital signature to a package with the updated `pkgtrans` command. For information about creating a signed package, see the *Application Packaging Developer's Guide*.
- You can download a package or patch from a HTTP or HTTPS server.

A signed package is identical to an unsigned package except for the signature. The package can be installed, queried, or removed with existing Solaris packaging tools. A signed package is also binary-compatible with an unsigned package.

Before you can add a package or patch with digital signatures to your system, you must set up a keystore with trusted certificates that are used to identify that the digital signature on the package or patch is valid.

For information about setting up the package keystore and adding signed packages or patches to your system, see the *System Administration Guide: Basic Administration*.

For information about booting and retrieving Solaris installation images from an HTTP or HTTPS server, see “WAN Boot Installation Method” on page 49.

System V IPC Configuration

This feature was introduced in the Solaris Express pilot program.

In the Solaris software, all System V IPC facilities are either automatically configured, or the facilities can be controlled by resource controls. These facilities may be shared memory, message queues or semaphores.

Resource controls allow IPC settings to be made on a per-project or per-user basis on the local system or in a name service environment.

In previous Solaris releases, IPC facilities were controlled by kernel tunables. You had to modify the `/etc/system` file and reboot the system to change the default values for these facilities.

Because the IPC facilities are now controlled by resource controls, their configuration can be modified while the system is running.

Many applications that previously required system tuning to function might now run without tuning because of increased defaults and automatic allocation of resources.

The following table identifies the now obsolete IPC tunables that are described in the *Solaris Tunable Parameters Reference Manual* and their replacement resource controls.

Resource Control	Obsolete Tunable	Old Default Value	Maximum Value	New Default Value
process.max-msg-qbytes	msginfo_msgmnb	4096	ULONG_MAX	65536
process.max-msg-messages	msginfo_msgtql	40	UINT_MAX	8192
process.max-sem-ops	seminfo_semopm	10	INT_MAX	512
process.max-sem-nsems	seminfo_semmsl	25	SHRT_MAX	512
project.max-shm-memory	shminfo_shmmax	0x800000	UINT64_MAX	1/4 of physical memory
project.max-shm-ids	shminfo_shmmni	100	2 ²⁴	128
project.max-msg-ids	msginfo_msgmni	50	2 ²⁴	128
project.max-sem-ids	seminfo_semmni	10	2 ²⁴	128

Obsolete parameters may be included in the `/etc/system` file on this Solaris system. If so, the parameters are used to initialize the default resource control values as in previous Solaris releases. However, using the obsolete parameters is not recommended.

The following related parameters have been removed. If these parameters are included in the `/etc/system` file on this Solaris system, the parameters are commented out.

<code>semsys:seminfo_semmns</code>	<code>semsys:seminfo_semvmx</code>
<code>semsys:seminfo_semmnu</code>	<code>semsys:seminfo_semaem</code>
<code>semsys:seminfo_semume</code>	<code>semsys:seminfo_semusz</code>
<code>semsys:seminfo_semmap</code>	<code>shmsys:shminfo_shmseg</code>
<code>shmsys:shminfo_shmmin</code>	<code>msgsys:msginfo_msgmap</code>
<code>msgsys:msginfo_msgseg</code>	<code>msgsys:msginfo_msgssz</code>

`msgsys:msginfo_msgmax`

For more information about using the new resource controls, see “System V IPC Resource Controls” on page 63.

Optional Interval for `netstat` Command

This feature was introduced in the Solaris Express pilot program.

The `netstat` command displays the contents of certain network-related data structures in various formats, depending on the options that you select. The `-s`, `-k`, `-i` and `-m` options now support an “interval.” With the specification of an optional interval, the output is continuously displayed in interval seconds until interrupted by the user. If a “count” is specified after “interval”, the output iterates for “count” times only. The “interval” and “count” must both be positive integers.

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

GMT-offset Quoted Time Zones

This feature was introduced in the Solaris Express pilot program.

The Solaris Install program installs a quoted time zone as the system default time zone, if a GMT-offset style time zone is selected. Previously, an unquoted time zone was installed.

Note that quoted time zones behave differently than unquoted time zones. When the time zone name is displayed by the `date` command, for example, quoted time zones display the offset from GMT. Unquoted time zones display only the string “GMT,” without the offset from GMT.

BIND 8.3.3

This feature was introduced in the Solaris Express pilot program.

In the Solaris Express releases, the version of BIND is now 8.3.3. DNS clients can now connect to IPv6 DNS servers by using IPv6 transport.

Solaris Patch Update Feature

This feature was introduced in the Solaris Express pilot program.

You can use the Update option of the Solaris Management Console’s Patches Tool to analyze, download, and install recommended patches from the SunSolve OnlineSM Web site. Or, the `smpatch` update command can be used for this purpose.

This Solaris update feature was previously only available for systems that run the Solaris 2.6, Solaris 7, or Solaris 8 releases.

You must install the PatchPro 2.1 software on your system before running the update feature. Download the PatchPro 2.1 packages from <http://www.sun.com/PatchPro>. Then follow the instructions to install the software on your system.

For more information, see the `smpatch(1M)` man page.

SPARC: Multiterabyte Volume Support in Solaris Volume Manager

This feature was introduced in the Solaris Express pilot program.

Solaris Volume Manager has been enhanced with multiterabyte volume support. With this support, the Solaris Volume Manager can create, manage, and delete large (>1Tbyte) RAID-0 (stripe), RAID-1 (mirror), RAID-5, and soft partition volumes. In addition, this large volume support enables Solaris Volume Manager to construct volumes on large or EFI-labeled logical unit numbers (LUNs).

Solaris Volume Manager large volume support is not available for systems that run a 32-bit Solaris kernel. For example, systems that run the Solaris release (x86 Platform Edition) or the Solaris release with the SPARC 32-bit kernel cannot use large volume support.

For further information, see “SPARC: Multiterabyte Volume Support With EFI Disk Labels” on page 61.

Reconfiguration Coordination Manager Support in Solaris Volume Manager

This feature was introduced in the Solaris Express pilot program.

Reconfiguration Coordination Manager (RCM) support adds to Solaris Volume Manager the ability to respond appropriately to dynamic reconfiguration (DR) requests. This addition ensures that removal of devices under Solaris Volume Manager control is blocked with an appropriate warning. This block remains in effect until the devices are no longer in use. This warning prevents system administrators from accidentally removing active volumes from a DR-configured system.

Solaris Dynamic Host Configuration Protocol (DHCP)

This feature was introduced in the Solaris Express pilot program.

The Solaris Dynamic Host Configuration Protocol (DHCP) service includes the DHCP Manager graphical tool and command-line tools. This Solaris release adds new options to the `dhcpconfig` and `dhtadm` commands.

With the new options for the `dhcpconfig` command, you can do the following:

- Enable and disable DHCP service for system starts
- Restart DHCP server
- Prompt DHCP server to read the `dhcptab`
- Specify DHCP server parameters

A new suboption for the `dhtadm` command causes the DHCP server to read the `dhcptab` to implement changes that were made with the `dhtadm` command.

The new options enable you to perform all DHCP management tasks from the command line as well as from DHCP Manager.

For more information, see the man pages for `dhcpconfig(1M)` and `dhtadm(1M)`.

Combine Data Address and Test Address for Single IP Network Multipathing Group

This feature was introduced in the Solaris Express pilot program.

The use of a dedicated test IP address for failure detection on single-adapter IP network multipathing groups is no longer required. Sometimes, failover is not possible because only one network information center (NIC) is in an IP network multipathing group. In this situation, you can now combine the test address and the data address. The `in.mpathd` daemon uses a data address to detect failure when a test address is not specified.

Security Enhancements

This section describes all security enhancements that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases. New features in the current release include the Basic Audit and Reporting Tool (BART).

Basic Audit and Reporting Tool

This feature is new in the Solaris Express 11/03 release.

Basic Audit and Reporting Tool (BART) is a command-line utility that enables OEMs, advanced users, and system administrators to do a check at file level of the software contents of a target system. The utility is useful for gathering information about what is installed on a system. BART also enables you to compare installed systems, and to compare the contents of a system over time.

For more information, see the `bart_manifest(4)`, `bart_rules(4)`, and `bart(1M)` man pages.

IPsec and the Solaris Cryptographic Framework

This feature is new in the Solaris Express 9/03 release.

IPsec uses the Solaris Cryptographic Framework instead of its own encryption and authentication modules. The modules are optimized for the SPARC platform. In addition, a new `ipsecalgs` command-line utility and APIs are provided to query the list of supported IPsec algorithms and other IPsec properties.

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

Solaris Cryptographic Framework for System Administrators

This feature was introduced in the Solaris Express pilot program.

The Solaris Cryptographic Framework provides cryptographic services to applications in the Solaris environment. The system administrator controls which encryption algorithms can be used through the `cryptoadm` command. The `cryptoadm` command enables you to perform the following functions:

- Manage available providers of cryptographic services
- Set cryptographic security policy, such as disabling algorithms from a particular provider

The framework comes with plugins for the AES, DES/3DES, RC4, MD5, SHA-1, DSA, RSA, and Diffie-Hellman algorithms. Plugins can be added or removed as needed.

The `encrypt`, `decrypt`, `digest`, and `mac` commands all use cryptographic algorithms from the framework.

See the following man pages for further information:

- `cryptoadm(1M)`
- `kcfd(1M)`
- `libpkcs11(3LIB)`
- `pkcs11_kernel(5)`
- `pkcs11_softtoken(5)`

Remote Audit Log

This feature was introduced in the Solaris Express pilot program.

In addition to recording audit events in the binary audit log, the Solaris Express releases enable you to record audit events to `syslog`.

The generation of `syslog` data allows you to use the same management and analysis tools that are available for `syslog` messages from a variety of Solaris and non-Solaris environments, including workstation, servers, firewalls, and routers. By using `syslog.conf` to route audit messages to remote storage, you protect log data from alteration or deletion by an attacker. However, the `syslog` option provides only a summary of audit record data. Also, when `syslog` data is stored on a remote system, the data is susceptible to network attacks such as denial of service and false or “spoofed” source addresses.

For setup and administration information, see the `audit(1M)`, `audit.log(4)`, `audit_control(4)`, `audit_syslog(5)`, `syslog(3C)`, and `syslog.conf(4)` man pages.

Kerberos Enhancements

This feature was introduced in the Solaris Express pilot program.

The Solaris Express software includes the following Kerberos enhancements:

- The Kerberos utilities, `kinit` and `ktutil`, are now based on MIT Kerberos version 1.2.1. This change adds new options to the `kinit` command and new subcommands to the `ktutil` command.
See the `kinit(1)` and `ktutil(1)` man pages for more information.
- Extensions to the password-changing utilities enable the Solaris Kerberos V5 administration server to accept password change requests from non-Solaris clients.
See the `kadmind(1M)` man page for more information.
- Kerberos protocol support is provided in remote applications, such as `ftp`, `rcp`, `rdist`, `rlogin`, `rsh`, and `telnet`.
See the man pages for each command and daemon and the `krb5_auth_rules(5)` man page, for more information.
- Support is provided for auto-discovery of the Kerberos KDC, admin server, `kpasswd` server, and host or domain name-to-realm mappings by using DNS lookups. This support reduces the steps that are needed to install a Kerberos client.
See the `krb5.conf(4)` man page for more information.
A new script is provided that helps automatically configure a Kerberos client. The Kerberos administrator can quickly and easily set up a Kerberos client by using this script.
See the `kclient(1M)` man page for more information.
- Additions to the `pam_krb5` module enable you to manage the Kerberos credentials cache by using the PAM framework.
See the `pam_krb5(5)` man page for more information.
- The Solaris Kerberos Key Distribution Center (KDC) is now based on MIT Kerberos version 1.2.1. The KDC now defaults to a btree-based database, which is more reliable than the current hash-based database.
See the `kdb5_util(1M)` man page for more information.
- Support is provided for Kerberos Triple DES (3DES) with SHA1 encryption type. This change increases security. In addition, this change enhances interoperability with other Kerberos implementations that support this encryption type.
See the `mech(4)` man page for more information.
- A new configuration file option makes the strict TGT verification feature optionally configurable on a per-realm basis.
See the `krb5.conf(4)` man page for more information.

FTP Server Enhancements

This feature was introduced in the Solaris Express pilot program.

Scalability and transfer logging enhancements have been made to the `in.ftpd` daemon including:

- The `sendfile()` function is used for binary downloads.
- New capabilities are supported in the `ftpaccess` file:
 - `flush-wait` controls the behavior at the end of a download or directory listing.
 - `ipcos` sets the IP Class of Service for either the control or data connection.
 - `quota-info` enables retrieval of quota information.
 - `recvbuf` sets the receive (upload) buffer size that is used for binary transfers.
 - `rhostlookup` allows or disallows the lookup of the remote hosts name.
 - `sendbuf` sets the send (download) buffer size that is used for binary transfers.
 - `xferlog format` customizes the format of the transfer log entry.
- A new `-4` option makes the FTP server only listen for connections on an IPv4 socket when running in standalone mode.

In addition, `ftpcount` and `ftpwho` now support the `-v` option, which displays user counts and process information for FTP server classes that are defined in virtual host `ftpaccess` files.

For further information about these changes see the `in.ftpd(1M)`, `ftpaccess(4)`, `ftpcount(1)`, `ftpwho(1)`, and `sendfile(3EXT)` man pages.

FTP Client

This feature was introduced in the Solaris Express pilot program.

The Solaris software includes a change to the FTP client. By default, a Solaris FTP client, connected to a Solaris FTP server, lists directories as well as plain files when the `ls` command is issued. If the FTP server is not running in the Solaris operating system, directories may not be listed.

To allow for the default Solaris behavior when connecting to non-Solaris FTP servers, the `/etc/default/ftp` file can be edited on each Solaris client. To make the change for individual users, the `FTP_LS_SENDS_NLST` environment variable can be set.

For more information, see the `ftp(4)` man page.

Pluggable Authentication Module (PAM) Enhancements

This feature was introduced in the Solaris Express pilot program.

The Solaris software includes the following changes to the Pluggable Authentication Module (PAM) framework.

- A new `binding` control flag which if successful, returns a success if no previous required modules have failed, and skips any additional checking. If the check fails, a required failure is recorded and the rest of the stack is processed.

See the `pam.conf(4)` man page for more information.

- The `pam_unix` module has been replaced by a set of equivalent service modules of equivalent or greater functionality. The replacement modules are as follows:
 - `pam_authtok_check`
 - `pam_authtok_get`
 - `pam_authtok_store`
 - `pam_dhkeys`
 - `pam_passwd_auth`
 - `pam_unix_account`
 - `pam_unix_auth`
 - `pam_unix_cred`
 - `pam_unix_session`

For more information, see the man pages for each PAM module.

- The functionality of the `pam_unix_auth` module was split into two modules. The `pam_unix_auth` module now verifies that the password is correct for the user. The new `pam_unix_cred` module provides functions that establish user credential information.

For more information, see the `pam_unix_auth(5)` and the `pam_unix_cred(5)` man pages.

- Additions have been made to the `pam_krb5` module to manage the Kerberos credentials cache by using the PAM framework.

For more information, see the `pam_krb5(5)` man page.

Internet Key Exchange (IKE) Key Storage on Sun Crypto Accelerator 4000 Board

This feature was introduced in the Solaris Express pilot program.

IKE now runs on IPv6 as well as IPv4 networks. For information about keywords that are specific to the IPv6 implementation, see the `ifconfig(1M)` and `ike.config(4)` man pages.

When a Sun™ Crypto Accelerator 4000 board is attached, IKE can offload computation-intensive operations to the board, thus freeing the operating system for other tasks. IKE can also use the attached board to store public keys, private keys, and public certificates. Key storage on a separate piece of hardware provides additional security.

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

IKE Hardware Acceleration

This feature was introduced in the Solaris Express pilot program.

Public key operations in IKE can be accelerated by a Sun Crypto Accelerator 1000 card. The operations are offloaded to the card. The offloading accelerates encryption and reduces demands on Solaris Operating System resources.

ipseckey Enhancement

This feature was introduced in the Solaris Express pilot program.

For network administrators who have installed IPsec or IKE on their systems, the `ipseckey` parser provides clearer help. The `ipseckey monitor` command now provides a timestamp for each event.

For more information, see the `ipseckey(1M)` man page.

Credential Propagation Over Loopback Connections

This feature was introduced in the Solaris Express pilot program.

This Solaris release introduces `ucred_t *` as an abstract representation of the credentials of a process. These credentials can be retrieved by using `door_ucred()` in door servers and `getpeerucred()` for loopback connections. The credentials can be received by using `recvmsg()`.

See the `socket.h(3HEAD)` man page for further information.

Solaris Privileges

This feature was introduced in the Solaris Express pilot program.

In the Solaris software, administrative tasks that previously required superuser rights now require appropriate privileges. Privileges restrict processes to only those privileges that are required to perform the current task. Therefore, fewer `root` processes are vulnerable to exploitation. The number of `setuid` programs has been greatly reduced.

As installed, the Solaris Express releases are completely compatible with previous releases of the Solaris Operating System in terms of the privileges enhancements. Unmodified programs that run as `root` run with all privileges.

Device Protection - Devices are protected with a security policy. The policy is enforced with Solaris privileges. Therefore, the permissions on a device file do not fully determine the device's availability. Privileges might also be required to operate the device.

System interfaces that were protected by UNIX[®] permissions are now protected by Solaris privileges. For example, members of the group `sys` are no longer automatically allowed to open the `/dev/ip` device. Processes that are running with the `net_rawaccess` privilege can access the `/dev/ip` device. When the system boots, access to all devices is restricted until the `devfsadm` command runs during the boot sequence. The initial policy is as strict as possible. The policy prevents all users except the superuser from initiating connections.

See the following man pages for more information:

- `getdevpolicy(1M)`
- `ppriv(1)`
- `add_drv(1M)`
- `update_drv(1M)`
- `rem_drv(1M)`

Processes that need to retrieve Solaris IP MIB information should open `/dev/arp` and push the “tcp” and “udp” modules. No privileges are required. This method is equivalent to opening `/dev/ip` and pushing the “arp”, “tcp” and “udp” modules. Because opening `/dev/ip` now requires a privilege, the `/dev/arp` method is preferred.

Auditing Header Token Contains Host Information

This feature was introduced in the Solaris Express pilot program.

The header token in audit records has been expanded to include the name of the host.

The old header is displayed as follows:

```
header,131,4,login - local,,Wed Dec 11 14:23:54 2002, + 471 msec
```

The new expanded header is displayed as follows:

```
header,162,4,login - local,,example-hostname,  
Fri Mar 07 22:27:49 2003, + 770 msec
```

Customized scripts or tools that parse `praudit` output might need to be updated to reflect this change.

Auditing Enhancements

This feature was introduced in the Solaris Express pilot program.

Enhancements to the audit features in the Solaris software reduce noise in the trail, and enable administrators to use XML scripting to parse the trail. These enhancements include the following:

- Public files are no longer audited for read-only events. The `public` policy flag for the `auditconfig` command controls whether public files are audited. By not auditing public objects, the audit trail is greatly reduced. Attempts to read sensitive files are therefore easier to monitor.
- The `praudit` command has an additional output format, XML. The XML format enables the output to be read in a browser, and provides source for XML scripting for reports. See the `praudit(1M)` man page.
- The default set of audit classes has been restructured. Audit metaclasses provide support for finer-grained audit classes. See the `audit_class(4)` man page.
- The `bsmconv` command no longer disables the use of the Stop-A key. The Stop-A event is now audited to maintain security.

New Audit Token `path_attr`

This feature was introduced in the Solaris Express pilot program.

The `path_attr` audit token contains access path information for an object. The access path specifies the sequence of attribute file objects below the path token object. Systems calls such as `openat()` access attribute files. For more information on extended file attributes, see the `fsattr(5)` man page.

The `path_attr` token has three fields:

- A token ID field that identifies this token as a `path_attr` token
- A count that represents the number of sections of attribute file paths
- One or more null-terminated strings

The `praudit` command displays the `path_attr` token as follows:

```
path_attr,1,attr_file_name
```

Password History Checking

This feature was introduced in the Solaris Express pilot program.

For login accounts that are defined in local files, a password history of up to 26 priorly changed passwords can be enabled. When a user changes a password, the attempt fails if the new password matches one of the passwords in the history. Also, login name checking can be disabled.

See the `passwd(1)` man page for more information.

Enhanced `crypt()` Function

This feature was introduced in the Solaris Express pilot program.

Password encryption protects passwords from being read by intruders. Three strong password encryption modules are now available in the software:

- A version of Blowfish that is compatible with Berkeley Software Distribution (BSD) systems
- A version of Message Digest 5 (MD5) that is compatible with BSD and Linux systems
- A stronger version of MD5 that is compatible with other Solaris systems

For information on the strength of the modules, see the `crypt_bsdbf(5)`, `crypt_bsmd5(5)`, and `crypt_sunmd5(5)` man pages.

Password Management Feature in `pam_ldap`

This feature was introduced in the Solaris Express pilot program.

The `pam_ldap` password management feature strengthens the overall security of the LDAP Naming Service when used in conjunction with the Sun ONE Directory Server (formerly iPlanet™ Directory Server). Specifically, the password management feature does the following:

- Allows for tracking password aging and expiration
- Prevents users from choosing trivial or previously used passwords
- Warns users if their passwords are about to expire
- Locks out users after repeated login failures
- Prevents users, other than the authorized system administrator, from deactivating initialized accounts

Networking Enhancements

This section describes all networking enhancements that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases. New features in the current release include additional IPv6 functionality and a new IPv6 Temporary Addresses feature.

Added IPv6 Functionality During Solaris Installation

This feature is new in the Solaris Express 11/03 release.

Several new IPv6 functions are started when you select Enable IPv6 during Solaris installation.

- The `/etc/nsswitch.conf` file policies for the hosts database and `ipnodes` repositories are synchronized so that the same naming repositories are searched for `ipnodes` and `hosts`. Now, `hosts` can resolve any IPv6 addresses that might be in any of the `ipnodes` repositories.
- Destination address selection has been modified to avoid using an IPv6 address for a remote host if no IPv6 routes serve that host. Instead, an IPv4 address is used, to avoid any delays when connecting to remote hosts.

For example, consider the case of an IPv6-enabled host on a network with no IPv6 router. Without the presence of a router, a host has no knowledge of IPv6 routes beyond the local link. Previously, the host experienced timeouts when trying to connect to a preferred IPv6 address. With the new feature for destination address selection, the host prefers an IPv4 destination address. This feature eliminates timeout problems.

IPv6 Temporary Addresses

The feature is new in the Solaris Express 11/03 release.

By default, the interface ID of an IPv6 address is autoconfigured with the interface's hardware-specific address. For example, if the interface is an Ethernet card, its interface ID is autoconfigured from the interface's MAC address. However, some system administrators might need to keep private the hardware-specific addresses of one or more interfaces on a node.

The IPv6 Temporary Addresses feature implements the privacy extensions standard that is defined in RFC 3041, "Privacy Extensions for Stateless Autoconfiguration in IPv6." The temporary address feature enables administrators to assign randomly generated, modified EUI-64 format interface IDs to interfaces of an IPv6 node. In addition, the administrators can specify a time limit for the lifetime of a temporary address. After configuration, the IPv6 daemon `in.ndpd` automatically generates the temporary interface ID for the interface, in addition to the automatically generated, MAC address-based interface ID.

SPARC: Sun ONE Application Server Integration

In the Solaris Express 9/03 release, the Sun ONE Application Server is now available for x86 platforms, in addition to SPARC platforms.

Sun ONE Application Server 7, Platform Edition (formerly iPlanet Application Server) is integrated in the Solaris Operating System. The Platform Edition of the Application Server provides the foundation for enterprise-class application services and web services. The server provides a high-performance, small-footprint Java™ 2 Platform, Enterprise Edition (J2EE™). J2EE enables the development, deployment, and management of enterprise applications and web services to a broad range of servers, clients, and devices.

The Sun ONE Application Server provides application portability and fast time to market for new Java and Extensible Markup Language (XML) applications. These new applications are J2EE 1.3 platform compatible. The Application Server enables developers to build applications that are based on JavaServer Pages™ (JSP™), Java Servlet, and Enterprise JavaBeans™ (EJB™) technology. This technology supports a broad range of business requirements from small departmental applications to enterprise-scale, mission-critical services.

Feature highlights include the following:

- Provides integration of the Sun ONE Message Queue and the Hypertext Transfer Protocol (HTTP) server from the Sun ONE Web Server
- Provides scalability with enhanced performance
- Provides comprehensive web services support – Java Web Services, Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL)
- Provides interoperability with the Sun ONE Portal Server 6.0 and Sun ONE Directory Server
- Uses the J2EE Reference Implementation

The following name changes have been made:

- Sun ONE Message Queue (formerly iPlanet Message Queue for Java)
- Sun ONE Web Server (formerly iPlanet Web Server)
- Sun ONE Portal Server (formerly iPlanet Portal Server)
- Sun ONE Directory Server (formerly iPlanet Directory Server)

For further information, see

http://wwws.sun.com/software/products/appsrvr/home_appsrvr.html.

For specific licensing terms, refer to the binary code license.

sendmail version 8.12 Uses TCP Wrappers

This feature is new in the Solaris Express 9/03 release.

TCP wrappers provide a way of implementing access controls by checking the address of a host that is requesting a particular network service against an access control list. Requests are granted or denied, accordingly. Besides providing this access control mechanism, TCP wrappers also log host requests for network services, which is a useful monitoring function. Examples of network services that might be placed under access control include `rlogind`, `telnetd`, and `ftpd`.

In this Solaris release, version 8.12 of `sendmail` now enables the use of TCP wrappers. This check does not bypass other security measures. By enabling TCP wrappers in `sendmail`, a check has been added to validate the source of a network request before the request is granted.

routeadm Command

This feature is new in the Solaris Express 9/03 release.

The new `routeadm` command enables system administrators to configure IP forwarding and routing on all interfaces of a system. Any settings that are established through `routeadm` override system defaults that are read from configuration files at boot-time.

The `routeadm` command provides options for enabling or disabling the global packet-forwarding function on all IPv4 or IPv6 interfaces of a system. `routeadm` can also be used to set up a system as a router, by turning on routing daemons for all system interfaces. In an IPv6 environment, `routeadm` can also be used to enable or disable dynamic routing for a host.

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

SPARC: TCP Multidata Transmit

This feature was introduced in the Solaris Express pilot program.

Multidata transmit (MDT) enables the network stack to send more than one packet at one time to the network device driver during transmission. Use of this feature reduces the per-packet processing costs by improving the host CPU utilization or network throughput.

The multidata transmit (MDT) feature is only effective for device drivers that support this feature.

MDT is enabled by default. MDT can be disabled by including the following line in the `/etc/system` file:

```
# ndd -set /dev/ip ip_multidata_outbound 0
```

For further information, see the `ip(7P)` and the `ndd(1M)` man pages.

ifconfig router Option

This feature was introduced in the Solaris Express pilot program.

The new router option of `ifconfig` allows you to configure IP packet forwarding on individual interfaces. `ifconfig router` and `ifconfig -router` enable or disable IP packet forwarding, respectively, for both IPv4 and IPv6 interfaces. The router option sets the `IFF_ROUTER` interface flag.

These new options replace the `ndd` variables `interface-name:ip_forwarding` and `interface-name:ip6_forwarding` for configuring IP packet forwarding on individual interfaces. Though now obsolete, the `ndd` variables remain in this Solaris release for backward compatibility. You can still use `ip_forwarding` and `ip6_forwarding` without the `interface-name` prefix to configure IP forwarding for all interfaces on a system.

For detailed information, refer to the `ifconfig(1M)`, `ip(7P)`, and `ip6(7P)` man pages.

Internet Protocol Version 6 (IPv6) Default Address Selection

This feature was introduced in the Solaris Express pilot program.

The Solaris Operating System now provides a documented and deterministic algorithm for doing IPv6 default source and destination address selection. This feature gives system administrators the limited ability to change address selection precedence through use of a policy table.

The IPv6 default address selection feature is a standardized method for IPv6 source and destination address selection.

The selection mechanisms can be configured by using a policy table. For example, you can edit the policy table to give higher precedence to a particular address prefix. Thereafter, addresses that are within the prefix are sorted before other addresses by name look-up APIs. You can also assign labels to source and destination prefixes in the policy table. This assignment makes sure that particular source addresses are only used with particular destination addresses.

To implement IPv6 default address selection, the Solaris Operating System now includes the `/etc/inet/ipaddrsel.conf` file and the `/usr/sbin/ipaddrsel` command. You use `ipaddrsel.conf` to edit the IPv6 default address policy table. Then you use `ipaddrsel` to commit the changes to the policy table.

Additionally, the `ifconfig` command now includes the “preferred” option. This option lets you designate a particular address to be used as the source address for all IPv6 communication.

For detailed information, refer to the `ipaddrsel.conf(4)`, `ipaddrsel(1M)` and the `ifconfig(1M)` man pages.

Sun ONE Message Queue

In the Solaris Express pilot program, the Sun ONE Message Queue is now available for SPARC platforms and x86 platforms.

The Solaris Operating System now supports Java Messaging Service (JMS) applications. This Solaris release uses Sun ONE Message Queue (formerly iPlanet Message Queue for Java) as a JMS provider.

JMS messaging enables applications and application components to exchange messages asynchronously and reliably. Processes that run on different platforms and different operating systems can connect to a common message service to exchange information.

Sun ONE Message Queue, Platform Edition provides a full implementation of the JMS specification. The message queue provides additional features such as the following:

- Centralized administration
- Tunable performance
- Support for the Java Naming and Directory Interface™ (JNDI)
- Support for SOAP messaging

For further information, see the following Web site:

http://www.sun.com/software/products/message_queue/home_message_queue.html

Disable NFS and automount Services

This feature was introduced in the Solaris Express pilot program.

The following NFS daemons are typically started by default at boot time by the `rc` scripts: `nfsd`, `mountd`, `statd`, `lockd`, and the automount daemon, `automountd`. Now, if a machine does not require NFS and automount services, the scripts do not start the NFS daemons and the automount daemon.

The following describes the new behavior:

- The automount utility, which is called by `/etc/init.d/autofs`, now starts `automountd` at boot time only if the automount maps have a valid entry.
- `/etc/init.d/nfs.server` starts `mountd`, `nfsd`, `statd`, and `lockd` at boot time only if the machine has any NFS exports.
- `/etc/init.d/nfs.client` starts `statd` and `lockd` only if `/etc/vfstab` includes NFS file systems.

NFS and automount services might not be started at boot time. The following commands can start these services when required by a machine:

- The `automount` utility can start the `automountd` daemon.
- The `mount` command with the `-nfs` option can start the `lockd` and `statd` daemons. The `automountd` daemon also can start `lockd` and `statd`.
- The `share` command with the `-nfs` option can start the `nfsd`, `mountd`, `lockd`, and `statd` daemons.

This mechanism for disabling NFS and automount services provides the following benefits:

- Extra security that comes from not running unnecessary daemons on a machine.
- A simplified process for exporting file systems. The `nfsd` and `mountd` daemons, and, if necessary, `lockd` and `statd`, are started by the `share` command with the `-nfs` option. So, you no longer have to edit the `/etc/dfs/dfstab` file and then invoke the `/etc/init.d/nfs.server` program. This new behavior permits an NFS export to be configured with a single command, without editing any configuration files. However, if the system reboots, such exports are not resumed automatically, unless the exports are included in the `/etc/dfs/dfstab` file.

For more information, refer to the man pages for `mountd(1M)`, `lockd(1M)`, `statd(1M)`, and `nfsd(1M)`.

Internet Protocol Version 6 (IPv6) 6to4 Router

This feature was introduced in the Solaris Express pilot program.

IPv6 networks can now transfer packets over Internet Protocol Version 4 (IPv4) networks by configuring one or more routers to support a 6to4 tunnel. System administrators can use 6to4 tunnels as a transitional method for migrating their networks from IPv4 to IPv6. This feature implements RFCs 3056 and 3068.

Packet Tunneling Over IPv6

This feature was introduced in the Solaris Express pilot program.

This feature enables tunneling over IPv6 for both IPv4 over IPv6 tunnels and IPv6 over IPv6 tunnels. IPv4 packets or IPv6 packets can be encapsulated in IPv6 packets.

Hosting Multiple Web Sites on a Single Solaris Machine

This feature was introduced in the Solaris Express pilot program.

The Solaris Network Cache and Accelerator (NCA) kernel module now supports multiple instances of a web server. This support enables you to use a Solaris machine to perform Internet protocol (IP) address-based virtual web hosting. The Solaris software uses a single configuration file, `/etc/nca/ncaport.conf`, to map NCA sockets to IP addresses.

For further information, see the `ncaport.conf(4)` man page.

IP Quality of Service

This feature was introduced in the Solaris Express pilot program.

IP Quality of Service (IPQoS) is a new feature in the Solaris Operating System. IPQoS enables system administrators to provide different levels of network service to customers and to critical applications. By using IPQoS, the administrator can set up service-level agreements. These agreements provide an Internet service provider's (ISP) clients with varying levels of service that are based on a price structure. A company could also use IPQoS to prioritize among applications so that critical applications get a higher quality of service than less critical applications.

User Selector for Internet Protocol Quality of Service (IPQoS)

This feature was introduced in the Solaris Express pilot program.

The Solaris IPQoS feature now includes the user selector, which supplements the existing `uid` selector. The user selector enables you to specify a user name or `userID` as criteria in a filter clause in the `ipqosconf` file. Previously, the `uid` selector only accepted a `userID` as a value. The following filter clause from an `ipqosconf` file shows the user selector:

```
filter {  
    name myhost;  
    user root;  
}
```

For information about filters and selectors, refer to the *IPQoS Administration Guide* and the `ipqosconf(1M)` man page.

Routing Information Protocol Version 2 (RIPv2)

This feature was introduced in the Solaris Express pilot program.

Solaris system software now supports Routing Information Protocol Version 2 (RIPv2).

RIPv2 adds Classless Inter-Domain Routing (CIDR) and Variable-Length Subnet Mask (VLSM) extensions to the RIPv1 protocol. Message Digest 5 (MD5) extensions protect routers against intentional misdirection by malicious users. The new `in.routed` implementation also includes a built-in Internet Control Message Protocol (ICMP) Router Discovery (RFC 1256) mechanism.

RIPv2 supports multicast, if the point-to-point links are enabled with multicast. RIPv2 also supports unicast. If you configure a broadcast address by using the `/etc/gateways` file, then RIPv2 supports broadcast.

For information on how to configure RIPv2, see the `in.rdisc(1M)`, `in.routed(1M)`, and `gateways(4)` man pages.

Development Enhancements

This section describes all development tools that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases. DTrace dynamic tracing facility is an important new feature in the Solaris Express 11/03 release.

Dynamic Tracing Facility

The Solaris DTrace is a new feature of importance for developers in the Solaris Express 11/03 release. DTrace is a comprehensive dynamic tracing facility that gives Solaris users, administrators, and developers a new level of observability into the kernel and user processes. For further information, see “DTrace Dynamic Tracing Facility” on page 15.

Atomic Operations

This feature is new in the Solaris Express 10/03 release.

Atomic operations provide APIs in `libc` that perform simple atomic operations quickly. This new feature permits applications to atomically update memory without using other synchronization primitives or platform-specific assembly language. The available operations include addition, the “and” Boolean function, and the “or” Boolean function.

For more information, see the `atomic_ops(3C)` man page.

Solaris WBEM File Changes

This feature description has been updated in the Solaris Express 9/03 release.

There have been several changes to the Managed Object Format (MOF) files in the `/usr/sadm/mof` directory.

- The `Solaris_VM1.0.mof` file was revised to `Solaris_VM2.0.mof` and subsequently to `Solaris_VM3.0.mof`.
- The local file system classes were relocated from `Solaris_VM2.0.mof` to the new file, `Solaris_FS1.0.mof`. `Solaris_FS1.0.mof` defines classes that pertain to storage devices.
- Two of the providers, `Solaris_DiskDrive` and `Solaris_DiskPartition`, that were in the `Solaris_VM1.0.mof` file, were moved into the new `Solaris_DMGT.1.0.mof` file. The `Solaris_DMGT.1.0.mof` file contains classes that represent disks, disk partitions, and other device management classes.
- This release includes another new MOF file, `Solaris_NFS1.0.mof`. The `Solaris_NFS1.0.mof` file defines classes that pertain to NFS devices. This file contains the NFS classes from `Solaris_VM2.0.mof` as well as new classes for configuring and monitoring NFS shares (or “exports”) and mounts.

Solaris Cryptographic Framework for Developers

This feature was introduced in the Solaris Express pilot program.

The Solaris Cryptographic Framework provides cryptographic services to applications. Applications can access the framework through `libpkcs11(3lib)` and at higher levels.

The Solaris Cryptographic Framework provides the following features for developers of applications that use encryption:

- User-level programming interfaces for various cryptographic functions. These interfaces cover, for example, encryption, decryption, message digests, and signing. The industry standard, RSA Security Inc. PKCS #11 Cryptographic Token Interface (Cryptoki), serves as the API. The framework supports the following encryption algorithms:
 - AES
 - DES/3DES
 - RC4
 - MD5
 - SHA-1
 - DSA
 - RSA
 - D-H
- User level pluggable interfaces for Sun and third-party developers. These interfaces enable administrators to add new plugins from providers of encryption algorithms at the user level. Administrators can replace an existing provider with a different implementation. The user service provider interface (SPI) also uses the PKCS#11 standard. Tools for signing, packaging, and installing third-party binaries are provided.
- An optimized software implementation of the most commonly used encryption and digital signing algorithms, such as AES, DES/3DES, and RSA. These implementations have been optimized for the SPARC platform and UltraSPARC platform.
- An administrative CLI tool, `cryptoadm`, for adding or removing encryption plugins, setting cryptographic security policy, and other related administrative functions. See the `cryptoadm(1M)` man page.

See the following man pages: `libpkcs11(3lib)`, `pkcs11_softtoken(5)` and `pkcs11_kernel(5)`. See also “Solaris Cryptographic Framework for System Administrators” on page 22.

SPARC: 64-bit Package Changes for Software Developers

This feature was introduced in the Solaris Express pilot program.

In this Solaris Operating System, packaging has been simplified with most 32-bit components and 64-bit components being delivered in a single package. See “SPARC: 64-bit Package Changes” on page 48.

SPNEGO Pseudo-Mechanism for GSS-API Applications

This feature was introduced in the Solaris Express pilot program.

The Solaris Express releases include a new GSS-API “pseudo-mechanism” for negotiating GSS-API security that is based on the SPNEGO protocol (IETF RFC 2478). Simple and Protected GSS-API Negotiation (SPNEGO) is most useful for applications that are based on GSS-API implementations which support multiple security mechanisms. SPNEGO can be applied when two applications use GSS-API to exchange data and do not know which mechanisms are supported by the other application.

SPNEGO is a pseudo-security mechanism that is represented by the following object identifier:

```
iso.org.dod.internet.security.mechanism.snego (1.3.6.1.5.5.2)
```

SPNEGO enables GSS-API peers to determine in-band whether their credentials share common GSS-API security mechanisms. If the mechanisms are shared, then the peers can select a common mechanism to establish the security context.

For further information, see the `mech(4)` and the `mech_spnego(5)` man pages.

SPARC: Locality Groups

This feature was introduced in the Solaris Express pilot program.

The *Programming Interfaces Guide* now includes a chapter that explains the interfaces that interact with locality groups (lgroups). These interfaces can be used to help an application efficiently allocate CPU and memory resources. This capability results in improved performance on some systems.

Solaris Privileges for Software Developers

This feature was introduced in the Solaris Express pilot program.

Processes need no longer run as `root` to have superuser rights. Instead, superuser rights can be distributed by system administrators as discrete process privileges. Through process privileges, developers can limit access to restricted operations. And developers can limit the periods for which privileges are effective. The use of privileges can reduce the harm that formerly resulted if a privileged program was compromised. For compatibility, unmodified programs that run as `root` continue to have all privileges.

For general information about privileges, see “Solaris Privileges” on page 26. For information about setting and getting privileges, see the `setppriv(2)` and `getppriv(2)` man pages. To learn more about manipulating privileges, see the `priv_str_to_set(3C)` and the `priv_addset(3C)` man pages.

Thread Stacks in pmap

This feature was introduced in the Solaris Express pilot program.

The pmap utility now labels threads' stacks so the stacks can be identified more easily.

See the pmap(1) man page for further information.

New DOOR_REFUSE_DESC Flag

This feature was introduced in the Solaris Express pilot program.

A new flag, DOOR_REFUSE_DESC, was added to the door_create() function. This new flag simplifies writing door servers which do not accept argument descriptors.

For further information, see the door_create(3door) man page.

Stack Check APIs

This feature was introduced in the Solaris Express pilot program.

Stack Check APIs allow for advanced interaction with stack-checking compiler support available in Forte™ 7.0. These APIs should be used in applications that are compiled with stack checking enabled, and either manage their own stacks or attempt to detect their own stack overflows.

Developers who maintain their own thread library need to use the setustack interface to enable consumers of their library to compile with stack checking enabled.

See the stack_getbounds(3C), stack_setbounds(3C), and the stack_inbounds(3C) man pages.

Enhanced crypt() Function for Software Developers

This feature was introduced in the Solaris Express pilot program.

The Solaris Express releases include new extensions to the crypt() function and introduces the crypt_gensalt() function. These enhancements allow administrators to change the algorithm that is used to obscure users' UNIX login passwords.

Modules are included for MD5 and Blowfish. The MD5 modules are at crypt_sunmd5 and crypt_bsdmd5. The Blowfish module is at crypt_bsdbf.

Developers can create new modules for alternate password-obscuring algorithms. Application developers must use the crypt_gensalt() function instead of manually generating the salt string for passing to the crypt() function.

Modules for alternate algorithms are specified in the crypt.conf(4) file. The module_path field specifies the path to the shared library object that implements the two required functions:

- crypt_gensalt_impl() – Generates the salt string

- `crypt_genhash_impl()` – Generates the encrypted password

For further information, see the `crypt(3C)` and the `policy.conf(4)` man pages.

New Flags for `madvise()` Function

This feature was introduced in the Solaris Express pilot program.

The `madvise()` function enables the kernel to optimize access to a user-defined region of memory. This Solaris release includes three new flags for the `madvise()` function:

- `MADV_ACCESS_LWP` – Gives a specified lightweight process (LWP) resource allocation priority
- `MADV_ACCESS_MANY` – Specifies an address range that is intensively used by processes across the machine
- `MADV_ACCESS_DEFAULT` – Resets an address range's access pattern to the system default

For further information on the `madvise()` function, see the `madvise(3C)` man page.

Linkers and Libraries Updates

This feature was introduced in the Solaris Express pilot program.

The Solaris Express releases include new linker-editor features such as string table compression, unreferenced section elimination, and unreferenced dependency detection:

- Thread-local storage is provided.
- The `-z ignore` option has been extended to eliminate unreferenced sections during a link-edit. See the `ld(1)` man page.
- Greater flexibility in defining a symbols' visibility is provided with the "protected" mapfile directive.
- `dlopen(3DL)` and `dlsym(3DL)` lookup semantics have been expanded with a new mode, `RTLD_FIRST`.
- Unreferenced dependencies can be determined by using the `ldd` utility. See the `-U` option in the `ldd(1)` man page.

Memory Allocation With `libumem`

This feature was introduced in the Solaris Express pilot program.

`libumem` is a user-mode (nonkernel mode) memory allocator library. `libumem` has features that enable you to debug memory leaks and other aberrations that involve memory usage.

This feature is used in the same way that a standard application binary interface (ABI) allocator, such as `malloc()`, is used. A user-mode application requests an arbitrary number of bytes of memory. Then a pointer is returned that is loaded with the address of the allocated memory.

For further information, see the `libumem(3LIB)` man page.

Smart Card Terminal Interfaces

This feature was introduced in the Solaris Express pilot program.

Solaris smart card interfaces are a set of public interfaces for Smart Card Terminals. Card terminal vendors can implement these interfaces in a user-level shared library to provide device-level support for their smart card terminals in Solaris. The set of Solaris smart card terminal interfaces is based on card terminal interfaces that are available as part of the Linux Smartcard framework. Card terminal support libraries from Linux can be ported to Solaris with minimum effort.

Smartcard Middleware APIs

This feature was introduced in the Solaris Express pilot program.

The Solaris Smartcard framework now provides low-level middleware APIs. These APIs can be used to exchange data with a smart card by using a smart card reader. The APIs can be used in platforms such as the Sun Blade™ and Sun Ray™ systems. Applications that are written in the Java language, or in C, can use these interfaces.

For more information, see the `libsmartcard(3LIB)` man page and the JavaDocs in `/usr/share/javadoc/smartcard`.

Language Support Enhancements

This section describes all language support features that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases. The locale administrator is a new feature in the Solaris Express 11/03 release.

Locale Administrator

This feature is new in the Solaris Express 11/03 release.

Locale administrator allows the user to query and configure the locales for a Solaris OS through a command-line interface. Using the `localeadm` tool, a user can display information about locale packages that are installed on the system or that reside on a particular device or directory. The user can add and remove locales on the current system on a per-region basis. For example, the user can add all locales in the Eastern European region to the current system. See the `localeadm(1M)` man page.

Prior to this feature being introduced, once a system was installed the user had to add or remove individual packages to change the locales on the machine. This process is prone to error since it is easy to miss or overlook packages. This tool is a supplement to the locale selection logic in the solaris installer. The installer is still the primary application for the correct installation of solaris locales.

Code Conversions for Internationalized Domain Name Support

This feature is new in the Solaris Express 10/03 release.

Internationalized Domain Name (IDN) enables the use of non-English native language names as host and domain names. To use such non-English host and domain names, application developers must convert such names into ASCII Compatible Encoding (ACE) names in their applications as specified in the RFC 3490. System administrators are also required to use ACE names in system files and applications where the system administration applications do not yet support the IDNs.

This feature aids in the conversion by providing the conversion API with various supported option arguments. Refer to the following man pages for more detail:

- `libidnkit(3LIB)`
- `idn_decodename(3EXT)`
- `idn_decodename2(3EXT)`
- `idn_encodename(3EXT)`
- `iconv_en_US.UTF-8(5)`

Standard Type Services Framework

This feature is new in the Solaris Express 9/03 release.

The Standard Type Services Framework (STSF) is a pluggable object-based architecture that allows users to access typographically sophisticated text layout and rendering. The pluggable architecture of the framework gives users the ability to use different font rasterization engines and text layout processors to achieve the desired visual representation. The pluggable architecture also manages fonts and enables application-specific fonts to be created. STSF includes both a standalone API and an X server extension to handle rendering on the server side for improved efficiency. STSF is an open source project that is sponsored by Sun Microsystems.

For more information about the project and how to use the API, see <http://stsf.sourceforge.net>.

Auto Encoding Finder

This feature is new in the Solaris Express 9/03 release.

The Auto Encoding Finder is a useful utility for global character handling. Through a general-purpose interface, Auto Encoding Finder provides an easy way to detect the encoding of particular file or string, simplifying access to various language character encodings. For example, the utility simplifies the display of web pages that do not

specify encoding information. Search engines, knowledge databases, and machine translation tools might also need to detect the encoding of the language data being accessed. The Auto Encoding Finder tool simplifies this process.

See the `auto_ef(1)` or `libauto_ef(3LIB)` man pages for more detail.

SunIM-based Korean Language Engine With Auxiliary Window Support

This feature was introduced in the Solaris Express pilot program.

Korean users of the Solaris Operating System can benefit from more comprehensive keyboard input method support for the Korean language. The new Korean Language Engine with auxiliary window support offers Korean users four auxiliary windows to control and configure the Korean input method (IM).

- User-based preferences can be set within one window.
- A virtual keyboard environment can be set within another window for point-and-click selection of Korean characters.
- Within another window, users can choose the symbols that they need from special characters that are based on code points.
- Users can organize all of the windows within a special palette of control.

This IM supports three separate keyboard layouts: 2 beol sik, 3 beol sik 390, and 3 beol sik final.

Common Transliteration-based Input Method for All Indian languages

This feature was introduced in the Solaris Express pilot program.

Users who operate within any Unicode (UTF-8) locale of the Solaris Operating System can now easily and intuitively input characters from Indian regional languages. Users who interact with CDE applications, StarOffice™, or Mozilla can more easily interact with Indian scripts. After selecting the transliteration-based input method (IM), users can type phonetic equivalents of Indian language scripts in English. These equivalents are then displayed in the script that is selected, and are correctly shaped and rendered with the help of an underlying layout and shaper module. As transliteration is the most commonly used input method for Indian languages, this support can greatly enhance the usability of the eight Indian scripts that are provided in the Solaris software.

Unicode Version 3.2 Support

This feature was introduced in the Solaris Express pilot program.

Solaris Unicode locales now support the Unicode Version 3.2. The Unicode Version 3.2 introduces 1016 new characters. This version also includes both normative changes and informative changes, as described in the *Unicode Standard Annex #28: UNICODE 3.2* at <http://www.unicode.org/unicode/reports/tr28/>.

The UTF-8 character representation has been also changed to a more secure form, as specified in the Unicode Version 3.2. This feature implements the more secure UTF-8 character representation, form, and byte sequences in `iconv` code conversions. This feature also implements all OS-level multibyte and wide-character functions, such as the following:

- `mbtowc(3C)`
- `wctomb(3C)`
- `mbstowcs(3C)`
- `wcstombs(3C)`
- `mbrtowc(3C)`
- `wcrtomb(3C)`
- `mbsrtowcs(3C)`
- `fgetwc(3C)`
- `mblen(3C)`

Additional Keyboard Support

This feature was introduced in the Solaris Express pilot program.

Software support for three additional keyboards has been added to the Solaris software: Estonian keyboard Type 6, French-Canadian keyboard Type 6, and Polish programmer's keyboard Type 5. This software support gives users in Canada, Estonia, and Poland greater flexibility for keyboard input. The support modifies standard U.S. keyboard layouts to different language needs.

Wubi Input Method

This feature was introduced in the Solaris Express pilot program.

The Wubi input method (IM) is widely used in China. The encoding rule for Wubi IM is based on the radical or stroke shape of Chinese characters. Users can rapidly type Chinese characters through a standard keyboard rather than through slower, phonetic-based input methods.

Input Method Support for Indic

This feature was introduced in the Solaris Express pilot program.

Input support for Indian regional language keyboards has been added to the Solaris Operating System. Indic language users can type Indic language characters by using their preferred keyboard layouts in the Solaris Operating System.

Seven Additional Indic Scripts for Support in Unicode Locales

This feature was introduced in the Solaris Express pilot program.

In addition to the current support for Hindi, the following Indic scripts are now supported in the Solaris software:

- Bengali

- Gurmukhi
- Gujarati
- Tamil
- Malayalam
- Telugu
- Kannada

Speakers of these Indian regional languages have language support in the Solaris Operating System for any of the Unicode locale environments that Solaris supports.

HKSCS-2001 Support in Hong Kong Locales

This feature was introduced in the Solaris Express pilot program.

HKSCS-2001 is a new version of the Hong Kong Supplementary Character Set (HKSCS). This new version adds 116 characters to the previous HKSC-1999 character set. HKSCS-2001 is supported in these Solaris Hong Kong locales: zh_HK.BIG5HK and zh_HK.UTF-8.

Solaris Documentation Changes

This section describes all Solaris documentation changes that have been made in the Solaris Express 11/03 release and in all prior Solaris Express releases. The most recent documentation changes include the new *Solaris Dynamic Tracing Guide* and revisions to the *Application Packaging Developer's Guide*.

Solaris Dynamic Tracing Guide

The *Solaris Dynamic Tracing Guide*, new in the Solaris Express 11/03 release, will help you learn DTrace. The guide includes a complete feature reference and examples to help you get started. See “DTrace Dynamic Tracing Facility” on page 15 for further information on DTrace.

Application Packaging Developer's Guide

The *Application Packaging Developer's Guide* has been revised in the Solaris Express pilot program. See “Signed Packages and Patches” on page 17. The manual was further revised in the Solaris Express 11/03 release.

Programming Interfaces Guide

The *Programming Interfaces Guide* has been revised in the Solaris Express 10/03 release. The material in the “Sockets” chapter has been updated to reflect new IPv6 functionality.

This guide was also revised during the prior Solaris Express pilot program. See the description for “SPARC: Locality Groups” on page 38. During the pilot program, the “Solaris ABI and ABI Tools” chapter in the *Programming Interfaces Guide* was revised for clarity. No functionality changes were made in this chapter.

System Administration Guide: Basic Administration

The *System Administration Guide: Basic Administration* has been revised in the Solaris Express 10/03 release. Revised information concerning USB devices and USB 2.0 features has been provided in this guide.

The *System Administration Guide: Basic Administration* was also revised in the prior Solaris Express 9/03 release. See the description for “Solaris Support for USB Devices” on page 55. See also the descriptions for “USB 2.0 Features” on page 54 and “USB Mass Storage Devices” on page 55.

The *System Administration Guide: Basic Administration* also covers features that were revised in the Solaris Express pilot program. See the following descriptions:

- “UFS Logging Performance Enhancements” on page 47
- “Signed Packages and Patches” on page 17
- “Command-Line Interface Enhancements to the Solaris Product Registry” on page 52
- “USB Mass Storage Devices” on page 55
- “SPARC: Multiterabyte UFS File Systems” on page 59
- “SPARC: Multiterabyte Volume Support With EFI Disk Labels” on page 61

Solaris Tunable Parameters Reference Manual

This feature description was part of the Solaris Express pilot program.

The *Solaris Tunable Parameters Reference Manual* has been revised to include information about NFSv4 parameters. For further information, see “NFS version 4” on page 58.

System Administration Guide: Network Services

This feature description was part of the Solaris Express pilot program.

The *System Administration Guide: Network Services* has been revised. For further information, see “NFS version 4” on page 58 and “Enhancements to the NFS Client” on page 59.

Solaris 10 Installation Guide

This feature description was part of the Solaris Express pilot program.

The *Solaris 10 Installation Guide* has been revised for the Solaris Express releases. For further information, see the following descriptions.

- “WAN Boot Installation Method” on page 49
- “Creating RAID-1 Volumes (Mirrors) With Custom JumpStart” on page 49
- “Solaris Live Upgrade 2.1” on page 50
- “Solaris Flash Archives” on page 51
- “Support for Lightweight Directory Access Protocol (LDAP) Version 2 Profiles” on page 53

Solaris WBEM Developer's Guide

This feature description was part of the Solaris Express pilot program.

The *Solaris WBEM Developer's Guide* has been revised. See “Solaris WBEM File Changes” on page 36.

System Performance Enhancements

This section describes all system performance features that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases. The “New Architecture for Network Stacks” description is of particular importance.

New Architecture for Network Stacks

This new architecture is of key importance in Solaris Express. This feature is introduced in the Solaris Express 10/03 release.

The networking stack for TCP connections has been rearchitected to deliver extremely high performance while improving the scalability. This innovation enables Solaris networking to vertically partition the workload by using a IP classifier-based lockless design. This design reduces the overheads of synchronization and reduces cross-communication between CPUs. The improvements are visible across all networking workloads and are transparent to the users.

Improved System Performance

This feature was introduced in the Solaris Express pilot program.

System performance during and after plumbing large numbers of interfaces, such as virtual area local networks (VLANs) and tunnels, has been improved significantly. However, avoid using large numbers (thousands) of routes with large numbers (thousands) of interfaces.

UFS Logging Performance Enhancements

This feature was introduced in the Solaris Express pilot program.

UFS logging enables faster system reboots. Because the file system transactions are already stored, checking the file system is not required if the file system is already consistent.

In addition, the performance of UFS logging improves or exceeds the level of performance of nonlogging file systems in this Solaris release. Results of the Standard Performance Evaluation Corporation system file server (SPECsfs) benchmark show that the performance of NFS-mounted file systems that have logging enabled equals the level of non-UFS logging performance. In some I/O-bound configurations, logging UFS file systems exceed the performance of nonlogging UFS file systems by approximately 25 percent. In other tests, performance of logging UFS file systems is 12 times better than the performance of nonlogging UFS file systems.

For information on how to enable logging on a UFS file system, see the *System Administration Guide: Basic Administration* or the `mount_ufs(1M)` man page.

SPARC: Memory Placement Optimization (MPO)

This feature was introduced in the Solaris Express pilot program.

The Solaris dispatcher and the virtual memory subsystem are enhanced to optimize the time that is required for an application to access memory. This enhancement can automatically improve the performance of many applications. This feature is currently implemented to support specific platforms such as the Sun Fire™ 3800–6800, the Sun Fire 12K, and the Sun Fire 15K.

A technical white paper about *Solaris Memory Placement Optimization and Sun Fire Servers* is available at the following Web site:

http://www.sun.com/servers/wp/docs/mpo_v7_CUSTOMER.pdf

SPARC: Dynamic Intimate Shared Memory (DISM) Large-Page Support

This feature was introduced in the Solaris Express pilot program.

Large-page support is now provided for Dynamic Intimate Shared Memory (DISM). This large-page support can increase the performance of applications that are capable of dynamically adjusting shared memory sizes.

Installation Enhancements

This section describes all installation enhancements that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases.

Replacement of Solaris Web Start Command-Line Interface

This feature is new in the Solaris Express 9/03 release.

The command-line interface of the Solaris Web Start OS installation program has been replaced to improve the usability of console-based installations. All console-based installations use the `suninstall` forms-based program to install the Solaris Operating System.

See the `suninstall(1M)` man page for further information.

SPARC: 64-bit Package Changes

This feature was introduced in the Solaris Express pilot program.

In previous Solaris releases, the Solaris software was delivered in separate packages for 32-bit components and 64-bit components. Now, packaging has been simplified with most 32-bit components and 64-bit components being delivered in a single package. The combined packages retain the names of the original 32-bit packages, and the 64-bit packages are no longer delivered.

The removal of the 64-bit packages simplifies installation and increases performance:

- Reduces the number of packages, which simplifies Custom JumpStart™ scripts that contain lists of packages
- Simplifies the packaging system with only one package that groups software functions
- Reduces installation time because fewer packages are installed

The 64-bit packages are renamed with the following conventions:

- If a 64-bit package has a 32-bit counterpart, the 64-bit package is named with the 32-bit package name. For example, a 64-bit library such as `/usr/lib/sparcv9/libc.so.1` previously would have been delivered in `SUNWcs1x`, but now is delivered in `SUNWcs1`. The 64-bit `SUNWcs1x` package is no longer delivered.
- If a package does not have a 32-bit counterpart, the “x” suffix is removed from the name. For example, `SUNW1394x` becomes `SUNW1394`.

This change means that you might need to modify your custom JumpStart script or other package installation scripts to remove references to the 64-bit packages.

WAN Boot Installation Method

This feature was introduced in the Solaris Express pilot program.

The Solaris software now enables you to boot and install software over a wide area network (WAN) by using HTTP. The WAN boot installation method enables you to install the Solaris release on systems over a large public network where the network infrastructure might be untrustworthy. You can use WAN boot with new security features to protect data confidentiality and installation image integrity.

The WAN boot installation method enables you to transmit an encrypted Solaris Flash archive over a public network to a remote client. The WAN boot programs then install the client system by performing a custom JumpStart installation. To protect the integrity of the installation, you can use private keys to authenticate and encrypt data. You can also transmit your installation data and files over a secure HTTP connection by configuring your systems to use digital certificates.

For more information about this feature, see the *Solaris 10 Installation Guide*.

Creating RAID-1 Volumes (Mirrors) With Custom JumpStart

This feature was introduced in the Solaris Express pilot program.

The custom JumpStart installation method now enables you to create RAID-1 volumes (mirrors) during the installation of the Solaris Operating System. By mirroring file systems, you can protect your system by duplicating data over two physical disks. If one of the mirrored disks fails, the system data is still accessible from the second mirrored disk.

In JumpStart, the following new custom profile keywords and values enable you to create mirrored file systems.

- The new `filesys` keyword value `mirror` creates a mirror. Then you can designate specific slices as single-slice concatenations to attach to the mirror.
- The new `metadb` profile keyword enables you to create the required state database replicas.

For more information, see the *Solaris 10 Installation Guide*.

Solaris Live Upgrade 2.1

This feature was introduced in the Solaris Express pilot program.

Solaris Live Upgrade provides a method of upgrading a system while the system continues to operate. While your current boot environment is running, you can duplicate the boot environment, then upgrade the duplicate. Or, rather than upgrading, you can install a Solaris Flash archive on a boot environment. The original system configuration remains fully functional and unaffected by the upgrade or installation of an archive. When you are ready, you can activate the new boot environment by rebooting the system. If a failure occurs, you have a safety net. You can quickly revert to the original boot environment with a simple reboot. Thus, you eliminate the normal downtime of the test and evaluation process.

New features in Solaris Live Upgrade 2.1 provide the following new functionality.

- Solaris Live Upgrade uses Solaris Volume Manager technology to create a duplicate boot environment that contains file systems with RAID-1 volumes (mirrors). The mirror provides data redundancy for any file systems, including the root (/) file system. With the `lucreate` command, you can create mirrored file systems that contain up to three submirrors.
- With the `lucreate` command, you can now exclude some files and directories that would normally be copied from the original boot environment. If you have excluded a directory, you can also include specified files and subdirectories under that directory.

For further information, see the *Solaris 10 Installation Guide*.

Custom JumpStart Installation Method Creates New Boot Environment

This feature was introduced in the Solaris Express pilot program.

You can now use the JumpStart installation method to create an empty boot environment when you install the Solaris Operating System. The empty boot environment can then be populated with a Solaris Flash archive for later use.

For further information, see the *Solaris 10 Installation Guide*.

Solaris Flash Archives

The Solaris Flash features described in this section were introduced in the Solaris Express pilot program.

The Solaris Flash installation feature enables you to use a single reference installation of the Solaris software on a system. This system is called the master system. Then, you can replicate that installation on a number of systems, which are called clone systems. The installation is an initial installation that overwrites all files on the clone system.

Solaris Flash Differential Archives and Configuration Scripts

This feature was introduced in the Solaris Express pilot program.

The Solaris Flash installation feature provides new enhancements for the Solaris software.

- A Solaris Flash installation can now update a clone system with minor changes. If you have a clone system and want to update it, you can create a differential archive that contains only the differences between two images, the original master image and an updated master image. When you update a clone system with a differential archive, only the files that are specified in the differential archive are changed. The installation is restricted to clone systems that contain software which is consistent with the original master image. You use the custom JumpStart installation method to install a differential archive on a clone system. Or, you can use Solaris Live Upgrade to install a differential archive on a duplicate boot environment.
- Special scripts can now be run for configuration of the master or clone or can be run to validate the archive. These scripts enable you to do the following tasks.
 - Configure applications on clone systems. You can use a custom JumpStart script for some uncomplicated configurations. For more complicated configurations, special configuration file processing might be necessary on the master system or before or after installation on the clone system. Also, local preinstallation and postinstallation scripts can reside on the clone. These scripts can protect local customizations from being overwritten by the Solaris Flash software.
 - Identify nonclonable, host-dependent data that enables you to make the flash archive host independent. Host independence is enabled by modifying such data or excluding the data from the archive. An example of host-dependent data is a log file.
 - Validate software integrity in the archive during creation.
 - Validate the installation on the clone system.

For further information, see the *Solaris 10 Installation Guide*. This guide also includes information on how to use Solaris Live Upgrade to install a differential archive.

Customizing Content in a Solaris Flash Archive

This feature was introduced in the Solaris Express pilot program.

The `flarcreate` command is used to create a Solaris Flash archive. This command has been updated with new options that increase your flexibility to define archive contents when creating an archive. You now can exclude more than one file or directory. From an excluded directory, you can add back a subdirectory or file. This feature is useful when you want to exclude large data files that you do not want cloned.

For information on how to use these options, see the *Solaris 10 Installation Guide*.

Note the following name change: Solaris Flash (formerly Web Start Flash).

Command-Line Interface Enhancements to the Solaris Product Registry

This feature was introduced in the Solaris Express pilot program.

The `prodreg` command has been updated to include functionality that is similar to the Solaris Product Registry graphical user interface. You can now use the following `prodreg` subcommands on the command line or in administration scripts to perform a variety of tasks.

- `browse` – The `browse` subcommand enables you to view registered software in a terminal window. By repeating the `browse` subcommand, you can navigate through the directory hierarchy of registered software.
- `info` – The `info` subcommand enables you to view information about registered software. You can use the `info` subcommand to identify the following specifics.
 - Installation location of the software
 - Other software that the specified software requires
 - Other software that depends on the specified software
 - Software that has been damaged by the removal of packages that the software requires
- `unregister` – The `unregister` subcommand removes software installation information from the Solaris Product Registry. If you remove software from your system without properly uninstalling the software from the Registry, you can use the `prodreg unregister` command to clean up the obsolete entries in the Solaris Product Registry.
- `uninstall` – The `uninstall` subcommand enables you to remove registered software from your system by launching that software's uninstall program.

For more information, see the `prodreg(1M)` man page and the *System Administration Guide: Basic Administration*.

Reduced Networking Software Group

This feature was introduced in the Solaris Express pilot program.

You can now create a more secure system with fewer enabled network services by selecting or specifying the Reduced Networking software group (`SUNWCnet`) during your installation. The Reduced Networking software group provides system

administration utilities and a multiuser text-based console. SUNWCrnet enables the system to recognize network interfaces. During the installation, you can customize your system's configuration by adding software packages and by activating network services as needed.

Modifying Disk Partition Tables by Using a Virtual Table of Contents (VTOC)

This feature was introduced in the Solaris Express pilot program.

The Solaris Web Start program now enables you to load existing slices from the virtual table of contents (VTOC.) You can now preserve and use the system's existing disk slice tables during your installation, rather than using the installer's default disk layout.

x86: Change in Default Boot Disk Partition Layout

This feature was introduced in the Solaris Express pilot program.

A new feature in the Solaris Web Start program is a boot disk partition layout. This layout, by default, accommodates the Service partition on Sun LX50 systems. This installation program enables you to preserve an existing Service partition.

The new default includes the following partitions.

- First partition – Service partition (existing size on system)
- Second partition – x86 boot partition (approximately 11 Mbytes)
- Third partition – Solaris Operating System partition (remaining space on the boot disk)

If you want to use this default layout, select Default when the Solaris Web Start program asks you to choose a boot-disk layout.

Note – If you install the Solaris Express *x86 Platform Edition* release on a system that does not currently include a Service partition, the Solaris Web Start program does not create a new Service partition. If you want to create a Service partition on your system, first use the Sun LX50 Diagnostics CD to create the Service partition. Once the Service partition is created, then install the Solaris Express Operating System.

Support for Lightweight Directory Access Protocol (LDAP) Version 2 Profiles

This feature was introduced in the Solaris Express pilot program.

The Solaris installation programs now support LDAP Version 2 profiles. These profiles enable you to configure your system to use a proxy credential level. During the Solaris Web Start or `suninstall` programs, you can specify the LDAP proxy-bind distinguished name and proxy-bind password. With any installation method, you can preconfigure LDAP before installation by using the `proxy_dn` and `proxy_password` keywords in the `sysidcfg` file.

For information, see the *Solaris 10 Installation Guide*.

Device Management

This section describes all device management features that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases.

USB 2.0 Features

This description has been updated in the Solaris Express 9/03 release.

The Solaris Express releases include the following USB 2.0 features:

- **Better Performance** – Increased data throughput for devices attached to USB 2.0 controllers, up to 40 times faster than USB 1.1 devices.
You can particularly benefit from the high-speed USB protocol when accessing high-speed mass storage devices, such as DVDs and hard drives.
- **Backward Compatibility** – Compatibility with 1.0 and 1.1 devices and drivers so that you can use the same cables, connectors, and software interfaces.

For a description of USB devices and terminology, see “Overview of USB Devices” in the *System Administration Guide: Basic Administration*.

USB 2.0 Devices

This description has been updated in the Solaris Express 9/03 release.

USB 2.0 devices are defined as high-speed devices that follow the USB 2.0 specification. You can refer to the USB 2.0 specification at <http://www.usb.org>.

Some of the USB devices that are supported on SPARC based and x86 based systems in this Solaris release are as follows:

- Mass storage devices – CD-RWs, hard disks, DVD, digital cameras, Zip drives, diskettes, and tape drives
- Keyboard, mouse devices, speakers, and microphones
- Audio devices

For a full listing of USB devices that have been verified on the Solaris release, go to:

http://www.sun.com/io_technologies/USB.html

Additional storage devices might work by modifying the `scsa2usb.conf` file. For more information, see the `scsa2usb(7D)` man page.

Solaris USB 2.0 device support includes the following features:

- Increased USB bus speed from 12 Mbps to 480 Mbps. This increase means devices that support the USB 2.0 specification can run significantly faster than their USB 1.1 counterparts when they are connected to a USB 2.0 port.

A USB 2.0 port is defined on SPARC and x86 systems as follows:

- A port on a USB 2.0 PCI card
- A port on a USB 2.0 hub that is connected to a USB 2.0 port
- **x86 only** - A system USB port, assuming that it has a USB 2.0 port on the motherboard
- USB 2.0 is Solaris Ready on all PCI-based platforms. A USB 2.0 PCI card is needed to provide USB 2.0 ports. For a list of USB 2.0 PCI cards that have been verified for the Solaris release, go to http://www.sun.com/io_technologies/USB.html.
- USB 1.1 devices work as they have in the past, even if you have both USB 1.1 and USB 2.0 devices on the same system, with one exception. See note.

Note – USB 1.1 devices will not operate when connected to a USB 2.0 hub that is connected to a USB 2.0 port.

- While USB 2.0 devices operate on a USB 1.x port, their performance is significantly better when connected to a USB 2.0 port.

For more information on USB 2.0 device support, see the `ehci(7D)` and `usba(7D)` man pages.

For information about USB cables and bus-powered devices, see “About USB in the Solaris Environment” in the *System Administration Guide: Basic Administration*.

Solaris Support for USB Devices

This description is new in the Solaris Express 9/03 release.

The following table describes Solaris support for USB 1.1 and USB 2.0 devices.

	Solaris 8 HW* Releases	Solaris 9 Releases	Solaris 10 Release
USB 1.1	SPARC and x86	SPARC and x86	SPARC and x86
USB 2.0	SPARC	N/A	SPARC and x86

*Note that this term does not describe the Solaris 8 releases, but the Solaris 8 Hardware (HW) releases, starting with the Solaris 8 HW 5/03 release. The patch number for the Solaris 8 HW 5/03 release is 109896.

USB Mass Storage Devices

This description has been updated in the Solaris Express 9/03 release.

All USB storage devices in the Solaris Express releases are now accessed as removable media devices through volume management. This change has the following advantages:

- USB storage devices with standard MS-DOS or Windows (FAT) file systems are now supported.
- You can use the user-friendly `rmformat` command instead of the `format` command to format and partition all USB storage devices. If you need the functionality of the `format` command, use the `format -e` command.
- You can use the `fdisk` command if you need to do `fdisk`-style partitioning.
- Nonroot users can now access USB storage devices, because the root-privileged `mount` command is no longer needed. The device is automatically mounted by `vold` and is available under the `/rmdisk` directory. If a new device is connected while the system is down, do a reconfiguration boot with the `boot -r` command so that `vold` recognizes the device. If a new device is connected while the system is up, restart `vold`. For more information, refer to the `vold(1)` and `scsa2usb(7D)` man pages.
- Disks with FAT file systems can be mounted and accessed. For example:

```
mount -F pcfs /dev/dsk/c2t0d0s0:c /mnt
```
- All USB storage devices are now power managed, except for those that support LOG SENSE pages. Devices with LOG SENSE pages are usually SCSI drives that are connected through a USB-to-SCSI bridge device. In previous Solaris releases, some USB storage devices were not power managed because they were not recognized as removable media.
- Applications might work differently with USB mass storage devices. Note the following issues when using applications with USB storage devices:
 - Applications might make incorrect assumptions about the size of the media because only smaller devices such as diskettes and Zip drives were removable previously.
 - Requests by applications to eject media on devices where this would be inapplicable, such as a hard drive, succeed and do nothing.

To revert to the behavior of previous Solaris releases that did not treat all USB mass storage as removable media devices, update the `/kernel/drv/scsa2usb.conf` file.

For more information about using USB mass storage devices, see the `scsa2usb(7D)` man page.

For information on troubleshooting USB mass storage device problems, see “What’s New in USB Devices?” in the *System Administration Guide: Basic Administration*.

USB Driver Enhancements

This description has been updated in the Solaris Express 9/03 release.

This section describes USB driver enhancements in the Solaris Express releases.

- **New generic USB driver** – USB devices can now be accessed and manipulated by applications using standard UNIX `read(2)` and `write(2)` system calls, and without writing a special kernel driver. Additional features include the following:

- Applications have access to raw device data and device status.
- Supports control, bulk, and interrupt (in and out) transfers.

For more information, refer to the `ugen(7D)` man page and the USB DDK at <http://developers.sun.com/solaris/developer/support/driver/usb.html>

- **Digi Edgeport USB support** – Provides support for several Digi Edgeport USB-to-serial-port converter devices.
 - New devices are accessed as `/dev/term/[0-9]*` and `/dev/cua/[0-9]*`.
 - USB serial ports are usable as any other serial port would be, except that they cannot serve as a local serial console. The fact that their data is run through a USB port is transparent to the user.

For more information, see the `usbser_edge(7D)` man page, or go to <http://www.digi.com> and <http://www.sun.com/io>.

- **Documentation and binary support for user-written kernel and userland drivers** – A Solaris 9 USB Driver Development Kit (DDK) is available, and its documentation is applicable to the Solaris 10 release. For up-to-date information about USB driver development, including information on the DDK, go to <http://developers.sun.com/solaris/developer/support/driver/usb.html>

The EHCI and OHCI Drivers

This description is new in the Solaris Express 9/03 release.

Features of the EHCI driver include the following:

- Compliance with enhanced host controller interface that supports USB 2.0.
- Support for high-speed control, bulk, and interrupt transfers.
- Currently, no support is available for high-speed isochronous or split transactions. For example, you cannot connect USB 1.x devices to a 2.0 hub that is connected to a USB 2.0 port.

Note the following when using EHCI and OHCI controlled devices:

- A USB 2.0 host controller has one high-speed Enhanced Host Controller (EHCI) and one or more low- or full-speed OpenHCI Host Controller (OHCI) embedded controllers. Devices connected to a USB 2.0 port are dynamically assigned to either an EHCI or OHCI controller, depending on whether or not they support USB 2.0.

- If USB 2.0 and USB 1.x devices are on the system, the EHCI and OHCI drivers “hand-off” device control depending on the type of device that is connected to the system.

Note – USB 2.0 storage devices connected to a port on a USB 2.0 PCI card, and that were used with a prior Solaris release in the same hardware configuration, can change device names after upgrading to this release. This change occurs because these devices are now seen as USB 2.0 devices and are taken over by the EHCI controller. The controller number, *w* in `/dev/[r]dsk/cwtxyzsz`, is changed for these devices.

Logical Unit Reset

This feature was introduced in the Solaris Express pilot program.

The Solaris SCSI disk driver now supports SCSI logical unit reset. This feature improves error recovery processing on multi-LUN SCSI target devices. This feature can reset a logical unit without affecting other logical units on that device.

The ability to reset LUNs is particularly useful in conjunction with multiple pathing and with storage units that provide a large number of LUNs. For more information, see the `scsi_reset(9F)` and the `tran_reset(9E)` man pages.

File System Enhancements

This section describes all file system enhancements that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases.

NFS version 4

This feature was introduced in the Solaris Express pilot program.

The Solaris Express releases include the Sun implementation of the NFS version 4 distributed file access protocol. This version is the next logical step in the evolution of NFS. The NFS version 4 protocol, specified in RFC 3530, was created under the auspices of the Internet Engineering Task Force (IETF). This version is designed to be both vendor neutral and operating system neutral.

NFS version 4 integrates file access, file locking, and mount protocols into a single, unified protocol to ease traversal through a firewall and improve security. The Solaris implementation of NFS version 4 is fully integrated with Kerberos V5, also known as SEAM, thus providing authentication, integrity, and privacy. NFS version 4 also enables the negotiation of security flavors to be used between the client and the server. With NFS version 4, a server can offer different security flavors for different file systems.

The Solaris implementation of NFS version 4 includes delegation, a technique by which the server can delegate the management of a file to a client. This technique can reduce the number of round-trip operations because the client is guaranteed that no modification can occur without the server informing the client. The protocol also includes operation compounding, which allows multiple operations to be combined into a single “over-the-wire” request.

For more information about NFS version 4, refer to “Accessing Network File Systems (Reference)” in the *System Administration Guide: Network Services*.

Enhancements to the NFS Client

This feature was introduced in the Solaris Express pilot program.

The following enhancements have improved the performance of the NFS client:

- Restrictions on wire transfer sizes have been relaxed. Now, the transfer size is based on the capabilities of the underlying transport. For example, the NFS transfer limit for UDP is still 32 Kbytes. However, because TCP is a streaming protocol without the datagram limits of UDP, maximum transfer sizes over TCP have been increased to 1 Mbyte.
- Previously, all write requests were serialized by both the NFS client and the NFS server. The NFS client has been modified to permit an application to issue concurrent writes, as well as concurrent reads and writes, to a single file. You can enable this functionality on the client by using the `forcedirectio` mount option. When you use this option, you are enabling this functionality for all files within the mounted file system. You could also enable this functionality on a single file on the client by using the `directio()` interface. Note that unless this new functionality has been enabled, writes to files are serialized. Also, if concurrent writes or concurrent reads and writes are occurring, then POSIX semantics are no longer being supported for that file.
- The NFS client no longer uses an excessive number of UDP ports. Previously, NFS transfers over UDP used a separate UDP port for each outstanding request. Now, by default, the NFS client uses only one UDP reserved port. However, this support is configurable. If the use of more simultaneous ports would increase system performance through increased scalability, then the system can be configured to use more ports. This capability also mirrors the NFS over TCP support, which has had this kind of configurability since its inception.

For further information, see the *System Administration Guide: Network Services*.

SPARC: Multiterabyte UFS File Systems

This feature was introduced in the Solaris Express pilot program.

The Solaris Express releases provide support for multiterabyte UFS file systems on systems that run a 64-bit Solaris kernel. Previously, UFS file systems were limited to approximately 1 terabyte (Tbyte) on both 64-bit systems and 32-bit systems. All UFS file system commands and utilities have been updated to support multiterabyte UFS file systems.

You can initially create a UFS file system that is less than one Tbyte. You can specify that the file system can eventually be grown to a multiterabyte file system by using the `newfs -T` command. This command sets the inode and fragment density to scale appropriately for a multiterabyte file system.

Support for a multiterabyte UFS file system assumes the availability of multiterabyte LUNs. These LUNs are provided as Solaris Volume Manager or Veritas VxVM volumes, or as physical disks that are greater than one Tbyte.

Features of multiterabyte UFS file systems include the following:

- You can create a UFS file system to a maximum of 16 Tbytes in size.
- You can create a file system that is less than 16 Tbytes, which can later be increased in size to a maximum of 16 Tbytes.
- Multiterabyte file systems can be created on physical disks, Solaris Volume Manager's logical volumes, and Veritas's VxVM logical volumes.
- UFS logging is enabled by default on file systems greater than 1 terabyte. Multiterabyte file systems benefit from the performance improvements of having UFS logging enabled. Multiterabyte file systems also benefit from the availability of logging because the `fsck` command might not have to be run when logging is enabled.

Limitations of multiterabyte UFS file systems include the following:

- You cannot mount a file system that is greater than 1 Tbyte on a system that runs a 32-bit Solaris kernel.
- You cannot boot from a file system that is greater than 1 Tbyte on a system that runs a 64-bit Solaris kernel. This limitation means that you cannot put a `root (/)` file system on a multiterabyte file system.
- There is no support for individual files greater than 1 terabyte.
- The maximum number of files per terabyte of UFS file system is 1 million. This limit is intended to reduce the time it takes to check the file system with the `fsck` command.
- The maximum quota that you can set on a multiterabyte UFS file system is 2 Tbytes of 1024-byte blocks.
- Using the `fsnap` command to create a snapshot of a multiterabyte UFS file system is not currently supported.

For more information, see "What's New in File Systems?" in the *System Administration Guide: Basic Administration*.

Device File System (devfs)

This feature was introduced in the Solaris Express pilot program.

The `devfs` file system manages devices in the Solaris Express releases. Users continue to access all devices through entries in the `/dev` directory. These entries are symbolic links to entries in the `/devices` directory. The content of the `/devices` directory is now controlled by the `devfs` file system. The entries in the `/devices` directory dynamically represent the current state of accessible devices on the system. These entries require no administration.

The `devfs` file system provides the following enhancements:

- Operations in the `/devices` directory result in attaching device entries. Unused device entries are detached.
- System boot performance is increased because only device entries that are needed to boot the system are attached. New device entries are added as the devices are accessed.

For more information, see the `devfs(7FS)` man page.

SPARC: Multiterabyte Volume Support With EFI Disk Labels

This feature was introduced in the Solaris Express pilot program.

The Solaris Express releases provide support for disks that are larger than 1 terabyte (Tbyte) on systems that run a 64-bit Solaris kernel.

The Extensible Firmware Interface (EFI) label provides support for physical disks and virtual disk volumes. The UFS file system is compatible with the EFI disk label, and you can create a UFS file system that is greater than 1 Tbyte. This release also includes updated disk utilities for managing disks that are greater than 1 Tbyte.

The EFI disk label differs from the VTOC disk label in the following ways:

- Support for disks that are greater than 1 Tbyte in size is provided.
- Slices 0–6, where slice 2 is just another slice, are provided.
- Partitions, or slices, cannot overlap with the primary or backup label, nor with any other partitions. The size of the EFI label is usually 34 sectors, so partitions start at sector 34. This feature means that no partition can start at sector zero (0).
- No cylinder, head, or sector information is stored in the label. Sizes are reported in blocks.
- Information that was stored in the alternate cylinders area, the last two cylinders of the disk, is now stored in slice 8.

For more information on using the EFI disk label, see the *System Administration Guide: Basic Administration*. This guide contains important information and restrictions. This information concerns using the EFI disk label with existing software products.

The Solaris Volume Manager software can also be used to manage disks that are greater than 1 Tbyte in this Solaris release. See “SPARC: Multiterabyte Volume Support in Solaris Volume Manager” on page 20.

New Configuration File for the Autofs Environment

This feature was introduced in the Solaris Express pilot program.

The new configuration file for your autofs environment, `/etc/default/autofs`, provides an additional way to configure your autofs commands and autofs daemons. Now, the same specifications that you would make on the command line can be made in this new configuration file. However, unlike the specifications you would make on the command line, this file preserves your specifications, even during upgrades to your operating system. Additionally, you no longer are required to update critical startup files to ensure that the existing behavior of your autofs environment is preserved. You can make your specifications by using the following keywords:

- `AUTOMOUNTD_ENV` permits you to assign different values to different environments. This keyword is the equivalent of the `-D` argument for `automountd`.
- `AUTOMOUNTD_NOBROWSE` turns browsing on, or turns browsing off, for all autofs mount points. This command is the equivalent of the `-n` argument for `automountd`.
- `AUTOMOUNTD_TRACE` expands each remote procedure call (RPC) and displays the expanded RPC on standard output. This keyword is the equivalent of the `-T` argument for `automountd`.
- `AUTOMOUNTD_VERBOSE` logs status messages to the console and is the equivalent of the `-v` argument for the `automountd` daemon.
- `AUTOMOUNT_TIMEOUT` sets the duration for a file system to remain idle before the file system is unmounted. This keyword is the equivalent of the `-t` argument for the `automount` command.
- `AUTOMOUNT_VERBOSE` provides notification of `autofs` mounts, unmounts, and other nonessential events. This keyword is the equivalent of the `-v` argument for `automount`.

For more information, refer to the `automount(1M)` and the `automountd(1M)` man pages.

System Resources Enhancements

This section describes all system resources enhancements that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases.

Extended Accounting Subsystem

This feature was introduced in the Solaris Express pilot program.

You can now use extended accounting processes in conjunction with the flow accounting module for IPQoS. For information about IPQoS, see the *IPQoS Administration Guide*.

A Perl interface to `libexacct` is available in this Solaris release. This interface enables you to create Perl scripts that can read the accounting files which are produced by the `exacct` framework. You can also create Perl scripts that write `exacct` files. The new interface is functionally equivalent to the underlying C API.

You can use the Perl interface to record system resource consumption on a task or process basis. Or, you can record consumption on the basis of selectors provided by the IPQoS `flowacct` module.

For further information, see the following man pages:

- `Exacct(3PERL)`
- `Exacct::Catalog(3PERL)`
- `Exacct::File(3PERL)`
- `Exacct::Object(3PERL)`
- `Exacct::Object::Group(3PERL)`
- `Exacct::Object::Item(3PERL)`
- `Kstat(3PERL)`
- `Project(3PERL)`
- `Task(3PERL)`

Physical Memory Control Using the Resource Capping Daemon

This feature was introduced in the Solaris Express pilot program.

Physical memory control that uses the resource capping daemon is an optional feature. The resource capping daemon `rcapd` regulates the consumption of physical memory by processes that run in projects that have defined resource caps. Associated utilities provide mechanisms for administering the daemon and reporting related statistics.

For additional information, see the man pages `rcapstat(1)`, `rcapadm(1M)`, `project(4)`, and `rcapd(1M)`.

System V IPC Resource Controls

This feature was introduced in the Solaris Express pilot program.

New project-based and new process-based resource controls (`rctls`) have been added. Resource controls provide a mechanism for controlling how applications use system resources.

For the current list of available resource controls, see the `rctladm(1M)` man page. For information about configuring resource controls, see the `project(4)` man page.

These new resource controls affect the System V IPC (interprocess communication) facilities, which include shared memory, message queues, and semaphores. See “System V IPC Configuration” on page 17 for more information.

Desktop Features

This section describes all desktop features that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases.

GNOME 2.0 Desktop

This feature was introduced in the Solaris Express pilot program.

The GNOME 2.0 desktop is an advanced, intuitive desktop for the Solaris Operating System. The GNOME 2.0 desktop provides an alternative to the Common Desktop Environment (CDE), which was provided with the Solaris environment through several releases.

The GNOME 2.0 desktop provides a familiar environment that enables users to easily interact with the applications and documents on their desktop. The following features are included in the GNOME 2.0 desktop.

- An advanced file manager
- A comprehensive online help system
- Software utilities and accessories
- Panels for launching applications and managing application windows and multiple workspaces

The GNOME 2.0 desktop includes the following important components:

1. Nautilus is an advanced file manager that enables users to organize and locate applications and documents. Nautilus also enables users to view the contents of text and picture files directly in the file manager.
2. Panels provide a quick way to launch applications and view the system status. You can create an unlimited number of panels. The Panel bar on the desktop contains the following features.
 - Application launchers
 - System menus
 - Panel drawers (subpanels)
 - Applets
3. Applications support a variety of everyday tasks. The following applications are provided in the GNOME 2.0 desktop.
 - Text Editor
 - Calculator
 - Character Map
 - Image Viewer
 - PostScript™ or PDF Viewer
 - Media Player
 - Sound Recorder
 - Performance Meter
 - Print Manager

- Terminal
- 4. Applets are sets of utilities that run in a panel. The GNOME 2.0 desktop includes the following applets.
 - Clock
 - Inbox Monitor
 - CD Player
 - Volume Control
 - Character Palette
 - Command Line
 - Keyboard Layout Switcher
 - Window List
 - Workspace Switcher

For further information on the GNOME 2.0 desktop, see the following Web site:

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

X11 Windowing Features

This section describes all X11 windowing features that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases.

Xscreensaver Program

This feature was introduced in the Solaris Express pilot program.

The Xscreensaver program locks your monitor screen after a period of inactivity in order to prevent others from gaining access to your system session. Xscreensaver supports a variety of animation displays that can be used while the screen is locked. This program is used in the GNOME 2.0 desktop.

X Event Interception Extension (XEvIE)

This feature was introduced in the Solaris Express pilot program.

XEvIE is a low-level interface that intercepts all the keyboard and mouse events so that these events can be read, consumed, or modified as needed. This X extension enables better integration of assistive technologies, including those technologies that are planned for inclusion in the GNOME 2.0 Desktop.

FreeType 2.1.x

This feature was introduced in the Solaris Express pilot program.

FreeType 2.1.x is an open source library that provides a simple application programming interface (API). This API can access font content in a uniform way, independently of the file format. Additionally, some format-specific APIs can be used to access special data in the font file.

Xserver Virtual Screen Enhancement

This feature was introduced in the Solaris Express pilot program.

The Xserver Virtual Screen enhancement provides accessibility support for the GNOME 2.0 Desktop for the Solaris Operating System. This capability enables support of magnification software on systems that have just one frame buffer.

Xrender Extension

This feature was introduced in the Solaris Express pilot program.

The new Xrender feature increases performance for applications, such as the StarOffice™ software suite, that run on the Solaris Operating System. The Xrender feature provides a modern appearance for these applications. Xrender uses hardware processing for alpha-blending and transparency effects.

Web Browser Enhancements

This section describes all web browser features that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases.

Netscape 7.0 for the Solaris Release

This feature was introduced in the Solaris Express pilot program.

The Netscape™ 7.0 cross-platform browser integrates features for navigating the Web, communicating with colleagues, participating in discussion groups, and creating dynamic Web pages. The following features are provided in Netscape 7.0.

- Fast and efficient browsing tools
- Integrated mail and instant messaging enhancements
- Faster and streamlined search functions
- Privacy that is more secure
- Enterprise functionality

Netscape 7.0 includes the following highlights.

- Internet Support is provided by the following applications.
 - Netscape Navigator™
 - Netscape Mail
 - Netscape Instant Messenger
 - Netscape Composer
 - Netscape Address Book
 - Additional utilities and plug-ins
- Support for the latest industry standards, including Java technology, IPv6, Platform for Privacy Preferences Project (P3P), XML, Cascading Style Sheets, level 1 (CSS1), and data output messaging (DOM) – Enables the creation of a new class of Web applications.

- My Sidebar – Connects users to important information such as breaking news stories, stock portfolios, buddy lists, and auctions.
- Fully customized, integrated search – Helps provide faster access to information, and eliminates waiting to load a search page or open a frame.
- Instant messaging that is integrated with browsing and mail – Increases productivity and enhances your communication with colleagues and friends.
- Themes – Enables users to add personal touches to their browsers. Web sites and businesses can create browsers to their customers' specifications.
- Flexible, modular design – Enhances the user's Web experience and increases productivity.
- Global support for English, European, and Asian languages.
- Integrated Java technology – Delivers cross-platform, seamless access to browser-based applications.

For more information on Netscape 7.0, see the following Web site:

<http://www.sun.com/software/solaris/netscape>

Freeware Enhancements

This section describes all freeware features that are new or have been revised in the Solaris Express 11/03 release and in all prior Solaris Express releases.

Note – To view license terms, attribution, and copyright statements for freeware, check under the license path that is provided for each freeware description. If the Solaris Operating System has been installed anywhere other than the default, modify the path to access the file at the installed location.

Ghostscript 7.05

This feature was introduced in the Solaris Express pilot program.

The following new freeware package is included in the Solaris Operating System.

Ghostscript 7.05 – Reads PostScript and PDF files. Displays these files on screen or converts the files to a form that is usable by many printers. See the `gs(1)` man page under `/usr/sfw/share/man`.

Note – To view license terms, attribution, and copyright statements for Ghostscript, the license path is `/usr/sfw/share/src/<freeware name>`.

New Freeware Packages: libxml2 2.4.16 and libxslt 1.0.19

This feature was introduced in the Solaris Express pilot program.

The following new freeware packages are new or revised in the Solaris Express releases:

- libxml2 2.4.16 – A standard for building tag-based structured documents or data
- libxslt 1.0.19 – An XML language to define transformation for XML

Note – To view license terms, attribution, and copyright statements for these packages, the license path is `/usr/share/src/<freeware name>/filename>`.

ANT 1.4.1 Freeware Package

This feature was introduced in the Solaris Express pilot program.

The following new freeware package is included in the Solaris Operating System: ANT 1.4.1 – This is a Jakarta ANT Java and XML-based build package.

Note – To view license terms, attribution, and copyright statements for ANT, the license path is `/usr/sfw/share/src/<freeware name>`.
