System Management Services (SMS) 1.6 Reference Manual

Sun Microsystems, Inc.
www.sun.com

Part No. 819-4662-11
June 2006, Revision A

Submit comments about this document at: http://www.sun.com/hwdocs/feedback

Contents

Preface

This reference manual comprises the System Management Services (SMS) 1.6 man pages. Both novice users and those familiar with the Solaris™ Operating System can use online man pages to obtain information about the system and its features. A man page is intended to answer concisely the question "What does it do?" Man pages are generally intended for reference, not a tutorial.

Overview

This section contains a brief description of the SMS man pages and the information they contain. The `intro`(1M) man page describes, in alphabetical order, commands that are used chiefly for system maintenance and administration purposes.

What follows is a generic format for man pages. Each man page is made up of the following sections, which usually appear in the order shown here. When a particular section is not needed for a particular command, it is omitted. For example, if there are no notes to report, there is no NOTES section. See the `intro`(1) man page for more information and detail about each section, and `man`(1) for more information about man pages in general.

NAME	This section gives the names of the commands or functions documented, followed by a brief description of what they do.

SYNOPSIS

This section shows the syntax of commands or functions. When a command or file does not exist in the standard path, its full path name is shown. Options and arguments are alphabetized, with single-letter arguments first, and options with arguments next, unless a different argument order is required.

The following special characters are used in this section:

`[ ]` Brackets. The option or argument enclosed in these brackets is optional. If the brackets are omitted, the argument must be specified.

… Ellipses. Several values may be provided for the previous argument, or the previous argument can be specified multiple times, for example `"filename..."`.

| Separator. Only one of the arguments separated by this character can be specified at one time.

`{ }` Braces. The options and/or arguments enclosed within braces are interdependent, such that everything enclosed must be treated as a unit.

DESCRIPTION

This section defines the functionality and behavior of the service. Thus it describes concisely what the command does. It does not discuss OPTIONS or cite EXAMPLES. Interactive commands, subcommands, requests, macros, functions and such, are described under USAGE.

OPTIONS

This lists the command options with a concise summary of what each option does. The options are listed literally and in the order in which they appear in the SYNOPSIS section. Possible arguments to options are discussed under the option, and where appropriate, default values are supplied.

OPERANDS

This section lists the command operands and describes how they affect the actions of the command.

EXTENDED DESCRIPTION	Provides additional information about the command and its options, beyond the basic information outlined in the Description section.
RETURN VALUES	If the man page documents functions that return values, this section lists these values and describes the conditions under which they are returned. If a function can return only constant values, such as 0 or –1, these values are listed in tagged paragraphs. Otherwise, a single paragraph describes the return values of each function. Functions declared void do not return values, so they are not discussed in RETURN VALUES.
ERRORS	On failure, most functions place an error code in the global variable `errno`, indicating why they failed. This section lists alphabetically all the error codes a function can generate and describes the conditions that cause each error. When more than one condition can cause the same error, each condition is described in a separate paragraph under the error code.
USAGE	This section lists special rules, features, and commands that require in-depth explanation. The subsections listed below are used to explain built-in functionality: Commands Modifiers Variables Expressions Input Grammar
EXAMPLES	This section provides examples of usage, including how to use a command or function. Wherever possible, a complete example, including command line entry and machine response, is shown. Examples are followed by explanations, variable substitution rules, or returned values. Most examples illustrate concepts from the SYNOPSIS, DESCRIPTION, OPTIONS, and USAGE sections.
ENVIRONMENT VARIABLES	This section lists any environment variables that the command or function affects, followed by a brief description of the effect.

EXIT STATUS	This section lists the values the command returns to the calling program or shell and the conditions that cause these values to be returned. Usually, zero is returned for successful completion and values other than zero for various error conditions.
FILES	This section lists all file names referred to by the man page, files of interest, and files created or required by commands. Each file name is followed by a descriptive summary or explanation.
ATTRIBUTES	This section lists characteristics of commands, utilities, and device drivers by defining the attribute type and its corresponding value. See `attributes`(5) for more information.
SEE ALSO	This section lists references to other man pages, in-house documentation, and outside publications.
NOTES	This section lists additional information that does not belong anywhere else on the page. It takes the form of an aside to the user, covering points of special interest. Critical information is never covered here.

NAME

Intro - SMS Administration.

DESCRIPTION

This section describes the commands executed in the system management software environment..

LIST OF COMMANDS

The following commands are supported:

`Intro`	SMS Administration
`addboard`	assign, connect and configure a board to a domain
`addcodlicense`	add a Capacity on Demand (COD) right-to-use (RTU) license key to the COD license database
`addtag`	assign a domain name (tag) to a domain
`cancelcmdsync`	command synchronization command
`codd`	Capacity on Demand (COD) daemon
`console`	access the domain console
`dca`	domain configuration agent
`deleteboard`	unconfigure, disconnect, and unassign a system board from a domain
`deletecodlicense`	remove a Capacity on Demand (COD) right-to-use (RTU) license key from the COD license database
`deletetag`	remove the domain tag name associated with the domain
`disablecomponent`	add the specified component to the specified blacklist file
`dsmd`	domain status-monitoring daemon
`dxs`	domain X server
`enablecomponent`	remove the specified component from the specified blacklist
`esmd`	environmental status monitoring daemon
`flashupdate`	update the flash PROMs located on the CPU boards, MaxCPU boards, and system controllers (SCs)
`fomd`	failover management daemon
`frad`	FRU access daemon
`help`	display help information for SMS commands
`hpost`	Sun Fire high-end system power-on self-test (POST) control application

`hwad`	hardware access daemon
`initcmdsync`	command synchronization command
`kmd`	SMS key management daemon
`mand`	management network daemon
`mld`	message-logging daemon
`moveboard`	move a board from one domain to another
`osd`	OpenBoot PROM server daemon
`pcd`	platform configuration database daemon
`poweroff`	control power off
`poweron`	control power on
`rcfgadm`	remote configuration administration
`reset`	send reset to all CPU ports of a specified domain
`resetsc`	reset the other system controller (SC)
`runcmdsync`	prepare a specified script for recovery after a failover
`savecmdsync`	saves the command synchronization command settings
`setbus`	perform dynamic bus reconfiguration on active expanders in a domain
`setcsn`	set the chassis serial number on the CPU
`setdatasync`	modify the data propagation list used in data synchronization
`setdate`	set the date and time for the system controller (SC) or a domain
`setdefaults`	remove all instances of a previously active domain and reset Capacity on Demand (COD) information
`setfailover`	modify the state of the system controller (SC) failover mechanism
`setkeyswitch`	change the position of the virtual keyswitch
`setobpparams`	set up OpenBoot PROM variables for a domain
`setpcimode`	Forces a PCI-X slot to run in PCI mode.
`setupplatform`	set up the available component list and Capacity on Demand (COD) resources used for domains
`showboards`	show the assignment information and status of the boards
`showbus`	display the bus configuration of expanders in active domains
`showcmdsync`	display the current command synchronization list

`hwad`	hardware access daemon
`initcmdsync`	command synchronization command
`kmd`	SMS key management daemon
`mand`	management network daemon
`mld`	message-logging daemon
`moveboard`	move a board from one domain to another
`osd`	OpenBoot PROM server daemon
`pcd`	platform configuration database daemon
`poweroff`	control power off
`poweron`	control power on
`rcfgadm`	remote configuration administration
`reset`	send reset to all CPU ports of a specified domain
`resetsc`	reset the other system controller (SC)
`runcmdsync`	prepare a specified script for recovery after a failover
`savecmdsync`	saves the command synchronization command settings
`setbus`	perform dynamic bus reconfiguration on active expanders in a domain
`setcsn`	set the chassis serial number on the CPU
`setdatasync`	modify the data propagation list used in data synchronization
`setdate`	set the date and time for the system controller (SC) or a domain
`setdefaults`	remove all instances of a previously active domain and reset Capacity on Demand (COD) information
`setfailover`	modify the state of the system controller (SC) failover mechanism
`setkeyswitch`	change the position of the virtual keyswitch
`setobpparams`	set up OpenBoot PROM variables for a domain
`setpcimode`	Forces a PCI-X slot to run in PCI mode.
`setupplatform`	set up the available component list and Capacity on Demand (COD) resources used for domains
`showboards`	show the assignment information and status of the boards
`showbus`	display the bus configuration of expanders in active domains
`showcmdsync`	display the current command synchronization list

`showcodlicense`	display the current Capacity on Demand (COD) right-to-use (RTU) licenses stored in the COD license database
`showcodusage`	display the current usage statistics for Capacity on Demand (COD) resources
`showcomponent`	display the blacklist status for a component
`showdatasync`	display the status of system controller (SC) data synchronization for failover
`showdate`	display the date and time for the system controller (SC) or a domain
`showdevices`	display system board devices and resource usage information
`showenvironment`	display the environmental data
`showfailover`	display system controller (SC) failover status or role
`showkeyswitch`	display the position of the virtual keyswitch
`showlogs`	display message log files or the event logs.
`showobpparams`	display OpenBoot PROM bring up parameters for a domain
`showpcimode`	Displays the mode settings for all PCI-X slots on the hsPCIX board.
`showplatform`	display the platform type, board available component list, the domain state for each domain, and Capacity on Demand (COD) information.
`showxirstate`	display CPU dump information only after a reset pulse has been sent to the processors
`smsbackup`	back up the SMS environment
`smsconfig`	configures the SMS environment
`smsconnectsc`	accesses a remote SC console
`smsinstall`	install the SMS environment
`smsrestore`	restore the SMS environment
`smsupgrade`	upgrades the SMS software to the current version
`smsversion`	change the active version of SMS to another co-resident version of the SMS software
`ssd`	SMS startup daemon
`tmd`	task management daemon
`wcapp`	wPCI application daemon

NAME

addboard - assign, connect and configure a board to a domain

SYNOPSIS

addboard `-d` *domain_indicator* [`-c` *function*] [`-r` *retry_count* [`-t` *timeout*]] [`-q`] [`-f`] [`-y|-n`] *location ...*

addboard `-h`

DESCRIPTION

`addboard`(1M) assigns, connects, and configures a *location* to the domain *domain_id* or *domain_tag.*

The board must be either `available` or `assigned` to the domain to which it is being added. The `-c` option is used to specify the transition of the board from the current configuration state to a new configuration state. Configuration states are `assign`, `connect`, and `configure`. If the `-c` option is not specified, the default expected configuration state is `configure`.

Note – `addboard` performs tasks synchronously and does not return control to the user until the command is complete. If the board is not powered on or tested and a `-c connect|configure` option is specified, then the command powers on the board and tests it. However, the new firmware will not be active until after `hpost` is run.

Note – If only one board is specified and it is in the automatic system recovery (ASR) blacklist file, `addboard` displays an error message and exits. If more than one board is specified, `addboard` displays a message that the board is being skipped, and then goes on to the next board or, after the last board, exits.

Note – If the `addboard` command fails, a board does not return to its original state. A `dxs` or `dca` error message is logged to the domain. If the error is recoverable, you can retry the command. If it is unrecoverable, you need to reboot the domain in order to use that board.

OPTIONS The following options are supported:

`-c` *function*

Valid *function* values are `assign`, `connect`, and `configure`. This option is used to control the configuration state transition. Each successive function builds upon the last. For example, `configure` first assigns and then connects the board before configuring it.

The possible transition states and their meanings are as follows:

`assign` — Assigns the board to the logical domain. This is a board state in which the domain has sole access to the board; however, the board is not active. Once assigned, the board can be connected or configured into the domain with the use of either `setkeyswitch on` or the `connect` or `configure` options.

`connect` — Assigns the board to the logical domain (if it is not already assigned). Transitions the board into the `connected|unconfigured` state. In this state the system board is assigned to the logical domain and connected (becomes `active`). This state allows normal system access to hardware resources on the board, but the hardware resources of the board are not represented by the normal Solaris software data structures and thus are not available for use by the Solaris operating system. Operations allowed on the board are limited to configuration administration operations. This is an intermediate state and does not have any standalone implementation at this time.

`configure` — Assigns the board to the logical domain (if it is not already assigned). Transitions the board into the `connected|configured` state. In this state the board is not only assigned, active, and connected to a domain, but also configured into the Solaris operating system. The hardware resources on the board can be used by Solaris software.

`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-f`	Forces the specified action to occur. Typically, this is a hardware-dependent override of a safety feature. Forcing a state change operation can allow use of the hardware resources of an occupant that is not in the `ok` or `unknown` conditions, at the discretion of any hardware-dependent safety checks.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-n`	Automatically answers no to all prompts. Prompts are displayed unless used with the `-q` option.
`-q`	Quiet. Suppresses all messages to `stdout` including prompts. When used alone `-q` defaults to the `-n` option for all prompts. When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts and automatically answers with either 'y' or 'n' based on the option chosen.
`-r` *retry_count*	This command argument enables the user to specify retries in case of failures encountered during state transitions. The `-r` *retry_count* option can be used alone and indicates the number of times the configuration state change request should be retried by the domain.
`-t` *timeout*	This command argument enables the user to specify retries in case of failures encountered during state transitions. The `-t` *timeout* option cannot be used without the `-r` *retry_count* option and specifies the number of seconds that the domain should wait before the next retry is made. If the `-t` *timeout* is not specified, the default *timeout* is zero, meaning that the request is retried immediately.
`-y`	Automatically answers yes to all prompts. Prompts are displayed unless used with the `-q` option.

OPERANDS

The following operands are supported:

location — List of board locations separated by a space. Multiple *location* arguments are permitted.

The following *location* forms are accepted:

Sun Fire 12K and E20K:

```
SB(0...8)
IO(0...8)
```

Sun Fire 15K and E25K:

```
SB(0...17)
IO(0...17)
```

Note – Use `showboards`(1M) to display board type.

EXTENDED DESCRIPTION

Group Privileges Required

If you have platform administrator privileges, you can perform only the `-c assign` option.

If you have domain administrator or configurator privileges, you can execute this command, but only on your respective domains. If the board(s) are not already assigned to the domain, the board(s) must be in the available component list of the domain.

Refer to Chapter 2 in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Assigning Boards to Domain C

To assign four boards to domain C you must have platform privileges or domain privileges and the boards must be in the domain available component list.

```
sc0:sms-user:> addboard -d C -c assign SB0  IO1 SB1 SB2
SB0 assigned to domain: C
IO1 assigned to domain: C
SB1 assigned to domain: C
SB2 assigned to domain: C
sc0:sms-user:>
```

EXAMPLE 2 Assigning a Blacklisted Board to Domain C

To assign four boards to domain C you must have platform privileges or domain privileges and the boards must be in the domain available component list.

```
sc0:sms-user:> addboard -d C -c assign SB0  IO2 SB1 SB2
SB0 assigned to domain: C
IO2 assigned to domain: C
Warning: IO2 is blacklisted.
You will not be able to connect or configure it.
SB1 assigned to domain: C
SB2 assigned to domain: C
sc0:sms-user:>
```

EXAMPLE 3 Connecting Boards to Domain A

This example connects three boards to domain A, setting retries to five and timeout to five seconds. You must have domain privileges for domain A.

```
sc0:sms-user:> addboard -d A -c connect -r 5 -t 5 IO3
IO4 IO5
```

EXAMPLE 4 Connecting Boards Containing an ASR Blacklisted Board to Domain C

You must have domain privileges for domain C. Blacklisted boards are skipped.

```
sc0:sms-user:> addboard -d C -c connect SB0
SB0 is blacklisted. Exiting.
sc0:sms-user:>
```

EXAMPLE 5 Configuring Boards to Domain A

You must have domain privileges for domain A.

```
sc0:sms-user:> addboard -d A -c configure IO3 IO4 IO5
```

EXAMPLE 6 Configuring Boards Containing an ASR Blacklisted Board to Domain A

You must have domain privileges for domain A. Blacklisted boards are skipped.

```
sc0:sms-user:> addboard -d A -c configure IO7 IO8 IO9
Skipping IO8. It is blacklisted.
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	No acknowledge.
2	Not supported.

3	Operation not supported.
4	Invalid privileges.
5	Busy.
6	System busy.
7	Data error.
8	Library error.
9	No library.
10	Insufficient condition.
11	Invalid.
12	Error.
13	A PID does not exist.
14	Invalid attribute.
30	Invalid board ID type.
31	Invalid permissions.
32	Assigned to another domain.
33	Unable to get permissions.
34	Unable to get domain board info.
35	Unable to get active board list.
36	Unable to get assigned board list.
38	Solaris not running.
39	Unable to assign/unassign.
40	Unable to get domain permissions.
41	Unable to get platform permissions.
51	Invalid domain.
52	Invalid privileges.
53	Internal error.
54	Library error.
56	DR command syntax error.
57	Location already assigned.
58	Internal error.

3	Operation not supported.
4	Invalid privileges.
5	Busy.
6	System busy.
7	Data error.
8	Library error.
9	No library.
10	Insufficient condition.
11	Invalid.
12	Error.
13	A PID does not exist.
14	Invalid attribute.
30	Invalid board ID type.
31	Invalid permissions.
32	Assigned to another domain.
33	Unable to get permissions.
34	Unable to get domain board info.
35	Unable to get active board list.
36	Unable to get assigned board list.
38	Solaris not running.
39	Unable to assign/unassign.
40	Unable to get domain permissions.
41	Unable to get platform permissions.
51	Invalid domain.
52	Invalid privileges.
53	Internal error.
54	Library error.
56	DR command syntax error.
57	Location already assigned.
58	Internal error.

59	Component blacklisted.
60	Unable to get ASR blacklist.
61	Unable to get domain blacklist.
62	Unable to get platform blacklist.
70	DR operation failed.

FILES

The following files are used by this command.

`/etc/opt/SUNWSMS/config/asr/blacklist`	List of components excluded by `esmd`.
`/etc/opt/SUNWSMS/config/platform/blacklist`	List of platform components excluded.
`/etc/opt/SUNWSMS/config/`*domain_id*`/blacklist`	List of domain components excluded.

Note – This file is created and used internally and should *not* be edited manually. To remove a component from the ASR blacklist file, use `enablecomponent`(1M).

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `enablecomponent(1M)`, `esmd(1M)`, `showcomponent(1M)`

NAME

addcodlicense - add a Capacity on Demand (COD) right-to-use (RTU) license key to the COD license database

SYNOPSIS

addcodlicense *license-signature*

addcodlicense `-h`

DESCRIPTION

`addcodlicense`(1M) adds the COD RTU specified license key to the COD license database on the system controller (SC).

Note – Before you run this command, you must obtain a COD license key from the Sun License Center. For details on COD RTU license keys, refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

OPTIONS

The following option is supported:

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

OPERANDS

The following operands are supported:

license-signature Specifies the COD RTU license key to be added to the COD license database.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator group privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Adding a COD RTU License Key

```
sc0:sms-user:> addcodlicense \
01:5014936C37048:01001:0201010302:4:20020430:jWGJdg/
kx78b0wyK2xrqIg
```

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`1` Invalid usage.

`2` Invalid group privileges.

3	Duplicate license exists in the COD license database.
4	Invalid license key.
>4	An internal error occurred. For further information, see `/var/opt/SUNWSMS/adm/platform/messages`.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`codd(1M)`, `deletecodlicense(1M)`, `showcodlicense(1M)`, `showcodusage(1M)`

NAME

addtag - assign a domain name (tag) to a domain

SYNOPSIS

addtag `-d` *domain_indicator* [`-q`] [`-y`|`-n`] *new_tag*

addtag `-h`

DESCRIPTION

`addtag`(1M) adds the specified domain tag name (*new_tag*) to a domain (*domain_id*|*domain_tag*). Only one name tag can be assigned to a domain, and it must be unique across all domains. `addtag` can also be used to change the *domain_tag*.

OPTIONS

The following options are supported:

`-d` *domain_indicator* — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain.

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-n` — Automatically answers no to all prompts. Prompts are displayed unless used with the `-q` option.

`-q` — Quiet. Suppresses all messages to `stdout` including prompts.

When used alone `-q` defaults to the `-n` option for all prompts.

When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts and automatically answers with either 'y' or 'n' based on the option chosen.

`-y` — Automatically answers yes to all prompts. Prompts are displayed unless used with the `-q` option.

OPERANDS

The following operands are supported:

new_tag — New tag name assigned to a domain. See Extended Description for a description of invalid domain names.

EXTENDED DESCRIPTION

Domain Name Tag Restrictions

The following restrictions are required on a domain name tag:

- No single character names are permitted.
- All domain name tags must be unique across all domains within a single chassis.
- Tags must adhere to the same restrictions as defined for Solaris software node names. Currently, the size restriction is set to 2 to 64 characters.

Group Privileges Required

You must have platform administrator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Assigning the Tag `eng2` to Domain A With Prompts

```
sc0:sms-user:> addtag -d A  eng2
```

If a tag for this domain exists, you are prompted.

EXAMPLE 2 Assigning the Tag `eng2` to Domain A Using the `-y` Option

```
sc0:sms-user:> addtag -d A  -y eng2
```

Prompts are displayed and are automatically answered yes. This forces the domain tag to be set even if a tag already exists for this domain.

EXAMPLE 3 Assigning the Tag `eng2` to Domain A Using the `-n` Option

```
sc0:sms-user:> addtag -d A  -n eng2
```

Prompts are displayed and are automatically answered no. This sets the tag for this domain unless it has already been set.

EXAMPLE 4 Assigning the Tag `eng2` to Domain A Using the `-qy` Options

```
sc0:sms-user:> addtag -d A  -qy eng2
```

You are not prompted.

EXAMPLE 5 Assigning the Tag `eng2` to Domain A Using the `-qn` Options

```
sc0:sms-user:> addtag -d A  -qn eng2
```

The example assigns the tag `eng2` to Domain `A` only if it has not already been set. You are not prompted.

EXAMPLE 6 Assigning the Tag `eng2` to Domain A Using the `-q` Option

```
sc0:sms-user:> addtag -d A  -q  eng2
```

The example assigns the tag `eng2` to Domain `A` if it is not already set. If it is set, the command does not change it. You are not prompted.

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See **attributes**`(5)` for descriptions of the following attributes

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`deletetag(1M)`

NAME

cancelcmdsync - command synchronization command

SYNOPSIS

cancelcmdsync *cmdsync_descriptor*

initcmdsync *script_name* [*parameters*]

savecmdsync `-M` *identifier cmdsync_descriptor*

[cancel | init | save]cmdsync `-h`

DESCRIPTION

The command synchronization commands work together to control the recovery of user-defined scripts interrupted by a system controller (SC) failover. Insert the following commands in user-defined scripts to enable command synchronization:

`initcmdsync`	Creates a command synchronization descriptor that identifies the script to be recovered. This descriptor is placed on a command synchronization list that identifies the scripts and commands to be restarted on the new main SC after a failover.
`savecmdsync`	Adds a marker that identifies a location in the script from which processing can be resumed after a failover.
`cancelcmdsync`	Removes a command synchronization descriptor from the command synchronization list. This ensures that the script is run only once and not after subsequent failovers. Be sure that all exit paths of a script have a `cancelcmdsync` sequence to remove the descriptor from the command synchronization list. If you do not remove the descriptor and a failover occurs, the script is rerun on the new main SC.

Note – Both an `initcmdsync` and a `cancelcmdsync` sequence must be contained within a script to enable command synchronization. The use of the `savecmdsync` command is optional and marks only specific points in a script from which processing can be resumed. If specific restart points are not necessary, consider using `runcmdsync`(1M) instead.

OPTIONS

The following options are supported:

cmdsync_descriptor — Specifies the command synchronization descriptor that identifies the user-defined script. This descriptor is the standard output value returned by the `initcmdsync` command.

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-M` *identifier* — Marks a location in the script from which the script can be resumed after a failover. The identifier must be a positive integer.

parameters — Specifies the options or parameters associated with the user-defined script. These parameters are stored on the spare SC and are used to restart the specified script after a failover.

script_name — Identifies the name of the user-defined script to be synchronized. *script_name* must be the absolute path name of an executable command. The command must exist in the same location on both SCs.

EXTENDED DESCRIPTION

The command synchronization commands are inserted at certain logical points within a user-defined script.

For instance, a Korn shell script might be structured as follows:

```
# MAIN CODE STARTS HERE
# Be sure to use a cleanup procedure to handle any
# interrupts.
# Use the cancelcmdsync to remove the script from the
# command synchronization list. Otherwise, the command
# will get restarted on the new main SC.
#
clean_up () {
        cancelcmdsync  $desc
        exit
    }

# Declare the clean_up function to capture system signals
# and cleanup.
trap "clean_up" INT HUP TERM QUIT PWR URG
goto_label=1
# Process the arguments, capturing the -M marker point
# if provided
#
for arg in $*; do
       case $arg in
            -M )
       goto_label=$arg;;
       .
       .
       .
       esac
   done
# Place this script and all its parameters in the command
# synchronization list, which indicates the commands to
# be restarted after an SC failover.
#
# NOTE: The script must be executable by the user defined
# in fomd.cf and reside in the same directory on both the
# main and the spare SC.
```

```
# If the command is not part of the defined PATH for
# the user, the absolute filename must be passed with the
# initcmdsync command
#
initcmdsync script_name parameters
# The marker point is stored in the goto_label variable.
# Keep executing this script until all cases have been
# processed or an error is detected.
#
while (( $goto_label !=  0 )) ; do
#
# Each case should represent a synchronization point
# in the script.
#
case $goto_label in
#
# Step 1: Do something
#
1 )               do_something
                  .
                  .
                  .

# Execute the savecmdsync command with the script's
# descriptor and a unique marker to save the position.
# If a failover occurs here, the commands represented in
# the next goto_label (2) will be resumed.
#
           savecmdsync -M $(( $goto_label + 1 )) $desc
           goto_label=$(( $goto_label + 1 ))
             ;;
#
# Step 2: Do more things
#
2 )          do_more_things
             .
             .
             .
             savecmdsync -M $(( $goto_label + 1 )) $desc
               goto_label=$(( $goto_label + 1 ))
               ;;
#
# Step 3: Finish the last step and set the goto_label to 0
# so that the script ends.
3 )
             finish_last_step
             .
             .
             .
             goto_label=0
               ;;
 esac
done
# END OF MAIN CODE
# Remember to execute cancelcmdsync to remove the script from the
# command synchronization list. Otherwise, the command will be restarted
# after the failover.
#
cancelcmdsync $desc
```

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or domain configurator privileges to run this command.

Refer to Chapter 2 in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

Note – The standard output for `initcmdsync` contains the command synchronization descriptor. Also, when failover is disabled (after a failover or in a single SC environment), scripts that contain synchronization commands generate error messages to the platform log file and return nonzero exit codes. These messages can be ignored.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`runcmdsync(1M), showcmdsync(1M)`

NAME

codd - Capacity on Demand (COD) daemon

SYNOPSIS

codd

DESCRIPTION

`codd`(1M) is a process that runs on the main system controller (SC). This process does the following:

- Monitors the COD resources being used and verifies that the resources used are in agreement with the COD right-to-use (RTU) licenses in the COD license database file. Also logs any warning messages.
- Provides information on installed licenses, resource use, and board status.
- Handles the requests to add or delete COD RTU license keys.
- Configures headroom and COD RTU licenses reserved for domains.

This daemon is started automatically by the `ssd`(1M) daemon. If the `codd` daemon terminates, it is restarted automatically. Do *not* manually start this daemon from the command line.

EXTENDED DESCRIPTION

The `codd` daemon releases COD RTU licenses when the following events occur:

- A COD CPU board is powered off or disconnected from a running domain.
- A domain virtual keyswitch state changes from `on/secure` to `standby/off`

Clients of the `codd` daemon include:

`addcodlicense`(1M)	Adds a COD RTU license key to the COD license database on the system controller (SC)
`dxs`(1M)	Domain X server daemon
`deletecodlicense`(1M)	Removes a COD RTU license from the SC.
`hpost`(1M)	Sun Fire high-end systems power-on self-test (POST) control application
`setdefaults`(1M)	Remove all instances and reset reserved COD RTUs for a previously active domain.
`setupplatform`(1M)	Sets up the available component list and reserved COD RTUs for domains and configures platform COD headroom.
`showcodlicense`(1M)	Shows installed COD RTU licenses.
`showcodusage`(1M)	Shows current usage statistics for COD resources.
`showplatform`(1M)	Displays the board available component list, domain state, and reserved COD RTUs for each domain, and platform COD headroom.

The codd daemon is a client of

dsmd(1M)	Domain status monitoring daemon
frad(1M)	FRU access daemon
pcd(1M)	Platform configuration database daemon
setkeyswitch(1M)	Virtual keyswitch control command

FILES

The following file is supported:

/var/opt/SUNWSMS/adm/platform/messages	Stores message files.

ATTRIBUTES

See attributes(5) for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

addcodlicense(1M), deletecodlicense(1M), dsmd(1M), dxs(1M), frad(1M), hpost(1M), pcd(1M), setdefaults(1M), setkeyswitch(1M), setupplatform(1M), showcodlicense(1M), showcodusage(1M), showplatform(1M), ssd(1M)

NAME

console - access the domain console

SYNOPSIS

console -d *domain_indicator* [[-f]| [-l]| [-g]| [-r]] [-e *escapeChar*]

console -h

DESCRIPTION

console(1M) creates a remote connection to the domain virtual console driver, making the window in which the command is executed a console window for the specified domain (*domain_id* or *domain_tag*). Many console commands can be attached simultaneously to a domain, but only one console has write permissions; all others have read-only permissions. Write permissions are in either locked or unlocked mode.

If console is invoked without any options it comes up in exclusive locked write mode (option -f). An exclusive session forcibly detaches all other sessions from the domain virtual console.

Locked write mode is more secure. It can be taken away only if another console is opened using console -f or ~* is entered from another running console window. In both cases the new console session is an exclusive session.

Unlocked write permission is not as secure. It can be taken away if another console command is started using console -g, console -l, or console -f, or if ~@, ~&, or ~* is entered from another console window.

console can utilize either IOSRAM or the network path for domain console communication. You can manually toggle the communication path by using the ~= (tilde-equal sign) command. Doing so is useful if the network becomes inoperable, in which case the console sessions appears to be hung.

Tilde commands are described in EXTENDED DESCRIPTION.

OPTIONS The following options are supported:

`-d` *domain_indicator*
Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-e` *escapeChar*
Set default escape character. Changes the escape character to be `escapeCharacter`. The default is ~ (tilde).

Valid escape characters are any *except* the following:

@ ^ & ? * = . |

See the note on `rlogin` in the Usage section that follows.

`-f`
Force option (the default). Opens a domain console window with locked write permission, terminates all other open sessions, and prevents new ones from being opened. This constitutes an exclusive session. Use it only when you need exclusive use of the console (e.g. for private debugging).

Note – To restore multiple-session mode, either release the lock (`~^`) or terminate the console session (`~.`).

`-g`
Grab option. Opens a console window with unlocked write permission. If another session has unlocked write permission, that session becomes read-only. If another session has locked permission, this request is denied and the console window opens in read-only mode instead.

`-h`
Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-l`
Lock option. Opens a console window with locked write permission. If another session has unlocked write permission, that session becomes read-only. If another session has locked permission, the request is denied and the console window opens in read-only mode instead.

`-r`
Opens a console window in read-only mode.

EXTENDED DESCRIPTION

Usage

In a domain console window, a `tilde` (`~`) that appears as the first character of a line is interpreted as an escape signal that directs `console` to perform some special action, as follows:

`~?`	Status message.
`~.`	Disconnect `console` session.
`~#`	Break to OpenBoot PROM or `kadb`.
`~@`	Acquire unlocked write permission; see `-g`.
`~^`	Release write permission.
`~=`	Toggle the communication path between the network and `IOSRAM` interfaces. You can use `~=` only in Private mode (see `~*`).
`~&`	Acquire locked write permission; see `-l`. You can issue this signal during a read-only or unlocked write session.
`~*`	Acquire locked write permission, terminate all other open sessions, and prevent new sessions from being opened; see `-f`. To restore multiple-session mode, either release the lock or terminate this session.

Note – `rlogin` also processes `tilde-escape` sequences whenever a `tilde` is entered at the beginning of a new line. If you need to send a `tilde` sequence at the beginning of a line and you are using `rlogin`, use two `tilde`s (the first escapes the second for `rlogin`). Alternatively, do not enter a `tilde` at the beginning of a line when running inside of `rlogin`.

Note – If you use a `kill -9` command to terminate a console session, the window or terminal in which the console command was executed goes into raw mode and appears hung. To escape this condition, type `CTRL-j`, then `stty sane`, and then `CTRL-j`.

Group Privileges Required

You must have domain administrator privileges on the domain specified. Users with only platform group privileges are not allowed access to a domain console.

Refer to Chapter 2 in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Opening a Console Window in Locked Mode in Domain a

```
sc0:sms-user:> console -d a -l
```

Note – In the domain console window, `vi`(1) runs properly and the escape sequences (`tilde` commands) work as intended only if the environment variable TERM has the same setting as that of the console window. For example:

```
sc0:sms-user:> setenv TERM xterm
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes:

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `dxs(1M)`, `kill(1)`, `rlogin(1)`, `set(1)`, `stty(1)`, `vi(1)`, `xterm(1M)`

NAME

dca - domain configuration agent

SYNOPSIS

dca `-d` *domain_indicator* [`-H` *hostname*]

dca `-h`

DESCRIPTION

`dca`(1M) provides a communication mechanism between the `dca` on the system controller and the domain configuration server (`dcs`) on the specified domain. The `dca` provides communication services for remote dynamic reconfiguration commands.

This agent is automatically started by `ssd`(1M). Do *not* start it manually from the command line.

OPTIONS

The following options are supported:

`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-h`	Help. Displays usage description for the `dca` command. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-H` *hostname*	The Solaris software host name of the domain associated with the `dca`.

FILES

The following files are used by this command:

`/var/opt/SUNWSMS/doors/<domain_id>/dca`

`/var/opt/SUNWSMS/pipes/<domain_id>/scdr0`

`/var/opt/SUNWSMS/pipes/<domain_id>/scdr1`

`/var/opt/SUNWSMS/pipes/<domain_id>/scdr2`

`/var/opt/SUNWSMS/pipes/<domain_id>/scdr3`

`/var/opt/SUNWSMS/pipes/<domain_id>/scdr4`

`/var/opt/SUNWSMS/pipes/<domain_id>/scdr5`

`/var/opt/SUNWSMS/pipes/<domain_id>/scdr6`

`/var/opt/SUNWSMS/pipes/<domain_id>/scdr7`

/var/opt/SUNWSMS/pipes/<domain_id>/scdr8

/var/opt/SUNWSMS/pipes/<domain_id>/scdr9

ATTRIBUTES

See attributes(5) for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

addboard(1M), deleteboard(1M), moveboard(1M), rcfgadm(1M)

NAME

deleteboard - unconfigure, disconnect, and unassign a system board from a domain

SYNOPSIS

deleteboard [`-c` *function*] [`-r` *retry_count* [`-t` *timeout*]] [`-q`] [`-f`] [`-y`|`-n`] *location...*

deleteboard`-h`

DESCRIPTION

`deleteboard`(1M) removes a *location* from the domain it is currently assigned to and possibly active in. The board at that *location* must be in either the `assigned` or `connected`|`configured` states. The `-c` option is used to specify the transition of the board from the current configuration state to the new configuration state. However, the new firmware will not be active on the board until `hpost` is run. In addition, to complete a `deleteboard` operation, you must use the `disconnect` function.

Configuration states are `unconfigure`, `disconnect`, and `unassign`. If the `-c` option is not specified, the default expected configuration state is `unassign`.

A domain administrator can unconfigure and disconnect a board but cannot unassign a board from a domain unless the board is in the domain available component list. See `setupplatform`(1M). This means the `deleteboard` *location* field must appear in the domain available component list.

OPTIONS

The following options are supported.

Note – If the `deleteboard` command fails, a board does not return to its original state. A `dxs` or `dca` error message is logged to the domain. If the error is recoverable, you can retry the command. If it is unrecoverable, you need to reboot the domain in order to use that board.

`-c` *function*

Valid *function* values are `unconfigure`, `disconnect`, or `unassign`. The `-c` option is used to control the configuration state transition. Each successive function builds upon the last. For example, `unassign` first unconfigures and then disconnects the board before unassigning it.

The possible transition states and their meanings are as follows:

`unconfigure`	Unconfigures the board from the Solaris operating system running on the domain. Solaris software stops using any of the hardware resources on the board. Transitions the board into the `connected|unconfigured` state. In this state the system board is assigned to the logical domain and connected (remains in the `active` state). This state allows normal system access to hardware resources on the board, but the hardware resources of the board are not represented by the normal Solaris software data structures and thus are not available to the Solaris operating system. Operations allowed on the board are limited to configuration administration operations.
`disconnect`	Unconfigures the board from the Solaris operating system running on the domain. See `unconfigure`.Transitions the board into the `disconnected|unconfigured` state. Removes the board from the physical domain. An UNCLAIM request is sent by the domain to the SC during this step. In this state the system board is assigned to the logical domain and disconnected.
`unassign`	Unconfigures the board from the Solaris operating system running on the domain. See `unconfigure`.Disconnects the board. See `disconnect`. Moves the board out of the logical domain by changing its state to `available`.

`-f`	Forces the specified action to occur. Typically, this is a hardware-dependent override of a safety feature. Forcing a state change operation can allow use of the hardware resources of an occupant that is not in the `ok` or `unknown` conditions, at the discretion of any hardware-dependent safety checks.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-n`	Automatically answers no to all prompts. Prompts are displayed unless used with the `-q` option.
`-q`	Quiet. Suppresses all messages to `stdout` including prompts. When used alone `-q` defaults to the `-n` option for all prompts. When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts, and automatically answers with either 'y' or 'n' based on the option chosen.
`-r` *retry_count*	This command argument enables the user to specify retries in case of failures encountered during state transitions. The `-r` *retry_count* option can be used alone and indicates the number of times the configuration state change request should be retried by the domain.
`-t` *timeout*	This command argument enables the user to specify retries in case of failures encountered during state transitions. The `-t` *timeout* option cannot be used without the `-r` *retry_count* option and specifies the number of seconds that the domain should wait before the next retry is made. If the `-t` *timeout* is not specified, the default *timeout* is zero, meaning that the request is retried immediately.
`-y`	Automatically answers yes to all prompts. Prompts are displayed unless used with the `-q` option.

OPERANDS

The following operands are supported:

location — List of board locations separated by a space. Multiple *location* arguments are permitted.

The following *location* forms are accepted:

Sun Fire 12K and E20K:

```
SB(0...8)
IO(0...8)
```

Sun Fire 15K and E25K:

```
SB(0...17)
IO(0...17)
```

Note – Use `showboards`(1M) to display board type.

EXTENDED DESCRIPTION

Group Privileges Required

Users with platform administrator privileges can perform only the `-c unassign` option if the board(s) are in the `assigned` state (that is, not active in a running domain).

Users with domain administrator or configurator privileges can execute this command but only on their respective domains. To unassign a board, the board must be in the domain available component list.

Refer to Chapter 2 in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Unconfiguring Boards From a Domain

To unconfigure four boards from their domain, you must have domain administrator/configurator privileges and the boards must be in the domain available component list.

All boards are in the `configured` state in the example domain.

```
sc0:sms-user:> deleteboard  -c unconfigure SB0 IO1 SB1 SB2
```

EXAMPLE 2 Unassigning Boards From a Running Domain

To unassign three active boards from their domain, set retries to five and timeout to three seconds. The boards are unconfigured and disconnected before being unassigned. You must have domain administrator/configurator privileges, and the boards must be in the domain available component list.

```
sc0:sms-user:> deleteboard  -r5 -t3 IO3 IO4 IO5
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	No acknowledge.
2	Not supported.
3	Operation not supported.
4	Invalid privileges.
5	Busy.
6	System busy.
7	Data error.
8	Library error.
9	No library.
10	Insufficient condition.
11	Invalid.
12	Error.
13	A PID does not exist.
14	Invalid attribute.
30	Invalid board ID type.
31	Invalid permissions.
32	Assigned to another domain.
33	Unable to get permissions.
34	Unable to get domain board info.
35	Unable to get active board list.
36	Unable to get assigned board list.
38	Solaris not running.
39	Unable to assign/unassign domain state.
40	Unable to get domain permissions.

41 Unable to get platform permissions.

52 Invalid privileges

53 Internal error.

54 Library error.

56 DR command syntax error.

58 Internal error.

68 Location not assigned.

69 Location not configured.

70 DR operation failed.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addboard(1M)`, `moveboard(1M)`

NAME

deletecodlicense - remove a Capacity on Demand (COD) right-to-use (RTU) license key from the COD license database

SYNOPSIS

deletecodlicense [-f] *license-signature*

deletecodlicense-h

DESCRIPTION

The deletecodlicense(1M) command removes the specified COD RTU license key from the COD license database on the SC. For further information about COD RTU license keys, refer to the *SMS Administrator Guide*.

The system checks the number of COD RTU licenses against the number of COD CPUs in use. If the license removal will result in an insufficient number of COD RTU licenses with respect to the CPU in use, the system does not delete the license key from the COD RTU license database. If you still want to delete the COD RTU license key, you must reduce the number of COD CPUs in use. You can either power off the appropriate number of domains or use dynamic reconfiguration (DR) to disconnect the appropriate number of boards.

However, you can force the deletion of a COD RTU license by specifying the -f option, even if the license removal will result in a license violation.

OPTIONS

The following options are supported:

-f	Forces the specified COD RTU license key to be deleted from the COD license database.
-h	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to -h is ignored.
license-signature	Specifies the COD RTU license key to be deleted from the COD license database.

Group Privileges Required

You must have platform administrator group privileges to run this command.

Refer to Chapter 2 in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Deleting a COD RTU license key

```
sc0:sms-user:>deletecodlicense\
01:5014936C37048:01001:0201010302:4:20020430:jWGJdg/
kx78b0wyK2xrqIg
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`1`	Invalid usage
`2`	Invalid group privileges.
`>2`	An internal error occurred. For further information, see `/var/opt/SUNWSMS/adm/platform/messages`.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addcodlicense(1M)`, `codd(1M)`, `showcodlicense(1M)`, `showcodusage(1M)`

NAME

deletetag - remove the domain tag name associated with the domain

SYNOPSIS

deletetag `-d` *domain_indicator* [`-q`] [`-y`|`-n`]

deletetag `-h`

DESCRIPTION

`deletetag`(1M) removes the domain tag associated with the domain.

OPTIONS

The following options are supported:

`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-n`	Automatically answers no to all prompts. Prompts are displayed unless used with the `-q` option.
`-q`	Quiet. Suppresses all messages to `stdout` including prompts. When used alone, `-q` defaults to the `-n` option for all prompts. When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts and automatically answers with either 'y' or 'n' based on the option chosen.
`-y`	Automatically answers yes to all prompts. Prompts are displayed unless used with the `-q` option.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator privileges to run this command.

Refer to Chapter 2, "SMS Security Options and Administrative Privileges" in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Deleting Tag `eng2` From the Domain to Which It Was Assigned

```
sc0:sms-user:> deletetag -d eng2 -qy
```

You are not prompted.

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

If the *domain_id* does not have a tag, no error is returned. `deletetag`(1M) is treated as successful.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`

NAME

disablecomponent - add the specified component to the specified blacklist file

SYNOPSIS

disablecomponent [`-d` *domain_indicator*] [`-i` *"reason"*] *location...*

disablecomponent `-h`

DESCRIPTION

`disablecomponent`(1M) adds a component to the domain or platform blacklist, making it ineligible for booting.

The *blacklist* is an internal file that lists components that POST cannot use at boot time. POST reads the blacklist file(s) before preparing the system for booting, and passes along to OpenBoot PROM a list of only those components that have been successfully tested; those on the blacklist are excluded. SMS supports three blacklists: one for domain boards, one for platform boards, and the internal automatic system recovery (ASR) blacklist.

`disablecomponent`, when used without any option, edits the platform blacklist file.

`disablecomponent` *cannot* be used on the ASR blacklist file; only `enablecomponent`(1M) can be used to write to the ASR blacklist file.

For more information on the use and editing of platform and domain blacklists, refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

OPTIONS

The following options are supported:

`-d` *domain_indicator* — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-i`*reason* — Short, descriptive explanation for adding a component to the domain blacklist. Must be enclosed in either single or double quotation marks, or be a single word.

OPERANDS The following operands are supported:

location List of component locations, separated by forward slashes and comprising the following:

board_loc/proc/bank/logical_bank

board_loc/proc/bank/all_dimms_on_that_bank

board_loc/proc/all_banks_on_that_proc

board_loc/all_banks_on_that_board

board_loc/proc

board_loc/procs

board_loc/cassette

board_loc/bus

board_loc/paroli_link

Multiple *location* arguments are permitted, separated by a space.

The *location* forms are optional and are used to specify particular components on boards in specific locations.

For example, the *location* SB5/P0/B1/L1 indicates Logical Bank 1 of Bank 1 on Processor 0 at SB5.

The SB0/PP1 *location* indicates Processor Pair 1 at SB0.

The CS0/ABUS *location* indicates address bus 0 on the centerplane.

The EX11/DBUS1 *location* indicates data bus 1 at expander 11.

The following *board_loc* forms are accepted:

Sun Fire 12K and E20K:

```
SB(0...8)

IO(0...8)
```

Sun Fire 15K and E25K:

```
SB(0...17)

IO(0...17)
```

Processor locations indicate single processors or processor pairs.

There are four possible processors on a CPU/memory board. Processor pairs on that board are procs 0 and 1, and procs 2 and 3.

Note – If you blacklist a processor cpu0 in procpair0, its CPU partner, cpu1, is also removed from the domain configuration, and neither processor is used. If cpu1 is disabled, cpu0 remains in the domain configuration. If either cpu2 or cpu3 in procpair1 is disabled, the other CPU in the pair remains unaffected.

The MaxCPU has two processors, procs 0 and 1, and only one proc pair (PP0). Using PP1 for this board causes `disablecomponent` to exit and display an error message.

The following *proc* forms are accepted:

```
P(0...3)    PP(0|1)
```

The following *bank* forms are accepted:

```
B(0|1)
```

The following *logical_bank* forms are accepted:

```
L(0|1)
```

The following *all_dimms_on_that_bank* form is accepted:

```
D
```

The following *all_banks_on_that_proc* form is accepted:

```
B
```

The following *all_banks_on_that_board* form is accepted:

```
B
```

The following *paroli_link* forms are accepted:

```
PAR(0|1)
```

The hsPCI assemblies contain hot-swappable cassettes.

The following *hsPCI* forms are accepted:

`C(3|5)V(0|1)`

The hsPCI+ assemblies contain hot-swappable cassettes.

The following *hsPCI+* forms are accepted:

`C3V(0|1|2) and C5V0`

There are three bus locations: address, data, and response.

The following *bus* forms are accepted:

`ABUS|DBUS|RBUS (0|1)`

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, domain administrator, or domain configurator privileges to run this command. If you have platform privileges, you can run this command for the platform components only. If you have domain privileges you can run this command only on the domain for which you have privileges.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Adding CSB 0 and Processor 2 on System Board 1 to the Domain A Blacklist

```
sc0:sms-user:> disablecomponent -dA CS0 SB1/P2
```

EXAMPLE 2 Adding the Logical Bank 0 of Bank 0 on Processor 0 on System Board 0 to the Domain A Blacklist

```
sc0:sms-user:> disablecomponent -dA SB0/P0/B0/L0
```

EXAMPLE 3 Adding All DIMMs on Bank 0 of Processor 1 on System Board 3 to the Domain A Blacklist

```
sc0:sms-user:> disablecomponent -dA SB3/P1/B0/D
```

EXAMPLE 4 Adding All Banks on Processor 0 on System Board 1 to the Domain B Blacklist

```
sc0:sms-user:> disablecomponent -dB SB1/P0/B
```

EXAMPLE 5 Adding All Banks on System Board 0 to the Domain D Blacklist

```
sc0:sms-user:> disablecomponent -dD SB0/B
```

EXAMPLE 6 Adding Processor Pair 1 on System Board 3 to the Platform Blacklist

```
sc0:sms-user:> disablecomponent SB3/PP1
```

EXAMPLE 7 Adding the hsPCI Cassette in the 5V slot 0 of IO Board 6 to the Domain A Blacklist

```
sc0:sms-user:> disablecomponent -dA IO6/C5V0
```

EXAMPLE 8 Adding Paroli Link 0 on wPCI Board 7 to the Platform Blacklist

```
sc0:sms-user:> disablecomponent  IO7/PAR0
```

EXAMPLE 9 Adding the Data Bus CS0 on EX9 to the Domain A Blacklist

```
sc0:sms-user:> disablecomponent -dA EX9/DBUS0
```

EXAMPLE 10 Adding CSB 0 and Processor 2 on System Board 1 to the Domain A Blacklist Because It Is Scheduled to Be Upgraded

```
sc0:sms-user:> disablecomponent -dA -i upgrade CS0 SB1/P2
```

EXAMPLE 11 Adding Processor Pair 1 on System Board 3 to the Platform Blacklist Because It Needs Service

```
sc0:sms-user:> disablecomponent -i "Needs service" SB3/PP1
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

FILES

The following files are used by this command.

`/etc/opt/SUNWSMS/config/platform/blacklist`	List of platform components excluded.
`/etc/opt/SUNWSMS/config/`*domain_id*`/blacklist`	List of domain components to be excluded.

ATTRIBUTES

See `attributes(5)` for a description of the following attribute.

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO

`addboard(1M)`, `enablecomponent(1M)`, `esmd(1M)`, `showcomponent(1M)`

NAME

dsmd - domain status monitoring daemon

SYNOPSIS

dsmd

DESCRIPTION

`dsmd`(1M) monitors domain status and operating system (OS) heartbeat for up to 18 domains on Sun Fire 15K/E25K systems, and up to nine domains on Sun Fire 12K/E20K systems.

`dsmd` automatically recovers the domain and handles domain-related hardware errors. In the event of a domain hang, `dsmd` resets the domain, collects CPU registers and hardware configuration dumps, and saves them to two files. This daemon also passes to `efhd`(1M) any automatic diagnosis (AD) information related to a domain stop.

All domain state changes are monitored and logged in domain-specific log files if the message level is INFO; otherwise there is no log for a state change.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

EXTENDED DESCRIPTION

`dsmd` logs the following events and attempts to recover from them:

- Domain boot failure
- Error reset
- Solaris OS hang
- Domain panic
- Domain reset/reboot
- DStop
- Boot/panic/error_reset_sync timeout

`dsmd` clients include:

`codd`(1M)	Capacity on Demand (COD) daemon
`dxs`(1M)	Domain *X* server daemon
`efe`	Event Front-end daemon
`osd`(1M)	OpenBoot PROM daemon
`pcd`(1M)	Platform configuration database daemon
`esmd`(1M)	Environment status monitoring daemon

`dsmd` is a client of:

`hwad`(1M)	Hardware access daemon
`setkeyswitch`(1M)	Virtual keyswitch control command

For more information refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

FILES

The following files are supported:

`/etc/opt/SUNWSMS/startup/ssd_start`	Default startup file for `ssd`.
`/var/opt/SUNWSMS/adm/`*domain_id*`/`	Stores message files and `hpost` dump files.
/var/opt/SUNWSMS/SMS/adm/*domain_id*/post/	Stores the `dstop` and hardware configuration dump files
`/export/home/`*sms-user*`/xir_dump/`	Stores `xir` dump files for all domains.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`codd(1M)`, `dxs(1M)`, `efhd(1M)`, `esmd(1M)`, `hwad(1M)`, `osd(1M)`, `pcd(1M)`, `reset(1M)`, `setkeyswitch(1M)`, `ssd(1M)`

NAME

dxs - domain X server

SYNOPSIS

dxs [-S] -d *domain_indicator*

dxs -h

DESCRIPTION

dxs(1M) provides software support for a domain. This support includes virtual console functionality, dynamic reconfiguration mailbox support, and PCI mailbox support. The mailbox support handles domain driver requests and events. The virtual console functionality enables one or more users running the console program to access the domain's virtual console.

When the domain is running Solaris software, dxs acts as a relay between the domain's console driver (cvcd) and the running console windows. When the domain is not running Solaris software, dxs acts as a relay between OpenBoot PROM and the running console windows.

A domain X server is automatically started for each active domain by the ssd(1M) daemon. Do *not* start it manually from the command line. dxs for the domain is terminated when the domain is shut down.

OPTIONS

The following options are supported:

-d *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using addtag(1M).
-h	Help. Displays usage descriptions for the specified *hostname*. **Note –** Use alone. Any option specified in addition to -h is ignored.
-S	Disables console output logging. By default, logging is enabled and is written to the /var/opt/SUNWSMS/adm/*domain_id*/console file.

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `console(1M)`, `ssd(1M)`

NAME

efhd - error and fault handling daemon

SYNOPSIS

efhd

DESCRIPTION

`efhd`(1M) performs automatic error diagnosis based on error information passed by `dsmd`(1M). `efhd` also updates the health status of components associated with a hardware failure, based on the list event that captures the diagnosis performed by the automatic diagnosis (AD) engine, POST, or the Solaris operating system on domains. `efhd` passes this list event to `erd`(1M) for reporting.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

EXTENDED DESCRIPTION

`efhd`(1M) does the following:

- Receives the diagnosis results performed by the other diagnosis engines in the system, POST, and the domain Solaris operating system
- Analyzes errors captured in dstop files
- Analyzes errors in the data path
- Analyzes non-fatal CPU errors
- Analyzes ECC errors
- Generates the error information used by the automatic diagnosis engine to determine components associated with the encountered errors
- Updates the component health status of the components associated with the encountered errors
- Passes the diagnosis results to `erd`(1M) for reporting

For more information, refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

FILES

The following files are supported:

`/etc/opt/SUNWSMS/startup/ssd_start`	Default startup file for `ssd`
`/etc/opt/SUNWSMS/SMS/config/efhd_rules.cf`	Contains specialized diagnosis rules
`/etc/opt/SUNWSMS/SMS/config/efhd.cf`	Contains `efhd` tunables
`/etc/opt/SUNWSMS/SMS/config/SF12000.dict`	Contains primary and known permutations of fault classes

`/etc/opt/SUNWSMS/SMS/config/SF15000.dict`	Contains primary and known permutations of fault classes
`/etc/opt/SUNWSMS/SMS/config/E20000.dict`	Contains primary and known permutations of fault classes
`/etc/opt/SUNWSMS/SMS/config/E25000.dict`	Contains primary and known permutations of fault classes

ATTRIBUTES

See `attributes(5)` for a description of the following attribute:

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO

`dsmd(1M)`, `erd(1M)`, `ssd(1M)`

NAME

elad - event log access daemon

SYNOPSIS

elad

DESCRIPTION

`elad`(1M) controls access to the SMS event log (`eventlog`), which records fault and error events identified by the automatic diagnosis (AD) engine on a Sun Fire high-end system. This daemon also performs the following archive tasks:

- Starts a new event log file whenever the current event log reaches its size limit.

  The current log file is archived as `eventlog.0`. Whenever a new event log is created, the file names of existing archive logs are incremented by 1. A maximum of ten archive files (`eventlog.0` through `eventlog.9`) is maintained.
- Deletes the oldest archive file, `eventlog.9`, whenever a new event log file is created.
- Passes error and list events to `elad`(1M) for recording.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

FILES

The following files are supported:

`/etc/opt/SUNWSMS/SMS/config/elad_tuning.txt`	Sets the size, number of archive files to be maintained, and number of days that the archive files are to be retained.
`/etc/opt/SUNWSMS/startup/ssd_start`	Default startup file for `ssd`.
`/var/opt/SUNWSMS/adm/`*`domain_id`*`/`	Stores message files and `hpost` dump files.
`/var/opt/SUNWSMS/SMS/adm/events/eventlog`	Stores all the hardware-related error and fault events

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`efhd(1M)`, `elad(1M)`, `erd(1M)`, `mld(1M)`, `ssd(1M)`

NAME

enablecomponent - remove the specified component from the specified blacklist

SYNOPSIS

enablecomponent [`-a|-d` *domain_indicator*] *location...*

enablecomponent `-h`

DESCRIPTION

`enablecomponent`(1M) removes a component from the platform, domain, or ASR blacklist, making it eligible for booting.

The *blacklist* is an internal file that lists components that POST cannot use at boot time. POST reads the blacklist file(s) before preparing the system for booting, and passes along to OpenBoot PROM a list of only those components that have been successfully tested; those on the blacklist are excluded. SMS supports three blacklists: one for domain boards, one for platform boards, and the internal ASR blacklist.

The ASR *blacklist* is an internal file created by `esmd` when it powers off components due to environmental conditions. The ASR blacklist is also used by the power libraries and SMS commands to prevent turning on a bad component.

`enablecomponent`, when used without any option, edits the platform blacklist.

Use `showcomponent`(1M) to display whether a particular component is currently blacklisted.

For more information on the use and editing of platform and domain blacklists refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

OPTIONS

The following options are supported:

`-a` — Specifies the component named in *location* will be removed from the ASR blacklist.

`-d` *domain_indicator* — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case insensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

OPERANDS The following operands are supported:

location List of component locations, separated by forward slashes and comprised of:

board_loc/proc/bank/logical_bank

board_loc/proc/bank/all_dimms_on_that_bank

board_loc/proc/all_banks_on_that_proc

board_loc/all_banks_on_that_board

board_loc/proc

board_loc/procs

board_loc/cassette

board_loc/bus

board_loc/paroli_link

Multiple *location* arguments are permitted, separated by a space.

The *location* forms are optional and are used to specify particular components on boards in specific locations.

For example, the *location* SB5/P0/B1/L1 indicates Logical Bank 1 of Bank 1 on Processor 0 at SB5.

The SB0/PP1 *location* indicates Processor Pair 1 at SB0.

The CS0/ABUS *location* indicates address bus 0 on the centerplane.

The EX11/DBUS1 *location* indicates data bus 1 at expander 11.

The following *board_loc* forms are accepted:

Sun Fire 12K and E20K:

```
SB(0...8)
IO(0...8)
```

Sun Fire 15K and E25K:

```
SB(0...17)
IO(0...17)
```

Processor locations indicate single processors or processor pairs.

There are four possible processors on a CPU/memory board. Processor pairs on that board are: procs 0 and 1, and procs 2 and 3.

The MaxCPU has two processors, procs 0 and 1, and only one proc pair (PP0). Using PP1 for this board causes `enablecomponent` to exit and display an error message.

The following *proc* forms are accepted:

```
P(0...3)    PP(0|1)
```

The following *bank* forms are accepted:

```
B(0|1)
```

The following *logical_bank* forms are accepted:

```
L(0|1)
```

The following *all_dimms_on_that_bank* forms are accepted:

```
D
```

The following *all_banks_on_that_proc* forms are accepted:

```
B
```

The following *all_banks_on_that_board* forms are accepted:

```
B
```

The following *paroli_link* forms are accepted:

```
PAR(0|1)
```

The hsPCI assemblies contain hot-swappable cassettes.

The following *hsPCI* forms are accepted:

`C(3|5)V(0|1)`

The hsPCI+ assemblies contain hot-swappable cassettes.

The following *hsPCI+* forms are accepted:

`C3V(0|1|2) and C5V0`

There are three bus locations: address, data, and response.

The following *bus* forms are accepted:

`ABUS|DBUS|RBUS (0|1)`

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, domain administrator, or domain configurator privileges to run this command. If you have platform privileges, you can run this command for the platform components only. If you have domain privileges, you can run this command only on the domain for which you have privileges.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Removing CSB0 From the ASR Blacklist

```
sc0:sms-user:> enablecomponent -a CS0
```

EXAMPLE 2 Removing the Logical Bank 0 of Bank 0 on Processor 0 on System Board 0 From the Domain A Blacklist

```
sc0:sms-user:> enablecomponent -dA SB0/P0/B0/L0
```

EXAMPLE 3 Removing All DIMMs on Bank 0 of Processor 1 on System Board 3 From the Domain A Blacklist

```
sc0:sms-user:> enablecomponent -dA SB3/P1/B0/D
```

EXAMPLE 4 Removing All Banks on Processor 0 on System Board 1 From the Domain B Blacklist

```
sc0:sms-user:> enablecomponent -dB SB1/P0/B
```

EXAMPLE 5 Removing All Banks on System Board 0 From the Domain D Blacklist

```
sc0:sms-user:> enablecomponent -dD SB0/B
```

EXAMPLE 6 Removing Processor Pair 0 on I/O Board 7 From the Platform Blacklist

```
sc0:sms-user:> enablecomponent IO7/PP0
```

EXAMPLE 7 Removing Processor 1 on System Board 3 From the Domain A Blacklist

```
sc0:sms-user:> enablecomponent -dA SB3/P1
```

EXAMPLE 8 Removing the hsPCI Cassette in the 3V slot 0 of IO Board 6 From the Domain A Blacklist

```
sc0:sms-user:> enablecomponent -dA IO6/C3V0
```

EXAMPLE 9 Removing the Paroli Link 0 on wPCI Board 5 From the Platform Blacklist

```
sc0:sms-user:> enablecomponent  IO5/PAR0
```

EXAMPLE 10 Removing the Address Bus CS0 on EX7 From the Domain A Blacklist

```
sc0:sms-user:> enablecomponent -dA EX7/ABUS0
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following files are used by this command.

`/etc/opt/SUNWSMS/config/asr/blacklist`	List of components excluded by `esmd`.
`/etc/opt/SUNWSMS/config/platform/blacklist`	List of platform components excluded.
`/etc/opt/SUNWSMS/config/`*domain_id*`/blacklist`	List of domain components excluded.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO | `addboard(1M)`, `disablecomponent(1M)`, `esmd(1M)`, `showcomponent(1M)`

NAME

erd - event reporting daemon

SYNOPSIS

erd

DESCRIPTION

`erd`(1M) provides reporting services that deliver fault event text messages to platform and domain logs, information for Sun Management Center and Sun Remote Services, and email reports that contain fault event messages.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

FILES

The following files are supported:

`/etc/opt/SUNWSMS/SMS/config/event_email.cf`	Controls email notifications
`/etc/opt/SUNWSMS/SMS/config/templates/sample_email`	Default format of event content in email
`/etc/opt/SUNWSMS/SMS/config/templates/sendmail.sh`	Default shell script to send email
`/etc/opt/SUNWSMS/startup/ssd_start`	Default startup file for `ssd`
`/var/opt/SUNWSMS/SMS/adm/platform/events/eventlog`	Stores all the hardware-related error and fault events

ATTRIBUTES

See `attributes(5)` for a description of the following attribute:

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO

`mld(1M)`, `ssd(1M)`, `testemail(1M)`

NAME

esmd - environmental status monitoring daemon

SYNOPSIS

esmd

DESCRIPTION

`esmd`(1M) monitors system cabinet environmental conditions —for example, voltage, temperature, fan tray, and power supply. `esmd` logs abnormal conditions and takes action, if necessary, to protect the hardware.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

EXTENDED DESCRIPTION

`esmd` monitors the following boards for out-of-range conditions:

- CPU board
- MaxCPU board
- HPCI board
- HPCI+ board
- Expander board
- Centerplane support board
- SC control board
- SC I/O
- Fan control board
- Power supplies (bulk)

`esmd` recognizes the following events and alerts the appropriate clients/daemons:

Component insertion	Notices component presence from one polling cycle to the next. `esmd` sends notification only if that client (`hwad`(1M), `pcd`(1M), `dsmd`(1M), and so on) has requested it for that particular component type.
Component removal	Notices component absence from one polling cycle to the next. `esmd` sends notification only if that client (`hwad`, `pcd`, and so on) has requested it for that particular component type.
PCI card insertion	Notices whenever a PCI card has been inserted into a PCI board.
PCI card removal	Notices whenever a PCI card has been removed from a PCI board.
Board power off	Notices whenever a board is powered off or when board power, previously on, is off.
Board power on	Notices when a board is powered on or when board power, previously off, is on.

Board temperature change	Notices when temperature sensors on a board register a two-degree difference or when a temperature crosses a temperature threshold.
Board voltage change	Notices if a voltage sensor value has changed so that it is close to being out of range and again if the new value is out of range. In this case `esmd` removes the board from the domain and powers it off.
Board current change	Notices whenever any of the monitored board current values change.
CSB state change	Notices when any of the monitored CSB board properties change.
EXB state change	Notices when any of the monitored EXB board properties change.
CPU state change	Notices when any of the monitored CPU board properties change.
Bulk power state change	Notices when any of the bulk power supply properties change.
Fan tray state change	Notices when any of the monitored fan tray properties change.
PCI card state change	Notices when any of the monitored PCI card properties change.

`esmd` clients include:

`hwad`	Hardware access daemon
`pcd`	Platform configuration database daemon
`dsmd`	Domain status monitoring daemon

For more information refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following file is supported.

`/var/opt/SUNWSMS/adm/platform/messages`	Stores message files.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`dsmd(1M)`, `hwad(1M)`, `pcd(1M)`, `ssd(1M)`

NAME

flashupdate - update the flash PROMs located on the CPU boards, MaxCPU boards, and system controllers (SCs)

SYNOPSIS

flashupdate `-d` *domain_indicator* `-f` *path* [`-q`|`-v`] [`-y`|`-n`]

flashupdate `-f` *path* [`-q`|`-v`] [`-y`|`-n`] *location*...

flashupdate `-h`

DESCRIPTION

`flashupdate`(1M) updates the flash PROMs (FPROMs) in the system controllers (SC), and the FPROMs in a domain's CPU and MaxCPU boards, given the board location.

To update the FPROMs in the system controller, you must log in to the SC you want to update, and specify the FPROM to be updated. (You cannot update one SC from the other SC.) Each FPROM has a specific image file associated with it:

FPROMs in CP1500-based SCs use the nSSCPOST.di image file.

FROMS in CP2140-based SCs use the oSSCPOST.di image file.

Once you have finished updating the SC FPROMs, you must shut down and reset the SC. See Example 6 in the EXAMPLES section. You do not need to reset the SC after updating CPU FPROMs.

Before you can update the CPU FPROMs, SMS must be running and the specified board must be powered on. This is not required for updating the SC FPROMs. If any of the domain's CPU or MaxCPU boards have the virtual keyswitch set to the `secure` position, the FPROM(s) are not updated.

`flashupdate` displays both the current FPROM and the flash image file information prior to any updates. You are prompted to update the FPROMs.

Note – No CLIs should be executed on a system board while `flashupdate` is running on that board. Wait until `flashupdate` completes before running any SMS commands involving that system board.

Note – The new firmware will not be active on system boards until `hpost` is run.

OPTIONS The following options are supported:

`-d` *domain_indicator* — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-f` *path* — Name of the flash image file.

The *path* argument specifies the name of the image file that is used to update the FPROM given in the *location* argument.

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-n` — Automatically answers no to all prompts. Prompts are displayed unless used with the `-q` option.

`-q` — Quiet. Suppresses all messages to `stdout` including prompts.

When used alone, `-q` defaults to the `-n` option for all prompts.

When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts and automatically answers with either 'y' or 'n' based on the option chosen.

`-v` — Verbose. Displays version information about the firmware. The information displayed is intended for internal use by Sun service personnel. Its format is private and subject to change.

`-y` — Automatically answers yes to all prompts. Prompts are displayed unless used with the `-q` option.

OPERANDS

The following operands are supported:

location — FPROM location.

The FPROM location consists of *board_loc/FPROM_id*, separated by a forward slash.

The *FPROM_id* is specified only when you want to update a particular FPROM (FP0 or FP1) on a CPU board and the system controller (SC).

For example, the location, SB4/FP0, indicates the FPROM 0 on the CPU board in slot 4.

Sun Fire 12K and E20K:

```
SB(0...8)

IO(0...8)
```

Sun Fire 15K and E25K:

```
SB(0...17)

IO(0...17)
```

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator or domain administrator privileges to run this command. If you have platform administrator privileges, all boards can be updated. For domain administrators, only boards that are active in the administrator's domain or available to the administrator's domain can be updated.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Updating FPROM 0 in System Controller 0

You must reset the SC after running this command.

```
sc0:sms-user:> flashupdate -f /opt/SUNWSMS/firmware/SCOBPimg.di
SC0/FP0
```

EXAMPLE 2 Updating FPROM 1 in System Controller 0

The first instance, which uses the nSSCPOST.di image, is for an SC with a CP1500 board. The second instance, which uses the oSSCPOST.di image, is for an SC with a CP2140 board. You must reset the SC after running this command.

```
sc0:sms-user:> flashupdate -f /opt/SUNWSMS/firmware/nSSCPOST.di
SC0/FP1

sc0:sms-user:> flashupdate -f /opt/SUNWSMS/firmware/oSSCPOST.di
SC0/FP1
```

EXAMPLE 3 Updating FPROM 0 in the System Controller 1

You must reset the SC after running this command.

```
sc1:sms-user:> flashupdate -f /opt/SUNWSMS/firmware/SCOBPimg.di
SC1/FP0
```

EXAMPLE 4 Updating Both FPROMs on CPU Board 0

SMS must be running, and the SB0 board must be powered on.

```
sc0:sms-user:> flashupdate -f /opt/SUNWSMS/hostobjs/sgcpu.flash SB0
Do you wish to update the FPROM (yes/no)? y
sc0:sms-user:>
```

EXAMPLE 5 Updating FPROMs in Domain A

SMS must be running, and the CPU boards in domain A must be powered on.

```
sc0:sms-user:> flashupdate -d A -f /opt/SUNWSMS/hostobjs/sgcpu.flash
```

EXAMPLE 6 Resetting the SC After Updating the SC FPROMs

Switch to superuser and shut down the SC.

```
sc0:sms-user:> su -


sc0:#  shutdown -y -g0 -i0
 ...[system messages]
ok
```

In order for the new firmware to be enabled on your SC, you must now perform a reset of the SC. Type:

```
ok reset-all
```

If this does not reset the SC, then you must perform a hard reset.

Physically locate your system controller within your Sun Fire high-end system cabinet, and depress the Abort button and then the Reset button on the SC board. Once the SC has been reset you should see OpenBoot PROM messages indicating that the new version of the firmware is loading. After the system successfully returns to the `ok` prompt, verify that the flash update worked by typing:

```
ok show-dropins


Dropins for Flash device: /pci@1f,0/pci@1,1/ebus@1/flashprom@10,400000
---------------------------------------------------------------------------
Dropin name     Size  Checksum  Date      Date     Version  Vendor
                                created   flashed
---------------------------------------------------------------------------
SSCOBP-dropins    90    c84e   11/13/2001  11/13/2001   1.2  SUNW,sscobp
....
....
```

Note the version number of the Dropins (`1.2`).

You can now boot your new installation of the Solaris software. Type:

```
ok boot new disk
```

Log in as a platform administrate and type:

```
sc0:sms-user:> flashupdate -f /opt/SUNWSMS/firmware/SCOBPimg.di
SC0/FP0

Current SC FPROM Information
============================
SC at SC0, FPROM 0:
Name: SSCOBP-dropins,
Version: 1.2
Size: 144,
Check Sum: 51278
Date Flashed: 11/13/01
Date Created: 11/13/01

Do you wish to update the SC User FPROM (yes/no)? n

sc0:sms-user:>
```

Compare version numbers. If they are the same, `flashupdate` was successful.

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following files are used by this command:

`/opt/SUNWSMS/firmware/SCOBPimg.di`	Used to update the FPROM 0 on the SC.
`/opt/SUNWSMS/firmware/nSSCPOST.di`	Used to update the FPROM 1 on the CP1500 SC.
`/opt/SUNWSMS/firmware/oSSCPOST.di`	Used to update the FPROM1 on the CP2140 SC.
`/opt/SUNWSMS/hostobjs/sgcpu.flash`	Used to update the FPROMs on the CPU and MaxCPU boards.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Interface stability	Evolving
Availability	SUNWSMSop

SEE ALSO

`setkeyswitch(1M)`

NAME

fomd - failover management daemon

SYNOPSIS

fomd

DESCRIPTION

`fomd`(1M) is the core of the system controller (SC) failover mechanism. The `fomd` daemon detects faults on the local and remote SCs and takes the appropriate action (directing a failover/takeover).

The `fomd` daemon ensures that the necessary synchronization data between the two SCs is current. `fomd` runs on both the main and the spare SCs.

This daemon is automatically started by `ssd`(1M). Do *not* start it manually from the command line.

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

FILES

The following configuration file is required:

`/etc/opt/SUNWSMS/config/fomd.cf` Failover daemon configuration file

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`setfailover(1M)`, `showfailover(1M)`

NAME

frad - FRU access daemon

SYNOPSIS

frad

DESCRIPTION

`frad`(1M) runs on the system controller (SC) and provides the exclusive mechanism by which SMS processes, including daemons, access any FRU SEEPROM within a given Sun Fire high-end system. `frad` also provides the platform-dependent interface to the Sun Fire high-end systems SEEPROMs required by the FRU ID software tools.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`ssd(1M)`

NAME

help - display help information for SMS commands

SYNOPSIS

help [*command_name*]

help -h

DESCRIPTION

If no argument is included, `help`(1M) displays a list of valid SMS commands along with their correct syntax. Otherwise, the *command_name* operand displays that command's man page.

OPTIONS

The following options are supported:.

`-h` Help. Displays usage descriptions.

OPERANDS

The following operands are supported:

command_name Specific command for which `help` displays the man page.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, domain configurator, or superuser privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Using Help

Displays all commands.

```
sc0:sms-user:> help
usage:
addboard -d domain_indicator [ -c  function] [-r  retry_count [-t timeout]]
  [-q ] [-y | -n] location...
addboard -h
....
....
....
smsversion -h
```

EXAMPLE 2 Using Help for a Command

Displays man(1M) page.

```
sc0:sms-user:> help addtag

Maintenance Commands                                addtag(1M)

NAME
 addtag - assign a domain name (tag) to a domain

SYNOPSIS

addtag -d domain_indicator -a new_tag [-q ] [-y | -n]
.....
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

ATTRIBUTES

See attributes(5) for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

man(1M)

NAME

hpost - Sun Fire high-end system power-on self-test (POST) control application

SYNOPSIS

hpost

DESCRIPTION

`hpost`(1M) is responsible for probing, testing, and configuring the hardware of a Sun Fire high-end system domain, preparing it for use by the OpenBoot PROM and the Solaris operating system. Alternate modes prepare a single board for `attach` to a running domain using dynamic reconfiguration (DR), create hardware state dump files on the system controller (SC), clear certain nonfatal hardware error states, and perform related Sun Fire high-end system hardware operations.

Note – This application is intended to be run only by other SMS applications or daemons. Invoking it directly from the command line can cause failures of running domains and is *not* a supported mode of use.

`hpost`'s clients include:

- `dsmd`(1M)
- `dxs`(1M)
- `setkeyswitch`(1M)

`hpost` is a client of:

- `hwad`(1M)
- `pcd`(1M)

`hpost` requires and uses flash PROM images and downloadable local POST executables delivered in the `SUNWSMSlp` package.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSpo

SEE ALSO

`dsmd(1m)`, `hwad(1m)`, `pcd(1m)`, `setkeyswitch(1m)`, `dxs(1m)`

NAME

hwad - hardware access daemon

SYNOPSIS

hwad

DESCRIPTION

`hwad`(1M) provides the exclusive mechanism by which SMS processes, including daemons, access, control, monitor, and configure the hardware.

`hwad` runs in either main or spare mode and asks the failover daemon (`fomd`(1M)) for the role that the system controller (SC) should play when it comes up.

At start-up, `hwad` opens all the drivers (`sbbc`, `echip`, `gchip`, and `console bus`) and uses the `ioctl` calls to interface with them. It reads the contents of the device presence register to identify the boards present in the system, and makes them accessible to the clients.

`IOSRAM` and `Mbox` interfaces are also provided by `hwad`. This helps communication between the SC and the domain. For dynamic reconfiguration (DR), `hwad` directs communication to the new `IOSRAM` (tunnel switch). For `darb` interrupts, `hwad` notifies the `dsmd`(1M) if there is a `dstop` or `rstop`. It also notifies related SMS daemon(s), depending on the type of `Mbox` interrupt that occurs.

`hwad` detects and recovers `console` bus and `jtag` errors.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`dsmd(1M)`, `ssd(1M)`

NAME

initcmdsync - command synchronization command

SYNOPSIS

cancelcmdsync *cmdsync_descriptor*

initcmdsync *script_name* [*parameters*]

savecmdsync `-M` *identifier cmdsync_descriptor*

[cancel | init | save]cmdsync `-h`

DESCRIPTION

The command synchronization commands work together to control the recovery of user-defined scripts interrupted by a system controller (SC) failover. Insert the following commands in user-defined scripts to enable command synchronization:

`initcmdsync` Creates a command synchronization descriptor that identifies the script to be recovered.

This descriptor is placed on a command synchronization list that identifies the scripts and commands to be restarted on the new main SC after a failover.

`savecmdsync` Adds a marker that identifies a location in the script from which processing can be resumed after a failover.

`cancelcmdsync` Removes a command synchronization descriptor from the command synchronization list. This ensures that the script is run only once and not after subsequent failovers.

Be sure that all exit paths of a script have a `cancelcmdsync` sequence to remove the descriptor from the command synchronization list. If you do not remove the descriptor and a failover occurs, the script is rerun on the new main SC.

Note – Both an `initcmdsync` and a `cancelcmdsync` sequence must be contained within a script to enable command synchronization. The use of the `savecmdsync` command is optional and marks only specific points in a script from which processing can be resumed. If specific restart points are not necessary, consider using `runcmdsync`(1M) instead.

OPTIONS

The following options are supported:

cmdsync_descriptor	Specifies the command synchronization descriptor that identifies the user-defined script. This descriptor is the standard output value returned by the `initcmdsync` command.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-M` *identifier*	Marks a location in the script from which the script can be resumed after a failover. The identifier must be a positive integer.
parameters	Specifies the options or parameters associated with the user-defined script. These parameters are stored on the spare SC and are used to restart the specified script after a failover.
script_name	Identifies the name of the user-defined script to be synchronized. *script_name* must be the absolute path name of an executable command. The command must exist in the same location on both SCs.

EXTENDED DESCRIPTION

The command synchronization commands are inserted at certain logical points within a user-defined script.

For instance, a Korn shell script might be structured as follows:

```
# MAIN CODE STARTS HERE
# Be sure to use a cleanup procedure to handle any
# interrupts.
# Use the cancelcmdsync to remove the script from the
# command synchronization list. Otherwise, the command
# will get restarted on the new main SC.
#
clean_up () {
        cancelcmdsync  $desc
        exit
    }

# Declare the clean_up function to capture system signals
# and cleanup.
trap "clean_up" INT HUP TERM QUIT PWR URG
goto_label=1
# Process the arguments, capturing the -M marker point
# if provided
#
for arg in $*; do
       case $arg in
            -M )
       goto_label=$arg;;
       .
       .
       .
       esac
   done
# Place this script and all its parameters in the command
# synchronization list, which indicates the commands to
# be restarted after an SC failover.
#
# NOTE: The script must be executable by the user defined
# in fomd.cf and reside in the same directory on both the
# main and the spare SC.
```

```
# If the command is not part of the defined PATH for
# the user, the absolute filename must be passed with the
# initcmdsync command
#
initcmdsync script_name parameters
# The marker point is stored in the goto_label variable.
# Keep executing this script until all cases have been
# processed or an error is detected.
#
while (( $goto_label !=  0 )) ; do
#
# Each case should represent a synchronization point
# in the script.
#
case $goto_label in
#
# Step 1: Do something
#
1 )               do_something
                  .
                  .
                  .

# Execute the savecmdsync command with the script's
# descriptor and a unique marker to save the position.
# If a failover occurs here, the commands represented in
# the next goto_label (2) will be resumed.
#
           savecmdsync -M $(( $goto_label + 1 )) $desc
           goto_label=$(( $goto_label + 1 ))
             ;;
#
# Step 2: Do more things
#
2 )          do_more_things
             .
             .
             .
             savecmdsync -M $(( $goto_label + 1 )) $desc
               goto_label=$(( $goto_label + 1 ))
               ;;
#
# Step 3: Finish the last step and set the goto_label to 0
# so that the script ends.
3 )
             finish_last_step
             .
             .
             .
             goto_label=0
               ;;
 esac
done
# END OF MAIN CODE
# Remember to execute cancelcmdsync to remove the script from the
# command synchronization list. Otherwise, the command will be restarted
# after the failover.
#
cancelcmdsync $desc
```

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or domain configurator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

Note – The standard output for `initcmdsync` contains the command synchronization descriptor. Also, when failover is disabled (after a failover or in a single SC environment), scripts that contain synchronization commands generate error messages to the platform log file and return nonzero exit codes. These messages can be ignored.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`runcmdsync`(1M), `showcmdsync`(1M)

NAME

kmd - SMS key management daemon

SYNOPSIS

kmd

DESCRIPTION

`kmd`(1M) manages the IPSec security associations (SAs) necessary for securing the communication between the system controller (SC) and servers running on a domain. `kmd` manages per-socket policies for connections initiated by clients on the SC to servers on a domain. `kmd` manages shared policies for connections initiated by clients on the domain to servers on the SC.

The current default configuration includes authentication policies for the `dca`(1M) and `dxs`(1M) clients on the SC, which connect to the `dcs`(1M) and `cvcd`(1M) servers on a domain.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

Note – `kmd` must be run as a `root` process to be permitted to use the `pf_key` interface to IPSec.

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following file is used to configure `kmd`:

`/etc/opt/SUNWSMS/config/kmd_policy.cf`	`kmd_policy.cf` configures the shared and per-socket policies managed by `kmd`.

Changes to the policies are made by editing the `kmd_policy.cf` file on the SC. Corresponding changes must be made on the affected domain(s).

The format of `kmd_policy.cf` is a table of eight fields separated by the pipe (`|`) character:

dir | *d_port* | *protocol* | *sa_type* | *auth_alg* | *encr_alg* | *domain* | *login*

The fields are defined as follows:

dir	Direction to connect from. Values: `sctodom`, `domtosc`
d_port	Destination port.
protocol	Protocol for the socket. Values: `tcp`, `udp`
sa_type	Security association type. Values: `ah`, `esp`
auth_alg	Authentication algorithm. Values: `none`, `md5`, `sha1`
encr_alg	Encryption algorithm. Values: `none`, `des`, `3des`
domain	Domain ID. Values: integers `0–17`or a [*space*]. A space for the domain ID defines a policy that applies to all domains. A policy for a specific domain overrides a policy that applied to all domains.
login	Login name. Values: Any valid login name.

The default policies in the `kmd_policy.cf` file are as follows:

```
sctodom|665|tcp|ah|md5|none|  |sms-dca|
sctodom|442|tcp|ah|md5|none|  |sms-dxs|
```

The configuration of policies on a domain is the standard IPSec configuration file (`/etc/inet/ipsecconf.init`).

The default policies are shown below.

```
{ dport  sun-dr } permit { auth_alg md5 }
{ sport  sun-dr } apply {auth_alg md5 sa  unique }
{ dport  cvc_hostd } permit {auth_alg md5 }
{ sport  cvc_hostd } apply {auth_alg md5 sa  unique }
```

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSr SUNWSMSop

SEE ALSO

`ssd(1M)`, `sckmd(1M)`, `ipsecconf(1M)`, `pf_key(1M)`, `ipsec(1M)`, `dca(1M)`, `dxs(1M)`, `dcs(1M)`, `cvcd(1M)`

NAME

mand - management network daemon

SYNOPSIS

mand

DESCRIPTION

`mand`(1M) supports the management network (MAN) drivers and the failover management daemon (`fomd`(1M)) by providing the required network configuration. This configuration information includes host names, IP addresses, and netmasks. `mand` is also responsible for initializing and updating these respective fields in the platform configuration database (`pcd`(1M)) daemon as well as the in MAN driver on the system controller (SC).

`mand` is an SMS daemon running on both the main and spare SCs. Its role is set up by `fomd`.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

EXTENDED DESCRIPTION

SC-to-Domain and Domain-to-SC Internal Network (I1) data includes:

- Network mask
- SC host name
- SC IP address
- Domain[A–R] host name
- Domain[A–R] IP address

SC-to-SC Internal Network (I2) data includes:

- Network mask
- SC 0 host name
- SC 0 IP address
- SC 1 host name
- SC 1 IP address

SC External Community (C) data includes:

- Community Failover IP address
- Community physical interface name

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following configuration file is required:

`/etc/opt/SUNWSMS/config/MAN.cf`	This file includes the domain-to-SC, the SC-to-domain, and the SC-to-SC management network data as well as the community data for external access to the SC. Do *not* manually modify the `MAN.cf` file.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`fomd(1M)`, `pcd(1M)`, `smsconfig(1M)`, `ssd(1M)`

NAME

mld - message-logging daemon

SYNOPSIS

mld [`-f` *config_file*] [`-t`]

DESCRIPTION

`mld`(1M) provides logging services to all SMS daemons and processes. `mld` is the first SMS daemon started by `ssd`(1M) in order to capture the output of all other SMS daemons and processes during their startup phases.

- Platform log messages are stored in:

  `/var/opt/SUNWSMS/adm/platform/messages`

  in the following format:

  time host program `[`*pid*`]:[`*msg_id hrtime_t level file_line*`]` *message*

  file_line is optional and appears only in verbose mode.

  For example:

```
Feb 2 09:16:10 2002 sun15 mld[904]:[209 2345678901
INFO MLDLOGGER.cc 141] Platform messages file created.
```

- Domain log messages are stored in

  `/var/opt/SUNWSMS/adm/`*domain_id*`/messages`

  in the same format as platform messages, with additional *domain_id*|*domain_tag* information following the pid:

  time host program `[`*pid*`]`*domain_id*|*domain_tag*`:[`*msg_id hrtime_t level file_line*`]` *message*

  For example:

```
Feb 2 09:18:55 2002 sun15 mld[904]-B (engB):
[314 2345678902 ERR LogManager.cc 424]
message queue limit exceeded, messages will be dropped.
```

- Domain syslog messages are stored in

  `/var/opt/SUNWSMS/adm/`*domain_id*`/syslog`

  in the same format in which they are received.

OPTIONS

The following options are supported:

`-f` *config_file*	Provides an absolute path to an alternative remote-message-reception configuration file.
`-t`	Disables remote message reception (for example, domain `syslog` messages).

EXIT STATUS

.The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following file is used by this command:

`/var/opt/SUNWSMS/adm/.logger`	Message logging daemon configuration file.

This file supports three configuration directives:

FILE	Specifies where to send messages. The default is `msgdaemon` and should *not* be changed.
LEVEL	Specifies the minimum level necessary for `mld` to log a message. The supported levels are: `debug`, `info`, `notice`, `warning`, `err`, `crit` , `alert`, and `emerg`. The default is `notice`.
MODE	Specifies the verbosity of the messages. Two modes are available: `verbose` and `terse`. The default is `verbose`.

Note – All directive arguments are not case sensitive.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`ssd(1M)`

NAME

moveboard - move a board from one domain to another

SYNOPSIS

moveboard `-d` *domain_indicator* [`-c` *function*] [`-r` *retry_count* [`-t` *timeout*]] [`-q`] [`-f`] [`-y|-n`] *location*

moveboard `-h`

DESCRIPTION

`moveboard`(1M) first attempts to unconfigure, disconnect, and unassign *location* from the domain it is currently assigned to and possibly active in, and then proceeds to assign, connect, and configure *location* to the domain *domain_id* or *domain_tag*.

The `-c` *function* command option is used to specify the transition of the board from the current configuration state to a new configuration state. Configuration states are `assign`, `connect`, and `configure`. If the `-c` option is not specified, the default expected configuration state is `configure`.

Note – `moveboard` performs tasks synchronously and does not return control to the user until the command is complete. If the board is not powered on or tested and a `-c connect|configure` option is specified, then the command powers on the board and tests it.

Note – If the specified board is in the automatic system recovery (ASR) blacklist file, `moveboard` displays an error message when assigning a board and then continues. When using the `connect` or `configure` functions, `moveboard` displays an error message and then exits.

Note – If the `moveboard` command fails, a board does not return to its original state. A `dxs` or `dca` error message is logged to the domain. If the error is recoverable, you can retry the command. If it is unrecoverable, you need to reboot the domain in order to use that board.

OPTIONS The following options are supported.

`-c` *function*

Valid *function* values are `assign`, `connect`, and `configure`. One of these values is used to control the configuration state transition.

The possible transition states and their meanings are as follows:

`assign`

Unconfigures the board from the Solaris operating system running on the domain. Solaris software stops using any of the hardware resources on the board. (See `deleteboard -c unconfigure`.) Disconnects the board. Transitions the board into the `disconnected|unconfigured` state. (See `deleteboard -c disconnect`.) Unassigns the board from the current domain. (See `deleteboard -c unassign`.)Moves the board out of the logical domain by changing its state to `available`. Assigns the board to the new logical domain. This is a board state in which the domain has sole access to the board; however, the board is not active. Once assigned, the board can be connected or configured into the domain with the use of either `setkeyswitch on` or the connect or configure options.

`connect`	Unconfigures the board from the Solaris operating system running on the domain. Solaris software stops using any of the hardware resources on the board. (See `deleteboard -c unconfigure`.) Disconnects the board. Transitions the board into the `disconnected|unconfigured` state. (See `deleteboard -c disconnect.`) Unassigns the board from the current domain. (See `deleteboard -c unassign.`)Moves the board out of the logical domain by changing its state to `available`. Assigns the board to the new logical domain. (See `assign` earlier in this section; see also `addboard`(1M) `-c assign`.)Transitions the board into the `connected|unconfigured` state. In this state the board is assigned to the logical domain and connected (active). This state allows the normal system access to hardware resources on the board. The hardware resources of the board are not represented by the normal Solaris software data structures, however, and cannot be used by the Solaris operating system. Allowed operations on the board are limited to configuration administration operations. (See also `addboard -c connect.`) This is an intermediate state and does not have any standalone implementation at this time.

`configure` Unconfigures the board from the Solaris operating system running on the domain. Solaris software stops using any of the hardware resources on the board. (See `deleteboard -c unconfigure`.) Disconnects the board. Transitions the board into the `disconnected|unconfigured` state. (See `deleteboard -c disconnect`.) Unassigns the board from the current domain. (See `deleteboard -c unassign`.) Moves the board out of the logical domain by changing its state to `available`. Assigns the board to the new logical domain. (See `assign` earlier in this section; see also `addboard -c assign`.) Transitions the board into the `connected|unconfigured` state. In this state, the board is assigned to the logical domain and connected (active). This state allows the normal system access to hardware resources on the board. The hardware resources of the board are not represented by the normal Solaris software data structures, however, and cannot be used by the Solaris operating system. Allowed operations on the board are limited to configuration administration operations. (See `connect` earlier in this section; see also `addboard -c connect`.) Transitions the board into the `connected|configured` state. In this state, the board is not only assigned and connected to a domain, but also configured into the Solaris operating system. The hardware resources on the board can be used by Solaris software. (See also `addboard -c configure`.)

`-d` *domain_indicator* Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

This is the domain to which the board is being moved.

`-f`	Forces the specified action to occur. Typically, this is a hardware-dependent override of a safety feature. Forcing a state change operation can allow use of the hardware resources of an occupant that is not in the `ok` or `unknown` conditions, at the discretion of any hardware-dependent safety checks.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-n`	Automatically answers no to all prompts. Prompts are displayed unless used with the `-q` option.
`-q`	Quiet. Suppresses all messages to `stdout` including prompts. When used alone `-q` defaults to the `-n` option for all prompts. When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts and automatically answers with either 'y' or 'n' based on the option chosen.
`-r` *retry_count*	This command argument enables the user to specify retries in case of failures encountered during state transitions. The `-r` *retry_count* option can be used alone and indicates the number of times the configuration state change request should be retried by the domain.
`-t` *timeout*	This command argument enables the user to specify retries in case of failures encountered during state transitions. The `-t` *timeout* option cannot be used without the `-r` *retry_count* option and specifies the number of seconds that the domain should wait before the next retry is made. If the `-t` *timeout* is not specified, the default *timeout* is zero, meaning that the request is retried immediately.
`-y`	Automatically answers yes to all prompts. Prompts are displayed unless used with the `-q` option.

OPERANDS

The following operands are supported:

location — Board location separated by a space. Multiple *location* arguments are *not* permitted.

The following *location* forms are accepted:

Sun Fire 12K and E20K:

```
SB(0...8)

IO(0...8)
```

Sun Fire 15K and E25K:

```
SB(0...17)

IO(0...17)
```

Note – Use `showboards`(1M) to display board type.

EXTENDED DESCRIPTION

Group Privileges Required

Users with platform administrator privileges can perform the `-c assign` option if the board is in the `assigned` state (not active in a running domain) of the domain from which the board is being removed.

Users with domain administrator or configurator privileges can execute this command, but only on their respective domains. You must belong to both domain groups affected, and the board must be in the available component list of both domains.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

For all examples, if the board is currently active in another domain, you must have domain administrator privileges for that domain. In addition, you must have platform administrator privileges or the board must be in the domain available component list in order for it to be unassigned from its current domain.

You must have platform privileges or the board must be in the domain's available component list for you to assign the system board to a new domain. In addition, the user must have domain privileges in the new domain in order to connect or configure the board into the domain.

EXAMPLE 1 Assigning a CPU Board at SB4 to Domain A

```
sc0:sms-user:> moveboard -d A -c assign SB4
SB4 assigned to domain: A
```

EXAMPLE 2 Assigning a Blacklisted CPU Board at SB4 to Domain A

```
sc0:sms-user:> moveboard -d A -c assign SB4
SB4 assigned to domain: A
Warning: SB4 is blacklisted.
You will not be able to connect or configure it.
sc0:sms-user:>
```

EXAMPLE 3 Configuring an IO Board Into Domain A

Note: The default function is set to `configure`.

```
sc0:sms-user:> moveboard -d A IO2
IO2 unassigned from domain: B
IO2 assigned to domain: A
assign IO2
assign IO2 done
poweron IO2
poweron IO2 done
test IO2
test IO2 done
connect IO2
connect IO2 done
configure IO2
configure IO2 done
notify online /devices/pci@5d,700000
notify online /devices/pci@5d,600000
notify online /devices/pci@5c,700000
notify online /devices/pci@5c,600000
notify add capacity IO2 done
```

EXAMPLE 4 Connecting an IO Board at IO7 to Domain R

You must have platform privileges, or the board must be in the domain available component list.

```
sc0:sms-user:> moveboard -d R -c connect IO7
```

EXAMPLE 5 Connecting a Blacklisted Board to Domain C

```
sc0:sms-user:> moveboard -d C -c connect SB0
SB0 is blacklisted. Exiting.
sc0:sms-user:>
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`1`	No acknowledge.
`2`	Not supported.

3	Operation not supported.
4	Invalid privileges.
5	Busy.
6	System busy.
7	Data error.
8	Library error.
9	No library.
10	Insufficient condition.
11	Invalid.
12	Error.
13	A PID does not exist.
14	Invalid attribute.
30	Invalid board ID type.
31	Invalid permissions.
32	Assigned to another domain.
33	Unable to get permissions.
34	Unable to get domain board info.
35	Unable to get active board list.
36	Unable to get assigned board list.
38	Solaris not running.
39	Unable to assign/unassign domain state.
40	Unable to get domain permissions.
41	Unable to get platform permissions.
51	Invalid domain.
52	Invalid privileges.
55	Library error.
56	DR command syntax error.
58	Internal error.
59	Component blacklisted.
60	Unable to get ASR blacklist.

3	Operation not supported.
4	Invalid privileges.
5	Busy.
6	System busy.
7	Data error.
8	Library error.
9	No library.
10	Insufficient condition.
11	Invalid.
12	Error.
13	A PID does not exist.
14	Invalid attribute.
30	Invalid board ID type.
31	Invalid permissions.
32	Assigned to another domain.
33	Unable to get permissions.
34	Unable to get domain board info.
35	Unable to get active board list.
36	Unable to get assigned board list.
38	Solaris not running.
39	Unable to assign/unassign domain state.
40	Unable to get domain permissions.
41	Unable to get platform permissions.
51	Invalid domain.
52	Invalid privileges.
55	Library error.
56	DR command syntax error.
58	Internal error.
59	Component blacklisted.
60	Unable to get ASR blacklist.

61	Unable to get domain blacklist.
62	Unable to get platform blacklist.
64	Activity check error.
65	Unassign check error.
66	Unassign, unrestricted check error.
67	Domain permissions check error.
70	DR operation failed.

FILES

The following files are used by this command:

`/etc/opt/SUNWSMS/config/asr/blacklist`	List of components excluded by `esmd`.
`/etc/opt/SUNWSMS/config/platform/blacklist`	List of platform components excluded.
`/etc/opt/SUNWSMS/config/`*domain_id*`/blacklist`	List of domain components excluded.

Note – This file is created and used internally and should *not* be edited manually. To remove a component from the ASR blacklist file, use `enablecomponent`(1M).

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `addboard(1M)`, `deleteboard(1M)`, `enablecomponent(1M)`, `esmd(1M)`, `showcomponent(1M)`

NAME

osd - OpenBoot PROM server daemon

SYNOPSIS

osd

DESCRIPTION

`osd`(1M) provides software support for OpenBoot PROM. It provides an SMS event-based interface to `setkeyswitch`(1M) for laying out `IDPROM` and `REBOOTARGS` information prior to domain bring up.

`osd` also receives mailbox commands from OpenBoot PROM. These mailbox commands are acted upon, and a result is returned to OpenBoot PROM. Commands include `get-time-of-day`, `set-time-of-day`, `get-idprom`, `get-nvram-data`, `set-nvram-data`, `get-reboot-args`, `set-reboot-args`, and `do-tunnel-switch`. One instance of `osd` on the system controller (SC) is shared between all domains.

This daemon is automatically started by `ssd`(1M). Do *not* start it manually from the command line.

EXTENDED DESCRIPTION

Group Privileges Required

`osd` is run as the `sms-osd` user.

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`setkeyswitch(1M)`

NAME

pcd - platform configuration database daemon

SYNOPSIS

pcd

DESCRIPTION

`pcd`(1M) provides and manages controlled access to platform, domain, and system board configuration data. It is an SMS daemon running on the system controller (SC) and a key component of SMS configuration. All system management applications access the database information through the `pcd` daemon.

In addition to managing platform configuration data, the `pcd` notifies registered system management applications when pertinent database changes have occurred. These notifications are registered as events and are transparent to the user.

This daemon is started automatically by the `ssd`(1M) daemon. Do *not* start it manually from the command line.

EXTENDED DESCRIPTION

Platform data for SMS 1.6 includes:

```
Platform Type
Platform Name
Rack ID
Coherent Memory Address Slice Map
Clock Frequency
Clock Device Type
SC IP Address
SC Logical IP Host Adress
SC Host Netmask
SC0 to SC1 IP Address
SC1 to SC0 IP Adress
SC to SC IP Netmask
COD Head Room
Chassis Serial Number
```

Domain data for SMS 1.6 includes:

```
Domain Number
Domain Tag
OS Version
OS Type
Slot 0 Available Board List
Slot 1 Available Board List
Slot 0 Assigned Board List
Slot 1 Assigned Board List
Slot 0 Active Board List
Slot 1 Active Board List
Golden SRAM Expansion Slot
Keyswitch
Active Ethernet Expansion Slot
Creation Time
Domain State
Bringup Priority
IP Host Address
Hostname
Host Netmask
Broadcast Address
OBP Virtual Address
OBP Physical Address
Domain Reserved RTU
```

System board data for SMS 1.6 includes:

```
Expansion Number
Slot Number
Board Type Number
Board State Number
Domain Assigned Number
ABL State; Domain(s) Board Is In
Test Status
Test Level
Memory Clear State
COD Enable
```

SIGNALS

`SIGHUP` Rereads the database files and recaches information.

FILES

Note – *Never* modify these files by hand.

The following files are supported:

`/var/opt/SUNWSMS/.pcd/platform_info`	Contains platform database information.
`/var/opt/SUNWSMS/.pcd/domain_info`	Contains domain database information.
`/var/opt/SUNWSMS/.pcd/sysboard_info`	Contains system board database information

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`ssd(1M)`

NAME

poweroff - control power off

SYNOPSIS

poweroff [-q] [-y | -n] [*location*]

poweroff -h

DESCRIPTION

poweroff(1M) powers off the specified dual 48V power supply, fan tray, or board. If no arguments are specified and you have platform administrator privileges, the entire system, with the exception of the bulk power supplies, the fan trays, and the spare system controller (SC), powers off. If active domains are utilizing the component that is going to be powered off, a listing of those domains and an “Are you sure?" prompt are given by default.

Note – When you power off a bulk power supply poweroff trips the circuit breaker. In that case the poweron command alone can not restore power. You must manually flip the breaker back on and then run poweron.

If you do not have platform administrator privileges, the [*location*] command operand *must* be specified and the board must be assigned to a domain for which you have domain administrator or configurator privileges.

Note – This command has no effect on the position of the virtual keyswitch.

OPTIONS

The following options are supported:

-h — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to -h is ignored.

-n — Automatically answers no to all prompts. Prompts are displayed unless used with the -q option.

-q — Quiet. Suppresses all messages to stdout including prompts.

When used alone -q defaults to the -n option for all prompts.

When used with either the -y or the -n option, -q suppresses all user prompts, and automatically answers with either 'y' or 'n' based on the option chosen.

-y — Automatically answers yes to all prompts. Prompts are displayed unless used with the -q option.

OPERANDS

The following operands are supported:

location — Component location separated by a space. Multiple *location* forms are *not* permitted.

The following *location* forms are accepted:

Sun Fire 12K/E20K

`SB(0...8)`

`IO(0...8)`

`CS(0|1)`

`FT(0...7)`

`PS(0...5)`

`EX (0...8)`

`SC(0|1)` [Only the spare SC can be powered off.]

Sun Fire 15K /E25K

`SB(0...17)`

`IO(0...17)`

`CS(0|1)`

`FT(0...7)`

`PS(0...5)`

`EX(0...17)`

`SC(0|1)` [Only the spare SC can be powered off.]

EXTENDED DESCRIPTION

Group Privileges Required

You must have either platform administrator or domain administrator/configurator privileges to run this command.

If you have domain privileges, you must also specify the [*location*] operand, and the [*location*] must be a domain configuration unit (DCU) that is assigned to a domain for which you have domain privileges.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Powering Off a Power Supply

In this example `poweroff` prompts you for a reply.

```
sc0:sms-user:> poweroff ps5

!!!WARNING!!!WARNING!!!WARNING!!!WARNING!!!WARNING!!!
!!!WARNING!!!WARNING!!!WARNING!!!WARNING!!!WARNING!!!

This will trip the breakers on PS at PS5, which must be turned on manually!
Are you sure you want to continue to power off this component? (yes/no)? y
```

EXAMPLE 2 Powering Off a CPU Board at Expander Position 0

In this example `poweroff` forces the board off by suppressing any `stdout` messages and answering yes to all prompts.

```
sc0:sms-user:> poweroff -qy SB0
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`poweron(1M)`

NAME

poweron - control power on

SYNOPSIS

poweron [-q] [-y|-n] [*location*]

poweron -h

DESCRIPTION

poweron(1M) powers on the specified dual 48V power supply, fan tray, or board. If no arguments are specified and you have platform administrator privileges, the entire system is powered on (provided that no additional 48V power supply modules need to be powered on to support the new power requirements).

Note – When you power off a bulk power supply, poweroff trips the circuit breaker. In that case the poweron command alone cannot restore power. You must manually flip the breaker back on, and then run poweron.

If you do not have platform administrator privileges, the *location* command operand *must* be specified and the board must be assigned to a domain for which you have domain administrator or configurator privileges. If sufficient power and cooling is not available for the requested board, you are prompted whether to continue poweron, or to discontinue poweron. A minimum of five 48V power supply modules (4-KW modules) are required to power a fully configured system. Thus, with N+1 redundancy, six power supply modules are used. If powering on a component requires another component to be powered on, such as a board requiring a 48V power supply module, the poweron request fails.

For SMS 1.5 and subsequent releases, the default behavior has changed. When the poweron command fails, instead of terminating automatically, it displays a prompt asking whether you want to continue with the attempt to power on the system:

```
sc0:sms-user:> poweron SB0
EXB at EX0 is off. Attempt to poweron CPU at SB0 is discouraged.
Are you sure you want to continue the power ON (yes/no)?
```

If you enter “Y,” the system will continue to attempt a poweron, and will display the results. If you enter “N,” the system will discontinue the attempt to power on.

The -y and -q option automatically selects the “No” answer to this question, replicating the previous default behavior. The -y option does not apply to this particular prompt.

If a specified component is in the automatic system recovery (ASR) blacklist file, an error message is displayed. If you have platform administrator privileges you are prompted to continue. If you do not, the operation is terminated and an error message is displayed.

OPTIONS The following options are supported.

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-n` Automatically answers no to all prompts. Prompts are displayed unless used with the `-q` option.

`-q` Quiet. Suppresses all messages to `stdout` including prompts.

When used alone, `-q` defaults to the `-n` option for all prompts.

When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts, and automatically answers with either "Y" or "N" based on the option chosen.

`-y` Automatically answers "yes" to all prompts (except the prompt that appears when the `poweron` command fails). Prompts are displayed unless used with the `-q` option.

OPERANDS

The following operands are supported:

location Component location separated by a space. Multiple *location* forms are *not* permitted.

The following *location* forms are accepted:

Sun Fire 12K/E20K

```
SB(0...8)
IO(0...8)
CS(0|1)
FT(0...7)
PS(0...5)
EX (0...8)
```

Sun Fire 15K /E25K

```
SB(0...17)
IO(0...17)
CS(0|1)
FT(0...7)
PS(0...5)
EX(0...17)
```

SC(0|1) [Only the spare SC can be powered on.]

EXTENDED DESCRIPTION

Group Privileges Required

You must have either platform administrator or domain administrator/configurator privileges to run this command.

If you have domain privileges, you must also specify the *location* operand and the *location* must be a domain configuration unit (DCU) that is assigned to a domain for which you have domain privileges.

You must have platform administrator privileges in order to power on a board listed in the ASR blacklist file.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Powering On the Dual 48V Power Supply

The power supply is located on the front side in the 0 bank position.

```
sc0:sms-user:> poweron PS0
```

EXAMPLE 2 Powering On the Dual 48V Power Supply When Both AC Inputs are Bad

When the AC inputs are bad you will receive a warning.

```
sc0:sms-user:> poweron PS0
Both AC inputs to PS0 are bad, did you remember to turn on the breakers?
```

EXAMPLE 3 Powering On a CPU in the ASR Blacklist File

You must have platform administrator privileges. Otherwise, `poweron` terminates and displays an error message.

```
sc0:sms-user:> poweron SB0
Component SB0 is in the ASR blacklist.
Are you sure you want to continue the power ON (yes/no)? Y
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following file is used by this command:

`/etc/opt/SUNWSMS/config/asr/blacklist`	List of components excluded by `esmd`

Note – This file is created and used internally and should *not* be edited manually.

ATTRIBUTES

See `attributes(5)` for description of the following attribute.

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO

`esmd(1M)`, `poweroff(1M)`

NAME

rcfgadm - remote configuration administration

SYNOPSIS

rcfgadm `-d` *domain_indicator* [`-f`] [`-y|-n`] [`-v`] [`-o` *hardware_options*] `-c` *function* [`-r` *retry_count* [`-T` *timeout*]] *ap_id...*

rcfgadm `-d` *domain_indicator* [`-f`] [`-y|-n`] [`-v`] [`-o` *hardware_options*] `-x` *hardware_function ap_id...*

rcfgadm `-d` *domain_indicator* [`-v`] [`-a`] [`-s` *listing_options*] [`-o` *hardware_options*] [`-l` [*ap_id|ap_type*]...]

rcfgadm `-d` *domain_indicator* [`-v`] [`-o` *hardware_options*] `-t` *ap_id...*

rcfgadm `-d` *domain_indicator* [`-v`] [`-o` *hardware_options*] `-h` [*ap_id|ap_type*]

DESCRIPTION

`rcfgadm`(1M) provides remote configuration administration operations on dynamically reconfigurable hardware resources. The `rcfgadm` command allows configuration administration operations on the specified domain from the system controller. These operations include displaying status (`-l`), initiating testing (`-t`), invoking configuration state changes (`-c`), invoking hardware specific functions (`-x`), and obtaining configuration administration help messages (`-h`).

`rcfgadm` performs configuration administration at attachment points, which are places where system software supports dynamic reconfiguration of hardware resources during continued operation of Solaris software.

Configuration administration distinguishes between hardware resources that are physically present in the machine and hardware resources that are configured and visible to the Solaris environment. The nature of configuration administration functions are hardware-specific and rcfgadm performs configuration by calling hardware-specific libraries.

Configuration administration operates on an attachment point. Hardware resources located at attachment points might or might not be physically replaceable during system operation but are dynamically reconfigurable by way of the configuration administration interfaces.

An attachment point defines two unique elements, which are distinct from the hardware resources that exist beyond the attachment point. The two elements of an attachment point are a receptacle and an occupant. Physical insertion or removal of hardware resources occurs at an attachment point and results in a receptacle gaining or losing an occupant. Configuration administration supports the physical insertion and removal operations, as well as other configuration administration functions at an attachment point.

Attachment points have associated state and condition information. The configuration administration interfaces provide control for transitioning attachment point states. A receptacle can exist in one of three states: `empty`, `disconnected`, or

connected. An occupant can exist in one of two states: configured or unconfigured.

A receptacle can provide the empty state, which is the normal state of a receptacle when the attachment point has no occupants. A receptacle can also provide the disconnected state if it has the capability of isolating its occupants from normal system access. Typically this state is used for various hardware-specific testing prior to bringing the occupant's resources into full use by the system, or as a step in preparing an occupant for physical removal or reconfiguration. A receptacle in the disconnected state isolates its occupant from the system as much as its hardware allows, but can provide access for testing and setup. A receptacle must provide the connected state, which allows normal access to hardware resources contained on any occupants. The connected state is the normal state of a receptacle that contains an occupant and that is not currently undergoing configuration administration operations.

The hardware resources contained on an occupant in the unconfigured state are not represented by normal Solaris software data structures and are thus not available for use by the Solaris operating system. Operations allowed on an unconfigured occupant are limited to configuration administration operations. The hardware resources of an occupant in the configured state are represented by normal Solaris software data structures, and thus some or all of those hardware resources can be in use by the Solaris operating system. All occupants provide both the configured and unconfigured states.

An attachment point can be in one of five conditions: unknown, ok, failing, failed, or unusable. An attachment point can enter the system in any condition, depending upon results of power-on tests and nonvolatile record keeping.

An attachment point with an occupant in the configured state is in one of four conditions: unknown, ok, failing, or failed. If the condition is not failing or failed, an attachment point can change to failing during the course of operation if a hardware-dependent recoverable error threshold is exceeded. If the condition is not failed, an attachment point can change to failed during operation as a result of an unrecoverable error.

An attachment point with an occupant in the unconfigured state can be in any of the defined conditions. The condition of an attachment point with an unconfigured occupant can decay from ok to unknown after a system-dependent time threshold. Initiating a test function changes the attachment point condition to ok, failing, or failed, depending on the outcome of the test. An attachment point that does not provide a test function can leave the attachment point in the unknown condition. If a test is interrupted, the attachment point condition can be set to the previous condition, to unknown, or to failed. An attachment point in the unknown, ok, failing, or failed conditions can be retested.

An attachment point can exist in the unusable condition for a variety of reasons, such as inadequate power or cooling for the receptacle, an occupant that is

unidentifiable, unsupported, incorrectly configured, and so on. An attachment point in the `unusable` condition can never be used by the system. It typically remains in this condition until the physical cause is remedied.

An attachment point also maintains `busy` information that indicates when a state change is in progress or the condition is being reevaluated.

Designate attachment points using hardware-specific identifiers (*ap_id*s) that are related to the type and location of the attachment points in the system device hierarchy. An *ap_id* cannot be ambiguous; it must identify a single attachment point. Two types of *ap_id* specifications are supported: physical and logical.

A physical *ap_id* contains a fully specified path name, while a logical *ap_id* contains a shorthand notation that identifies an attachment point in a more user-friendly way.

For example, an attachment point representing system board 6 would have a physical *ap_id* of `/devices/pseudo/dr@0:SB6`, while the logical *ap_id* would be `SB6`.

Attachment points can also be created dynamically. A dynamic attachment point is named relative to a base attachment point that is present in the system. *ap_ids* for dynamic attachment points consist of a base component followed by two colons (::) and a dynamic component. The base component is the base attachment point *ap_id*. The dynamic component is hardware-specific and is generated by the corresponding hardware-specific library.

For example, consider a base attachment point, which represents a system board, with the physical *ap_id* `/devices/pseudo/dr@0:SB16` and logical *ap_id* `SB16`.

A CPU attached to this system board could be represented by a dynamic attachment point with logical *ap_id* `SB16::cpu2`, where `SB16` is the base component and `cpu2` is the hardware-specific dynamic component. Similarly, the physical *ap_id* for this dynamic attachment point would be:

```
/devices/pseudo/dr@0:SB16::cpu2.
```

An *ap_type* is a partial form of a logical *ap_id* that can be ambiguous and not specify a particular attachment point. An *ap_type* is a substring of the portion of the logical *ap_id*, up to but not including, the colon (:) separator. For example, an *ap_type* of `pci` would show all attachment points whose logical *ap_ids* begin with `pci`.

The use of *ap_type*s is discouraged. The new select suboption to the `-s` option provides a more general and flexible mechanism for selecting attachment points. See OPTIONS.

`rcfgadm` interacts primarily with hardware-dependent functions contained in hardware-specific libraries, and thus its behavior is hardware-dependent.

For each configuration administration operation, a service interruption can be required. If the requested operation requires a noticeable service interruption to interactive users, confirmation is requested before the operation is started.

A prompt is displayed on the standard error output for confirmation on the standard input. Confirmation can be overridden with the `-y` or `-n` option to always answer `yes` or `no`, respectively. Hardware-specific options, such as *test level*, are supplied as suboptions using the `-o` option.

Operations that change the state of the system configuration are audited by the system log daemon `syslogd`(1M).

The arguments for this command conform to the `getopt`(3C) and `getsubopt`(3C) syntax conventions.

Refer to the *Sun Fire High-end and Midrange Servers Dynamic Reconfiguration User Guide* for more information.

OPTIONS

The following options are supported.

Note – If the `rcfgadm` command fails, a board does not return to its original state. A `dxs` or `dca` error message is logged to the domain. If the error is recoverable, you can retry the command.

- **If you are running Solaris 8 or Solaris 9 OS on the domains, perform the following check:**

  a. **Before you retry the command, ensure that the following `dcs` entries exist in `/etc/inetd.conf` on the domain, and that they have not been disabled:**

```
sun-dr stream tcp wait root /usr/lib/dcs dcs
sun-dr stream tcp6 wait root /usr/lib/dcs dcs
```

  b. **If the error is unrecoverable, you must reboot the domain in order to use the board.**

- **If you are running the Solaris 10 OS on the domain, the `dcs` is now part of the SMF (Service Management Facility). Perform the following steps:**

  a. **Make sure you are logged in as root.**

  b. **Type the following command at the system prompt on the domain:**

```
% inetadm | grep dcs

disabled disabled svc: /platform/sun4u/dcs: default
```

c. If the `dcs` is disabled as shown in the above example, enable it by typing the following command:

```
% svcadm enable svc:/platform/sun4u/dcs:tcp
```

`-a` Specifies that the `-l` option must also list dynamic attachment points.

`-c` *function* Performs the state change function on the attachment point specified by *ap_id*.

Specify function as `disconnect`, `connect`, `configure`, or `unconfigure`. These functions cause state transitions at the attachment point by calling hardware-specific library routines.

The possible transition states and their meanings are as follows:

- `disconnect`

  Change the receptacle state to disconnected.

  If the occupant state is configured, the disconnect function first attempts to unconfigure the occupant. The `disconnect` function powers the board off by default. The board is ready to be removed from the slot at that point. The `-o nopoweroff` option specifies skipping the power off step, leaving the board powered on. The board is left assigned to the domain by default. The `-o unassign` option instructs the domain to give up the ownership of the board once the board is disconnected. Once the board has been unassigned, it may no longer be accessible to `cfgadm` because another domain might have `assigned` the board to itself.

- `connect`

  Performs hardware-specific operations to put the receptacle into the `connected` state, which allows an occupant to operate normally through the receptacle.

- `configure`

  Performs hardware-specific operations that allow an occupant's hardware resources to be used by Solaris software. Occupants that are configured are part of the system configuration and are available for manipulation by Solaris software device manipulation maintenance commands (for example, `psradm`(1M), `mount`(1M), and `ifconfig`(1M)).

- `unconfigure`

  Performs hardware-specific operations that logically remove an occupant's hardware resources from the system. The occupant must currently be configured, and its hardware resources must not be in use by the Solaris operating system.

State transition functions can fail due to the condition of the attachment point or other hardware-dependent considerations. All state change functions in the direction of adding resources (`connect` and `configure`) are passed on to the hardware-specific library when the attachment point is in the `ok` or `unknown` condition. All other conditions require the use of the force (`-f`) option to allow these functions to be passed on to the hardware-specific library. Attachment point condition does not prevent a hardware-specific library being called for the removal (`disconnect` and `unconfigure`) of hardware resources from the system. Hardware-specific libraries can reject state change functions if the attachment point is in the `unknown` condition.

The condition of an attachment point is not necessarily changed by the state change functions; however, errors during state change operations can change the attachment point condition. Specify the force option (`-f`) to override a condition and force a state change that would otherwise fail. Hardware-specific safety and integrity checks can prevent the force option from having any effect.

`-d` domain_indicator — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-f` — Forces the specified action to occur. Typically, this is a hardware-dependent override of a safety feature. Forcing a state change operation can allow use of the hardware resources of an occupant that is not in the `ok` or `unknown` condition, at the discretion of any hardware-dependent safety checks.

`-h` [*ap_id* | *ap_type*] — Prints the help message text. If *ap_id* or *ap_type* is specified, the help routine of the hardware-specific library for the attachment point indicated by the argument is called.

`-l` [*ap_id* | *ap_type*]	Lists the state and condition of attachment points specified. Filter attachment points with the `-s` option and select suboption. Invoking `rcfgadm` without one of the action options is equivalent to `-l` without an argument. The format of the list display is controlled by the `-v` and `-s` options. When the `-a` option is specified, attachment points are dynamically expanded.
`-n`	Automatically answers no to all prompts.
`-o` *hardware_options*	Supplies hardware-specific options to the main command option. The following are valid *hardware_options*: ■ `parsable` Applies only when the `-s` option is used. The `parsable` suboption specifies that info is returned for system board and IO assembly only. ■ `unassign` Applies only when the `-c` disconnect option is used. The `unassign` suboption specifies that the domain is to give up ownership of the board. ■ `nopoweroff` Applies only when the `-c` disconnect option is used. The `nopoweroff` suboption specifies that the board is not to be powered off after it is disconnected.
`-r` *retry_count*	Specifies the number of times the dynamic reconfiguration (DR) request is retried on the domain. The default is zero.
`-s` *listing_options*	Supplies listing options to the list (`-l`) command. *listing_options* conforms to the `getsubopt`(3C) syntax convention. The suboptions are used to specify the attachment point selection criteria (`select=select_string`), the type of matching desired (`match=match_type`), the order of listing (`sort=field_spec`), the data that is displayed (`cols=field_spec` and `cols2=field_spec`), the column delimiter (`delim=string`), and whether to suppress column headings (`noheadings`).

When the select suboption is specified, only attachment points that match the specified criteria are listed. The select suboption has the following syntax:

```
rcfgadm -s
select=attr1(value1):attr2(value2)...
```

where an `attr` is one of *ap_id*, *class*, or *type*. *ap_id* refers to the logical *ap_id* field, *class* refers to attachment point class, and *type* refers to the type field. *value1*, *value2*, and so on are the corresponding values to be matched. The type of match can be specified by the match suboption as follows:

```
rcfgadm -s
match=match_type,select=attr1(value1)...
```

where *match_type* can be either `exact` or `partial`. The default value is `exact`.

Suboptions can contain special characters that can be interpreted in ways other than as part of `rcfgadm` suboptions. For example, a command might contain parentheses, which are acceptable for suboptions but are interpreted as special characters when entered on the command line. Arguments to the select suboption can be quoted to protect them from the UNIX C shell interpretation.

A *field_spec* is one or more data-fields concatenated with the use of a colon (:), as in `data-field:data-field:data-field`. A `data-field` is comprised of *ap_id*, *physid*, *r_state*, *o_state*, *condition*, *type*, *busy*, *status_time*, *status_time_p* and *info*. The *ap_id* field output is the logical name for the attachment point, while the *physid* field contains the physical name. The *r_state* field can be `empty`, `disconnected`, or `connected`. The *o_state* field can be `configured` or `unconfigured`. The *busy* field can be either `y` if the attachment point is *busy*, or `n` if it is not. The *type* and *info* fields are hardware-specific. The *status_time_p* field is a parsable version of the *status_time* field. If an attachment point has an associated class, the *class* field lists the class name.

The order of the fields in *field_spec* is significant. For the sort suboption, the first field given is the primary sort key. For the `cols` and `cols2` suboptions, the fields are printed in the order requested. Reverse the order of sorting on a data-field by placing a minus (-) before the data-field name within the *field_spec* for the sort suboption. The default value for sort is *ap_id*. The default values for `cols` and `cols2` depend on whether the `-v` option is given. Without it, `cols` is `ap_id:r_state:o_state:condition` and `cols2` is not set; with `-v`, `cols` is `ap_id:r_state:o_state:condition:info` and `cols2` is `status_time:type:busy:physid`.

The default value for `delim` is a single space. The value of `delim` can be a string of arbitrary length. The delimiter cannot include a comma (,) character; see `getsubopt`(3C). These listing options can be used to create parsable output.

`-T` *timeout*

Specifies the time interval, in seconds, between retries. This option cannot be used alone and must be specified with the `-r` *retry_count* option. The default value is zero, meaning the that DR request is retried immediately.

`-t`

Performs a test of one or more attachment points. The test function is used to reevaluate the condition of the attachment point.

The results of the test are used to update the condition of the specified occupant to `ok` if no faults are found, `failing` if recoverable faults are found, or `failed` if any unrecoverable faults are found.

If a test is interrupted, the attachment point condition can be restored to its previous value, set to `unknown` if no errors were found, set to `failing` if only recoverable errors were found, or set to `failed` if any unrecoverable errors were found. The attachment point should be set to `ok` only upon normal completion of testing with no errors.

`-v` Executes in verbose mode. For the `-c`, `-t`, and `-x` options, displays a message giving the results of each attempted operation. Displays detailed help information for the `-h` option. Displays verbose information for each attachment point for the `-l` option.

`-x` *hardware_function* Performs hardware-specific functions.

Lists hardware-specific private functions using `rcfgadm -h` *ap_id*.

The following are valid for *hardware_function*:

- `assign` *ap_id*

  Assign a board to a domain.

- `unassign` *ap_id*

  Unassign a board from a domain.

- `poweron` *ap_id*

  Power on a board.

- `poweroff` *ap_id*

  Power off a board.

`-y` Automatically answers yes to all prompts. Prompts are displayed.

OPERANDS **The following operands are supported:**

ap_id

Refer to attachment points by hardware-specific identifiers (*ap_id*s) that are related to the type and location of the attachment points in the system device hierarchy. An *ap_id* cannot be ambiguous; it must identify a single attachment point. Two types of *ap_id* specifications are supported: physical and logical. A physical *ap_id* contains a fully specified path name, while a logical *ap_id* contains a shorthand notation that identifies an attachment point in a more user-friendly way.

Physical *ap_ids*:

```
/devices/pseudo/dr@0:IO4

/devices/pseudo/dr@0:IO6

/devices/pseudo/dr@0:IO14

/devices/pseudo/dr@0:SB4

/devices/pseudo/dr@0:SB6
```

Logical *ap_ids*:

```
IO4

IO6

IO14

SB4

SB6
```

ap_type An *ap_type* is a partial form of a logical *ap_id* that can be ambiguous and not specify a particular attachment point. An *ap_type* is a substring of the portion of the logical *ap_id* up to, but not including, the colon (:) separator. For example, an *ap_type* of `pci` would show all attachment points whose logical *ap_id*s begin with `pci`. The two *ap_types* shown here are static and dynamic.

Static *ap_types*:

`HPCI`

`CPU`

`MCPU`

`pci-pci/hp`

Dynamic *ap_types*:

`cpu`

`mem`

`io`

EXTENDED DESCRIPTION

Group Privileges Required

The privileges required for using this command depend on the desired operation. `rcfgadm` can assign or unassign boards that are not connected to a domain. To assign or unassign a board, you must have either platform administrator privileges or domain administrator/configurator privileges for the specified domain *and* the board must be in the domain's available component list. For more information see `setupplatform`(1M) and `showplatform`(1M).

The `assign` and `unassign` operations are private, hardware-specific operations. Assign a board using `rcfgadm -x assign` *ap_id*. Unassign a board using `rcfgadm -x unassign` *ap_id*. The *ap_id*s for `assign` and `unassign` must be logical *ap_id*s specifying a board, such as SB0 or IO2.

Domain administrator or domain configurator privileges are required for test, state change, list or hardware-specific operations.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Listing Attachment Points in the Device Tree for Domain A

The following example lists all attachment points except dynamic attachment points.

```
sc0:sms-user:>  rcfgadm -d a
Ap_Id       Type     Receptacle     Occupant       Condition
IO4         PCI      connected      configured     ok
IO6         MCPU     disconnected   unconfigured   unknown
IO14        PCI      connected      configured     ok
SB4         CPU      disconnected   unconfigured   unknown
SB6         CPU      connected      configured     ok
SB16        CPU      connected      configured     ok
```

EXAMPLE 2 Listing All Configurable Hardware Information for Domain A

The following example lists all current configurable hardware information, including those represented by dynamic attachment points:

```
sc0:sms-user:>  rcfgadm -d a -al
Ap_Id            Type       Receptacle    Occupant       Condition
IO4              PCI        connected     configured     ok
IO4::pci0        io         connected     configured     ok
IO4::pci1        io         connected     configured     ok
IO4::pci2        io         connected     configured     ok
IO4::pci3        io         connected     configured     ok
IO6              MCPU       disconnected  unconfigured   unknown
IO14             PCI        connected     configured     ok
IO14::pci0       io         connected     configured     ok
IO14::pci1       io         connected     configured     ok
IO14::pci2       io         connected     configured     ok
IO14::pci3       io         connected     configured     ok
SB4              CPU        disconnected  unconfigured   unknown
SB6              CPU        connected     configured     ok
SB6::cpu0        cpu        connected     configured     ok
SB6::cpu1        cpu        connected     configured     ok
SB6::cpu2        cpu        connected     configured     ok
SB6::cpu3        cpu        connected     configured     ok
SB6::memory      memory     connected     configured     ok
SB16             CPU        connected     configured     ok
SB16::cpu0       cpu        connected     configured     ok
SB16::cpu1       cpu        connected     configured     ok
SB16::cpu2       cpu        connected     configured     ok
SB16::cpu3       cpu        connected     configured     ok
SB16::memory     memory     connected     configured     ok
```

EXAMPLE 3 Creating a Selective List Based on Attachment Point Attributes for Domain A

The following example lists all attachment points at *location* SB6 and of *type* `cpu`. The argument to the `-s` option is quoted to protect it from the shell.

```
sc0:sms-user:>  rcfgadm -d a  -s match=partial,select="type(cpu)"  -
la SB6

Ap_Id          Type      Receptacle   Occupant      Condition
SB6::cpu0      cpu       connected    configured    ok
SB6::cpu1      cpu       connected    configured    ok
SB6::cpu2      cpu       connected    configured    ok
SB6::cpu3      cpu       connected    configured    ok
```

EXAMPLE 4 Listing Current Configurable Hardware Information in Verbose Mode for Domain A

The following example lists current configurable hardware information in verbose mode:

```
sc0:sms-user:> rcfgadm -d a -v -l  SB16
Ap_Id   Receptacle   Occupant     Condition   Information
SB16    connected    configured   ok          powered-on, assigned
When          Type  Busy  Phys_Id
Mar 6 13:30   CPU   n     /devices/pseudo/dr@0:SB16
```

EXAMPLE 5 Using the Force Option on Domain A

The following example configures an occupant in the failing state to the system using the force option:

```
sc0:sms-user:> rcfgadm -d a  -f -c configure SB6
```

EXAMPLE 6 Unconfiguring an Occupant From the System on Domain A

The following example unconfigures an occupant from the system:

```
sc0:sms-user:> rcfgadm -d a -c unconfigure IO14
```

EXAMPLE 7 Configuring an Occupant at an Attachment Point

The following example configures an occupant:

```
sc0:sms-user:> rcfgadm -d a -c configure SB6
```

EXAMPLE 8 Using the -o parsable option:

The following example displays system board and IO assembly information as a set of "name=value" pairs:

```
sc0:sms-user:> rcfgadm -d G -s cols=ap_id:type -o parsable
Ap_Id                          Type
IO0                            unknown
IO5                            HPCI
IO11                           HPCI
SB0                            CPU
SB11                           CPU
```

EXAMPLE 9 Disconnecting But Not Powering Off SB0:

The following example displays domain G giving up ownership of the board.

```
sc0:sms-user:> rcfgadm  -d G -c disconnect -o unassign,nopoweroff
SB0
```

ENVIRONMENT VARIABLES

See **environ**(5) for descriptions of the following environment variables that affect the execution of command_name: LC_TIME,LC-MESSAGES,TZ.

LC_MESSAGES	Determines how rcfgadm displays column headings and error messages. Listing output data is not affected by the setting of this variable.
LC_TIME	Determines how rcfgadm displays human-readable status changed time (*status_time*).
TZ	Specifies the time zone used when converting the status changed time. This applies to both the human-readable (*status_time*) and parsable (*status_time_p*) formats.

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	No acknowledge.
2	Not supported.
3	Operation not supported.
4	Invalid privileges.
5	Busy.
6	System busy.
7	Data error.
8	Library error.

9	No library.
10	Insufficient condition.
11	Invalid.
12	Error.
13	A PID does not exist.
14	Invalid attribute.
30	Invalid board ID type.
31	Invalid permissions.
32	Assigned to another domain.
33	Unable to get permissions.
34	Unable to get domain board info.
35	Unable to get active board list.
36	Unable to get assigned board list.
37	Get blacklist failed.
38	Solaris not running.
39	Invalid privileges.
40	Unable to get domain permissions.
41	Unable to get platform permissions.
42	Failed to get domain blacklist.
43	Failed to get platform blacklist.
56	DR command syntax error.
70	DR operation failed.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `cfgadm_sbd(1M)`, `setupplatform(1M)`, `showplatform(1M)`

DIAGNOSTICS

Diagnostic messages appear on the standard error output. Other than options and usage errors, the following are diagnostic messages produced by this utility:

`rcfgadm`: Configuration administration not supported on *ap_id*

`rcfgadm`: No library found for *ap_id*

`rcfgadm`: *ap_id* is ambiguous

`rcfgadm`: Operation: Insufficient privileges

`rcfgadm`: Attachment point is `busy`, try again

`rcfgadm`: No attachment points with specified attributes found

`rcfgadm`: System is busy, try again

`rcfgadm`: Operation: Operation requires a service interruption

`rcfgadm`: Operation: Data error: error_text

`rcfgadm`: Operation: Hardware specific failure: error_text

`rcfgadm`: Attachment point not found

`rcfgadm`: Configuration operation succeeded

`rcfgadm`: Configuration operation canceled

`rcfgadm`: Configuration operation invalid

`rcfgadm`: Configuration operation not supported

`rcfgadm`: Library error

`rcfgadm`: Insufficient condition

`rcfgadm`: SCDR/DCA door failure

`rcfgadm`: DCA/DCS communication error

`rcfgadm`: DCA internal failure

`rcfgadm`: PCD event failure

`rcfgadm`: Callback function failure

`rcfgadm`: SCDR library internal error

`rcfgadm`: Board is already assigned to another domain

`rcfgadm`: Unable to get `active` or `assigned` domain info

`rcfgadm`: Unable to get privileges

`rcfgadm`: DRCMD library invalid parameter

See `config_admin`(3CFGADM) for additional details regarding error messages.

NAME

reset - send reset to all CPU ports of a specified domain

SYNOPSIS

reset `-d` *domain_indicator* [, *domain_indicator*]...
[`-d` *domain_indicator* [, *domain_indicator*]...]... [`-q`] [`-y`|`-n`] [`-x`]

reset `-h`

DESCRIPTION

`reset`(1M) enables you to reset one or more domains in one of two ways: reset the hardware to a clean state or send an externally initiated reset (XIR) signal. The default is to reset the hardware to a clean state. You receive an error if the virtual keyswitch is in the `secure` position. By default, `reset` gives an optional confirmation prompt. Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

OPTIONS

The following options are supported:

`-d` *domain_indicator*

Specifies the domain using one of the following:

domain_id –ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

Note – Multiple *domain_indicator*s must be separated by a comma.

`-h`

Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-n`

Automatically answers “no” to all prompts. Prompts are displayed unless used with the `-q` option.

`-q`

Quiet. Suppresses all messages to `stdout`, including prompts.

When used alone, `-q` defaults to the `-n` option for all prompts.

When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts and automatically answers with either “y” or “n” based on the option chosen.

`-x`

Sends an XIR signal to the processors in the specified domain.

`-y`

Automatically answers “yes” to all prompts. Prompts are displayed unless used with the `-q` option.

EXTENDED DESCRIPTION

Group Privileges Required

You must have domain administrator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Resetting Domain C

```
sc0:sms-user:> reset -d C
Do you want to send RESET to domain C? [y|n]: y

RESET initiated for domain: C

Powering on: CSB at CS0

Already powered on: CSB at CS0

Powering on: CSB at CS1

Already powered on: CSB at CS1

Powering on: EXB at EX0

Already powered on: EXB at EX0

Powering on: HPCI at IO0
...
...
...
```

EXAMPLE 2 XIR Reset of Domain C

```
sc0:sms-user:> reset -d C -x
Do you want to send XIR to domain C? [y|n]:y
XIR to processor SB3/P0 initiated
XIR to processor SB3/P1 initiated
XIR to processor SB3/P2 initiated
XIR to processor SB3/P3 initiated
XIR initiated to all processors for domain: C
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

ATTRIBUTES

See `attributes(5)` for a description of the following attribute.

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO | `addtag(1M)`

NAME

resetsc - reset the *other* system controller (SC)

SYNOPSIS

resetsc [-q] [-y|-n]

resetsc -h

DESCRIPTION

resetsc(1M) resets the *other* SC. This might typically be done after failover. This command either runs from the main SC and resets the spare or runs from the spare and resets the main. An SC cannot reset itself. If the SC chosen is not powered on, resetsc prompts the user to power it on. If the chosen SC does not power on, resetsc terminates and returns an error.

OPTIONS

The following options are supported:

-h — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to -h is ignored.

-n — Automatically answers no to all prompts. Prompts are displayed unless used with the -q option.

-q — Quiet. Suppresses all messages to stdout including prompts.

When used alone -q defaults to the -n option for all prompts.

When used with either the -y or the -n option, -q suppresses all user prompts and automatically answers with either 'y' or 'n' based on the option chosen.

-y — Automatically answers yes to all prompts. Prompts are displayed unless used with the -q option.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Resetting the Other SC Using Prompts

```
sc0:sms-user:> resetsc
About to reset other SC.
Are you sure you want to continue? (y or [n])
```

EXAMPLE 2 Resetting the Other SC When the Other SC Is Powered Off

```
sc0:sms-user:> resetsc
The other SC is not powered on.
Do you want to try to power it on? (y or [n])
```

EXAMPLE 3 Resetting the Other SC Answering Yes to All Prompts

```
sc0:sms-user:> resetsc -y
About to reset other SC.
Are you sure you want to continue? [y]
```

EXAMPLE 4 Resetting the Other SC Suppressing All Prompts

```
sc0:sms-user:> resetsc -q
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	The user has invalid permission.
2	Memory allocation failed.
3	Cannot determine other SC's presence.
4	Other SC is not present.
5	Cannot determine power state of other SC.
6	Unable to power on other SC.
7	Unable to reset other SC.
8	Flag registration failed.
9	Invalid command-line argument.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

NAME

runcmdsync - prepare a specified script for recovery after a failover

SYNOPSIS

runcmdsync *script_name* [*parameters*]

runcmdsync `-h`

DESCRIPTION

The `runcmdsync`(1M) command prepares the specified script for automatic synchronization (recovery) after a failover. `runcmdsync` creates a command synchronization descriptor that identifies the script to be recovered. This descriptor is added to the command synchronization list that determines the scripts to be restarted after a failover. The `runcmdsync` command also removes this descriptor from the command synchronization list when the script terminates.

To specify restart points in a script, see `initcmdsync`(1M) and the family of synchronization commands.

OPTIONS

The following options are supported:

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

script_name Identifies the script to be prepared for command synchronization. *script_name* must be the absolute path name of an executable command. The command must exist in the same location on both SCs.

parameters Specifies the options or parameters associated with the specified script. These parameters are stored on the spare system controller (SC) and are used to restart the specified command or script after an automatic failover.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or domain configurator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`cancelcmdsync(1M)`, `initcmdsync(1M)`, `savecmdsync(1M)`, `showcmdsync(1M)`

NAME

savecmdsync - command synchronization command

SYNOPSIS

cancelcmdsync *cmdsync_descriptor*

initcmdsync *script_name* [*parameters*]

savecmdsync `-M` *identifier cmdsync_descriptor*

[cancel | init | save]cmdsync `-h`

DESCRIPTION

The command synchronization commands work together to control the recovery of user-defined scripts interrupted by a system controller (SC) failover. Insert the following commands in user-defined scripts to enable command synchronization:

`initcmdsync` Creates a command synchronization descriptor that identifies the script to be recovered.

This descriptor is placed on a command synchronization list that identifies the scripts and commands to be restarted on the new main SC after a failover.

`savecmdsync` Adds a marker that identifies a location in the script from which processing can be resumed after a failover.

`cancelcmdsync` Removes a command synchronization descriptor from the command synchronization list. This ensures that the script is run only once and not after subsequent failovers.

Be sure that all exit paths of a script have a `cancelcmdsync` sequence to remove the descriptor from the command synchronization list. If you do not remove the descriptor and a failover occurs, the script is rerun on the new main SC.

Note – Both an `initcmdsync` and a `cancelcmdsync` sequence must be contained within a script to enable command synchronization. The use of the `savecmdsync` command is optional, and marks only specific points in a script from which processing can be resumed. If specific restart points are not necessary, consider using `runcmdsync`(1M) instead.

OPTIONS

The following options are supported:

cmdsync_descriptor	Specifies the command synchronization descriptor that identifies the user-defined script. This descriptor is the standard output value returned by the `initcmdsync` command.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-M` *identifier*	Marks a location in the script from which the script can be resumed after a failover. The identifier must be a positive integer.
parameters	Specifies the options or parameters associated with the user-defined script. These parameters are stored on the spare SC and are used to restart the specified script after a failover.
script_name	Identifies the name of the user-defined script to be synchronized. *script_name* must be the absolute path name of an executable command. The command must exist in the same location on both SCs.

EXTENDED DESCRIPTION

The command synchronization commands are inserted at certain logical points within a user-defined script.

For instance, a Korn shell script might be structured as follows:

```
# MAIN CODE STARTS HERE
# Be sure to use a cleanup procedure to handle any
# interrupts.
# Use the cancelcmdsync to remove the script from the
# command synchronization list. Otherwise, the command
# will get restarted on the new main SC.
#
clean_up () {
        cancelcmdsync  $desc
        exit
    }

# Declare the clean_up function to capture system signals
# and cleanup.
trap "clean_up" INT HUP TERM QUIT PWR URG
goto_label=1
# Process the arguments, capturing the -M marker point
# if provided
#
for arg in $*; do
       case $arg in
            -M )
       goto_label=$arg;;
       .
       .
       .
       esac
   done
# Place this script and all its parameters in the command
# synchronization list, which indicates the commands to
# be restarted after an SC failover.
#
# NOTE: The script must be executable by the user defined
# in fomd.cf and reside in the same directory on both the
# main and the spare SC.
```

```
# If the command is not part of the defined PATH for
# the user, the absolute filename must be passed with the
# initcmdsync command
#
initcmdsync script_name parameters
# The marker point is stored in the goto_label variable.
# Keep executing this script until all cases have been
# processed or an error is detected.
#
while (( $goto_label !=  0 )) ; do
#
# Each case should represent a synchronization point
# in the script.
#
case $goto_label in
#
# Step 1: Do something
#
1 )               do_something
                  .
                  .
                  .

# Execute the savecmdsync command with the script's
# descriptor and a unique marker to save the position.
# If a failover occurs here, the commands represented in
# the next goto_label (2) will be resumed.
#
           savecmdsync -M $(( $goto_label + 1 )) $desc
           goto_label=$(( $goto_label + 1 ))
             ;;
#
# Step 2: Do more things
#
2 )          do_more_things
             .
             .
             .
             savecmdsync -M $(( $goto_label + 1 )) $desc
               goto_label=$(( $goto_label + 1 ))
               ;;
#
# Step 3: Finish the last step and set the goto_label to 0
# so that the script ends.
3 )
             finish_last_step
             .
             .
             .
             goto_label=0
               ;;
 esac
done
# END OF MAIN CODE
# Remember to execute cancelcmdsync to remove the script from the
# command synchronization list. Otherwise, the command will be restarted
# after the failover.
#
cancelcmdsync $desc
```

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or domain configurator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

Note – The standard output for `initcmdsync` contains the command synchronization descriptor. Also, when failover is disabled (after a failover or in a single SC environment), scripts that contain synchronization commands generate error messages to the platform log file and return nonzero exit codes. These messages can be ignored.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`runcmdsync(1M)`, `showcmdsync(1M)`

NAME

setbus - perform dynamic bus reconfiguration on active expanders in a domain

SYNOPSIS

setbus [`-q`] [`-y`|`-n`] `-c` *csb* [`-b` *buses*] [*location*...]

setbus `-h`

DESCRIPTION

`setbus`(1M) dynamically reconfigures bus traffic on active expanders in a domain to use either one centerplane support board (CSB) or both. Using both CSBs is considered `normal` mode. Using one CSB is considered `degraded` mode.

This feature enables you to swap out a CSB without having to power off the system.

OPTIONS

The `-y` and `-n` are optional arguments that take effect only if the `setbus` command displays a confirmation message such as the one shown below. The -y argument supplies an automatic affirmative response to the confirmation message. The -n argument supplies an automatic negative response.

If changing the configuration on the chosen expander requires changing the configuration on additional expanders, `setbus` displays the following prompt:

```
The expander board in position location communicates with expanders not
already listed, and will be added to the list of boards to reconfigure.
Are you sure you want to continue the reconfiguration? (yes/no)?
```

Additionally, `setbus` resets any boards that are powered on but not active. Any attach-ready state is lost. To bring the boards back to an attach-ready state, refer to the *Sun Fire High-end and Midrange Systems Dynamic Reconfiguration User Guide*.

Note – If you have degraded all expanders to one CSB you cannot un-degrade a single expander; you must set them all at the same time, otherwise `setbus` fails.

The following options are supported

`-b` *buses*
: Specifies which *buses* to configure. There are three *buses* to configure. Valid *buses* are:

  `a` — Configures the address bus

  `d` — Configures the data bus

  `r` — Configures the response bus

  The default is to configure all three buses.

`-c` *csb*
: Specifies which CSB(s) to use.

  `CS0` — Configures the hardware to use CS0 (`degraded` mode)

  `CS1` — Configures the hardware to use CS1 (`degraded` mode)

  `CS0,CS1` — Configures the hardware to use both CSBs (`normal` mode)

`-h`
: Help. Displays usage descriptions.

  Note – Use alone. Any option specified in addition to `-h` is ignored.

`-n`
: Automatically answers “no” to any prompt displayed by the setbus command.

`-q`
: Quiet. Suppresses all messages to `stdout` including prompts.

  When used alone, `-q` defaults to the `-n` option for all prompts.

  When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts and automatically answers with either 'y' or 'n' based on the option chosen.

`-y`
: Automatically answers “yes” to any prompt displayed by the setbus command.

OPERANDS

The following operands are supported:

location Specifies which expander slots to configure. The default is to configure all. Multiple *location*s are separated by spaces.

Valid *location*s are:

Sun Fire 12K/E20K:

```
EX(0...8)
```

Sun Fire 15K/E25K:

```
EX(0...17)
```

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator privileges to reconfigure any set of communicating expanders (SOCX) in the system.

Domain administrators or configurators can reconfigure only the SOCX assigned to the domain(s) in which they have privileges.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Setting All Buses on All Active Domains to Use CS0

This example displays setbus output when there are inactive boards powered on in one or more of the domains.

```
sc0:sms-user:> setbus -c CS0
The following boards are powered on but are not active in a domain:
SB13
IO9 assigned to domain J
IO16 assigned to domain Q
SB17
These boards will be reset, and any attach-ready state will be lost.
```

EXAMPLE 2 Setting All Buses on All Active Domains to Use Both CSBs

```
sc0:sms-user:> setbus -c CS0,CS1
```

EXAMPLE 3 Setting Address Bus on All Domains to Use CS0

```
sc0:sms-user:> setbus -c CS0 -b a
```

EXAMPLE 4 Setting Address and Data Buses on Active EX1 to Use CS1

```
sc0:sms-user:> setbus -c CS1 -b ad EX1
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See `attributes(5)` for a description of the following attribute.

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO

`showbus(1M)`

NAME

setcsn - set the chassis serial number for a Sun Fire high-end system

SYNOPSIS

setcsn `-c` *chassis_serial_number*

setcsn `-h`

DESCRIPTION

`setcsn`(1M) enables you to record the chassis serial number that identifies a Sun Fire high-end system. The chassis serial number is printed on a label located on the front of the system chassis, near the bottom center.

If you are upgrading to SMS 1.6 from an earlier SMS version, you must run the `setcsn` command to record the chassis serial number. The chassis serial number can be recorded only once. This command verifies the specified serial number, and if the number was previously recorded, this command will not allow you to set a different serial number.

Run this command on the main system controller. The chassis serial number is maintained as part of the platform configuration information.

Note – Sun manufacturing records the chassis serial number of Sun Fire high-end systems that ship with SMS 1.6 installed. Run the `showplatform`(1M) command to determine whether a chassis serial number was previously assigned to your system.

OPTIONS

The following options are supported:

`-c` *chassis_serial_number*	Specifies the chassis serial number for a Sun Fire high-end system. You obtain this number from a label on the front of the system chassis, near the bottom center. The serial number can be a maximum of 20 alphanumeric characters.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator or platform service privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Setting the Chassis Serial Number

```
sc0:sms-user:> setcsn -c 352A00008
```

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`2` Usage error.

`4` Permission error.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes:

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`pcd(1M)`, `showplatform(1M)`

NAME

setdatasync - modify the data propagation list used in data synchronization

SYNOPSIS

setdatasync [`-i interval`] `schedule` *filename*

setdatasync `cancel` *filename*

setdatasync `push` *filename*

setdatasync `backup`

setdatasync `-h`

DESCRIPTION

`setdatasync` enables you to specify a user-created file to be added to or removed from the data propagation list. This list identifies the files to be copied from the main to the spare system controller (SC) as part of data synchronization for automatic failover. The specified user file and the directory in which it resides must have read and write permissions for the user on both SCs.

The data synchronization process checks the user-created files on the main SC for any changes. If the user-created files on the main SC have changed since the last propagation, they are repropagated to the spare SC. By default, the data synchronization process checks a specified file every 60 minutes; however, you can use `setdatasync` to indicate how often a user file should be checked for modifications.

Note – After a file is propagated from the main SC to the spare, the file is repropagated to the spare only when the file on the main SC is updated. If you remove a propagated file from the spare SC, that file is not automatically repropagated until the corresponding file on the main SC has been changed.

You can also use `setdatasync` to do the following:

- Propagate a specified file to the spare SC without adding the file to the data propagation list.
- Resynchronize the SC configuration files on the main and spare SCs.

OPTIONS The following options are supported:

`backup` Forces re-synchronization after something other than SMS creates undesired changes to SMS files on the spare SC.

`setdatasync backup` re-propagates every SMS configuration, data and log file. `fomd` does this automatically. Using `setdatasync backup` can slow down automatic `fomd` file propagation.

The time required to execute `setdatasync backup` is approximately proportional to the number of files being transferred. Other factors that can affect the speed of file transfer include: the average size of files being transferred, the amount of memory available on the SCs, the load (CPU cycles and disk traffic) on the SCs, and whether the I2 network is functioning. For more information, refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

Use `setdatasync backup` only in the following situations.

- SMS was re-installed on the spare SC while SMS was running on the main SC.

Note – SMS users groups must be setup correctly on the spare before running `setdatasync backup`

- SMS files were deleted from the spare SC while SMS was running on the main SC.
- SMS files were overwritten or corrupted on the spare SC (regardless of whether SMS was running or not).

`cancel` *filename* Removes (cancels) the specified file from the data propagation list, which means the specified file will no longer be propagated to the spare SC. This option does not actually remove the specified file from the spare SC. The file name must contain the absolute path and cannot be a symbolic link to another file.

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-i` *interval*	Indicates how often the specified file should be checked for modifications. The default interval is 60 minutes. The interval can range from 1 to 1440 minutes (24 hours).
`push` *filename*	Propagates (pushes) the specified file to the spare SC without adding it to the data propagation list. The file name must contain the absolute path and cannot be a symbolic link to another file.
`schedule` *filename*	Adds the specified file to the data propagation list. The file name must contain the absolute path and cannot be a symbolic link to another file. During data synchronization, the file is propagated to the same absolute path on the spare SC.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or domain configurator privileges to run this command.

Refer to Chapter 2, "SMS Security Options and Administrative Privileges" in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Propagating a User File From Main to Spare Every 30 Minutes

The path to the user-specified file must be an absolute path and cannot contain a symbolic link.

```
sc0:sms-user:> setdatasync -i 30 schedule /path/filename
```

EXAMPLE 2 Removing File Name From Data Propagation List

The path to the user-specified file must be an absolute path and cannot contain a symbolic link.

```
sc0:sms-user:> setdatasync cancel /path/filename
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`showdatasync(1M)`, `smsbackup(1M)`

NAME

setdate - set the date and time for the system controller (SC) or a domain

SYNOPSIS

setdate [-d *domain_indicator*] [-u] [-q] *[mmdd]HHMM | mmddHHMM[cc]yy[.SS]*

setdate -h

DESCRIPTION

setdate(1M) enables the SC platform administrator to set the SC or optionally set a domain date and time values. Enables domain administrators to set the date and time values for their domains. After the date and time are set setdate(1M) displays the current date and time.

OPTIONS

The following options are supported:

-d *domain_indicator*	Specifies the domain using one of the following: *domain_id* –ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using addtag(1M). Sets the domain time of day (TOD) when the domain keyswitch is in the OFF or STANDBY position. This option is not the primary use of setdate. Normally, setdate is used without this option to set the SC TOD.
-h	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to -h is ignored.
-q	Does not display current date and time after you set the new value.
-u	Interprets and displays the time using Greenwich Mean Time (GMT). The default is the local time zone.

OPERANDS

The following operands are supported:

[mmdd]HHMM[.SS]	Date and time format. mm is the month (1–12), dd is the day of the month (1–31), HH is the hour (0–23), MM is the minute (0–59), and SS is the second (0–59).
mmddHHMM[cc]yy[.SS]	Date and time format. mm is the month (1–12), dd is the day of the month (1–31), HH is the hour (0–23), MM is the minute (0–59), cc is century minus one, yy is the two-digit year, and SS is the second (0–59).

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform or domain administrator privileges to run this command. If you have domain administrator privileges you can run this command only for your domain.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Setting the Local Date in Pacific Standard Time

```
sc0:sms-user:> setdate 020210302002.00
System Controller: Sat Feb 2 10:30:00 PST 2002
```

EXAMPLE 2 Setting the Date Using GMT

```
sc0:sms-user:> setdate -u 020218302002.00
System Controller: Sat Feb 2 18:30:00 GMT 2002
```

EXAMPLE 3 Setting the Local Time in Pacific Standard Time for Domain A

```
sc0:sms-user:> setdate -d a 020210302002.00
Domain a: Sat Feb 2 10:30:00 PST 2002
```

EXAMPLE 4 Setting the Date for Domain A Using GMT

```
sc0:sms-user:> setdate -d a -u 020218302002.00
Domain a: Sat Feb 2 18:30:00 GMT 2002
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `setkeyswitch(1M)`, `showdate(1M)`

NAME

setdefaults - remove all instances of a previously active domain and reset Capacity on Demand (COD) information

SYNOPSIS

setdefaults [`-d` *domain_indicator* [`-p`]] [`-y`|`-n`]

setdefaults `-h`

DESCRIPTION

`setdefaults`(1M) removes all SMS instances of a previously active domain and Capacity on Demand (COD) information. A domain instance includes all `pcd` entries *except* network information; all message, console, and syslog log files; and, optionally, all NVRAM and boot parameters. `pcd` entries and NVRAM and boot parameters are returned to system default settings. `IDPROM` data is not affected.

The COD information that is removed includes instant access CPUs (headroom) and reserved domain COD right-to-use (RTU) licenses. Only the platform administrator can reset the headroom value. The platform or domain administrator can reset reserved domain COD RTU licenses.

OPTIONS

The following options are supported:

`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* –ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-n`	Automatically answers no to all prompts.
`-p`	Preserves NVRAM and boot parameter data. By default, you are asked whether or not you want to remove the NVRAM and boot parameter data. If the `-p` option is used, you are not prompted and the data is preserved.
`-y`	Automatically answers yes to all prompts.

EXTENDED DESCRIPTION

If the `-d` *domain_indicator* is specified, the `setdefaults` command resets domain information. The domain cannot be active, and the virtual keyswitch must be set to off. Otherwise, the `setdefaults` command exits with an error. Platform administrators can reset the access control list (ACL) and the domain COD RTU licenses, but the domain administrator cannot.

If you have platform administrator privileges and you do not specify the `-d` *domain_indicator*, the `setdefaults` command resets the COD headroom, provided that the reset does not cause any COD RTU license violations.

Group Privileges Required

You must have platform administrator or domain administrator privileges for the specified domain to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Setting Defaults on Domain A With Domain, NVRAM, and Boot Parameter Prompts

```
sc0:sms-user:> setdefaults -d a
Are you sure you want to remove domain info? y
Do you want to remove NVRAM and boot parameter data? y
```

EXAMPLE 2 Setting Defaults on Domain A Without Prompts, Saving NVRAM and Boot Parameter Data

```
sc0:sms-user:> setdefaults -d a -p -y
```

EXAMPLE 3 Setting Defaults on Domain A Without Prompts and Without Saving NVRAM and Boot Parameter Data

```
sc0:sms-user:> setdefaults -d a  -y
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	An invalid domain was specified.
2	An invalid option was entered.
3	No domain, or more than one domain, was specified.
4	The user has invalid permission.
5	The keyswitch is in an invalid position.
6	The domain is currently active.
7	An error occurred talking to the `pcd`.
8	An error occurred talking to the `mld`.
9	An error occurred talking to the `osd`.

10	An internal error occurred.
11	The user canceled the operation.
12	An error occurred talking to the `codd`.

FILES

The following files are affected by this command:

`/var/opt/SUNWSMS/.pcd/domain_info`	Domain `pcd` information file.
`/var/opt/SUNWSMS/.pcd/sysboard_info`	Platform `pcd` information file.
`/var/opt/SUNWSMS/adm/`*domain_id*`/console`	Domain console log file. Up to ten console files are stored on the system at any one time— `console.0` through `console.9`.
`/var/opt/SUNWSMS/adm/`*domain_id*`/messages`	Domain log file. Up to ten message files are stored on the system at any one time— `message.0` through `message.9`.
`/var/opt/SUNWSMS/adm/`*domain_id*`/syslog`	Domain syslog file. Up to ten syslog files are stored on the system at any one time— `syslog.0` through `syslog.9`.
`/var/opt/SUNWSMS/data/`*domain_id*`/bootparamdata`	Domain boot parameter information file.
`/var/opt/SUNWSMS/data/`*domain_id*`/nvramdata`	Domain `nvram` information file.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes:

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO | `addtag(1M)`, `mld(1M)`, `osd(1M)`, `pcd(1M)`, `setobpparams(1M)`, `showobpparams(1M)`

NAME

setfailover - modify the state of the system controller (SC) failover mechanism

SYNOPSIS

setfailover [`-q] [-y|-n] on|off|force`

setfailover `-h`

DESCRIPTION

`setfailover`(1M) provides the ability to modify the state of failover for the SC failover mechanisms.

OPTIONS

The following options are supported:

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-q` "Quiet" option. When used with the `on`, `off`, or `force` options, no output message is returned. When used with the `-y` or `-n` options and the `force` option, no confirmation message is returned. When used alone, the command defaults to the `-n` option.

`-y|-n` Used with the `force` option to bypass the confirmation message (see Example 3). Specifying `-y` allows the force action to continue without asking for confirmation. Specifying `-n` means that no forced failover occurs. Note that using the `-y` option with the `-q` option suppresses all prompts, including the spare clock warning prompt (see Example 4).

OPERANDS

The following operands are supported:

`on|off|force` The following are valid actions:

`force`	Forces a failover to the spare SC. The spare SC must be available. If the clock on the spare SC is bad, the command returns a warning that forcing the failover could cause the boards to domain stop (Dstop). See Example 4. Answering `yes` to this prompt forces the failover. Answering `no` cancels the failover. If the `force` option is used with both the `-q` and `-y` options and the clock on the spare SC is bad, then no prompts are displayed, including the spare clock warning. The failover is forced, and the Dstop might occur.
`off`	Disables the failover mechanism. This prevents a failover until the mechanism is reenabled.
`on`	Enables failover for systems that previously had failover disabled due to a failover or an operator request. `on` instructs the command to attempt to reenable failover only. If failover cannot be reenabled, subsequent use of the `showfailover` command indicates the current failure that prevented the enable.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

These commands produce no output when successful. To see the results of the commands, use the `showfailover` command. An error message appears if the action can not be performed.

EXAMPLE 1 Turning Failover On

```
sc0:sms-user:> setfailover on
```

EXAMPLE 2 Turning Failover Off

```
sc0:sms-user:> setfailover off
```

EXAMPLE 3 Forcing a Failover

```
sc0:sms-user:> setfailover force
Forcing failover. Do you want to continue (yes/no)? yes
```

EXAMPLE 4 Forcing a Failover With a Bad Spare Clock

```
sc0:sms-user:> setfailover force
Forcing failover. Do you want to continue (yes/no)? yes
The spare clock input on some boards might be bad.Forcing a failover now
is likely to cause the affected domains to domain stop (Dstop).
Do you want to continue (yes/no)? no
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`showfailover(1M)`

NAME

setkeyswitch - change the position of the virtual keyswitch

SYNOPSIS

setkeyswitch `-d` *domain_indicator* [`-q`] [`-y`|`-n`] *position* `-l` *level*

setkeyswitch `-h`

DESCRIPTION

`setkeyswitch`(1M) changes the position of the virtual keyswitch to the specified value. `setkeyswitch` is responsible for powering on or powering off boards and bringing up a domain. See the OPERANDS section for more information. Note that the new firmware is not active on system boards until `hpost` is run.

If the domain specified contains a board in the automatic system recovery (ASR) blacklist file, an error message is displayed, `setkeyswitch` skips power on of that board and `setkeyswitch` continues.

The state of each virtual keyswitch is maintained between power cycles of the system controller (SC) or physical power cycling of the power supplies by the `pcd`(1M). Use `showkeyswitch` to display the current position of a virtual keyswitch.

OPTIONS

The following options are supported.

Note – The `-y` and `-n` are optional arguments to the `setkeyswitch`(1M) command. If one of these optional arguments is not provided, `setkeyswitch` prompts you for confirmation when changing from the `on`, `diag`, or `secure` position to the `off` or `standby` position.

`-d` *domain_indicator*

Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag -– Name assigned to a domain using `addtag`(1M).

`-h`

Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-n`

Automatically answers no to all prompts. Prompts are displayed unless used with the `-q` option.

`-q`	Quiet. Suppresses all messages to `stdout` including prompts. When used alone `-q` defaults to the `-n` option for all prompts. When used with either the `-y` or the `-n` option, `-q` suppresses all user prompts and automatically answers with either 'y' or 'n' based on the option chosen.
`-y`	Automatically answers yes to all prompts. Prompts are displayed unless used with the `-q` option.
`-l` *level*	Allows you to specify the `hpost` level to be used at system startup. This option lets you override the default `hpost` level. This value is passed to any `hpost` (1M) process invoked by the `setkeyswitch` command. Note that `hpost` is not intended to be invoked by users, but that the value of *level* will appear in the `hpost` log. Use this option only with the guidance of Sun Service.

OPERANDS This section describes operands for the position option. The following operands are supported:

position Valid *position* operands are:

`on` From the `off` or `standby` position, `on` powers on all boards assigned to the domain (if not already powered on). Then the domain is brought up. From the `diag` position, `on` is nothing more than a position change and does not affect a running domain. From the `secure` position, `on` restores write permission to the domain.

`standby` From the `on`, `diag`, or `secure` position, `standby` optionally displays a confirmation prompt. If you answer 'yes' then it determines if the domain is in a suitable state to be reset and deconfigured (for example, the OS is not running). If the domain is in a suitable state to be reset and deconfigured, then `setkeyswitch` resets and deconfigures all boards assigned to the domain. If not, then prior to the reset and deconfiguration, `setkeyswitch` gracefully shuts down the domain. From the `off` position, `standby` powers on all boards assigned to the domain (if not already powered on).

`off` — From the `on`, `diag`, or `secure` position, `off` optionally displays a confirmation prompt. If you answer 'yes' then it determines if the domain is in a suitable state to be powered off (for example, the OS is not running). If the domain is in a suitable state to be powered off, then `setkeyswitch` powers off all boards assigned to the domain. If not, then `setkeyswitch` aborts and logs a message to the domain log. From the `standby` position, `off` powers off all the boards in the domain.

`diag` — From the `off` or `standby` position, `diag` powers on all boards assigned to the domain (if not already powered on). Then the domain is brought up just as in the `on` position, except that `POST` is invoked with verbosity and diagnostic levels set to, at least, their defaults. From the `on` position, `diag` results in nothing more than a position change, but upon automatic system recovery (ASR) of the domain, `POST` is invoked with verbosity and diagnostic levels set to, at least, their defaults. From the `secure` position, `diag` restores write permission to the domain and upon ASR, `post` is invoked with verbosity and diagnostic levels set to, at least, their defaults. For more information on ASR, refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

`secure` — From the `off` or `standby` position, `secure` powers on all boards assigned to the domain (if not already powered on). Then the domain is brought up just as in the `on` position, except that the `secure` position removes write permission to the domain; for example, `flashupdate`s and `reset`s will not work. From the `on` position, `secure` removes write permission to the domain (as previously described). From the `diag` position, `secure` removes write permission to the domain (as previously described).

`off`	From the `on`, `diag`, or `secure` position, `off` optionally displays a confirmation prompt. If you answer 'yes' then it determines if the domain is in a suitable state to be powered off (for example, the OS is not running). If the domain is in a suitable state to be powered off, then `setkeyswitch` powers off all boards assigned to the domain. If not, then `setkeyswitch` aborts and logs a message to the domain log. From the `standby` position, `off` powers off all the boards in the domain.
`diag`	From the `off` or `standby` position, `diag` powers on all boards assigned to the domain (if not already powered on). Then the domain is brought up just as in the `on` position, except that `POST` is invoked with verbosity and diagnostic levels set to, at least, their defaults. From the `on` position, `diag` results in nothing more than a position change, but upon automatic system recovery (ASR) of the domain, `POST` is invoked with verbosity and diagnostic levels set to, at least, their defaults. From the `secure` position, `diag` restores write permission to the domain and upon ASR, `post` is invoked with verbosity and diagnostic levels set to, at least, their defaults. For more information on ASR, refer to the *System Management Services (SMS) 1.6 Administrator Guide*.
`secure`	From the `off` or `standby` position, `secure` powers on all boards assigned to the domain (if not already powered on). Then the domain is brought up just as in the `on` position, except that the `secure` position removes write permission to the domain; for example, `flashupdate`s and `reset`s will not work. From the `on` position, `secure` removes write permission to the domain (as previously described). From the `diag` position, `secure` removes write permission to the domain (as previously described).

EXTENDED DESCRIPTION

Group Privileges Required

You must have domain administrator privileges for the specified domain to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Setting Keyswitch on Domain A On

```
sc0:sms-user:> setkeyswitch -d A on
```

EXAMPLE 2 Using Keyswitch on a Domain Containing a Board in the ASR Blacklist File

```
sc0:sms-user:> setkeyswitch -d A on
SB0 is in the ASR Blacklist.
```

EXAMPLE 3 Changing the `hpost` Level

```
sc0:sms-user:> setkeyswitch -d A -l 7 on
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following file is used by this command:

`/etc/opt/SUNWSMS/config/asr/blacklist`	List of components excluded by `esmd`.

Note – This file is created and used internally and should *not* be edited manually.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `esmd(1M)`, `flashupdate(1M)`, `pcd(1M)`, `reset(1M)`, `showkeyswitch(1M)`

NAME

setobpparams - set up OpenBoot PROM variables for a domain

SYNOPSIS

setobpparams `-d` *domain_indicator param=value...*

setobpparams `-h`

DESCRIPTION

`setobpparams`(1M) enables a domain administrator to set the virtual `NVRAM` and `REBOOT` variables passed to OpenBoot PROM by `setkeyswitch`(1M). The `-d` option with *domain_id* or a *domain_tag* is required. You must reboot the domain in order for any changes to take effect.

This command is intended for error recovery and not for routine system administration. For more information refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

OPTIONS

The following options are supported:

`-d` *domain_indicator* — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

OPERANDS The following operands are supported:

param=value NVRAM and REBOOT variable values for OpenBoot PROM. Valid variables are:

- diag-switch?

  Default value= false

  When set to false, the default boot device is specified by boot-device and the default boot file by boot-file.

  When set to true, OpenBoot PROM runs in diagnostic mode and you need to set either diag-device or diag-file to specify the correct default boot device or file. These default boot device and file settings cannot be set using setobpparams. Use setenv(1) in OpenBoot PROM.

- auto-boot?

  Default value= false

  When set to true, the domain boots automatically after power-on or reset-all. The boot device and boot file used are based on the settings for diag-switch (see above). Neither boot-device nor boot-file can be set using setobpparams. In the event the OK prompt is unavailable, such as a repeated panic, use setobpparams to set auto-boot? to false.

  When the auto-boot? variable is set to false using setobpparams, the reboot variables are invalidated, the system will not boot automatically and will stop in OpenBoot PROM where new NVRAM variables can be set.

- fcode-debug?

  Default value= false

  When set to true, this variable includes name fields for plug-in device FCodes.

- use-nvramrc?

  Default value= false

  When set to true, this variable executes commands in NVRAMRC during system start-up.

- security-mode

  Default value= none

  Firmware security level.

Valid variable values for all but security mode are:

- `true`
- `false`

Valid variable values for security mode are:

- `none`
- `command`
- `full`

where:

none - Means that no password is required (default)

command - Means that all commands except `boot`(1M) and `go` require the password.

full - Means that all commands except for `go` require the password.

Note – It is important to remember your security password and to set the security password before setting the security mode. If you forget this password, you cannot use your system; you must call your vendor's customer support service to make your system bootable again. For more information on `security-mode` and other OpenBoot PROM variables, see the *OpenBoot 4.x Command Reference Manual*.

Note – Most, but not all, shells require single quotes around the variable values to prevent the question mark from being treated as a special character. See Example 1.

EXTENDED DESCRIPTION

Group Privileges Required

Domain administrator or configurator privileges for the specified domain are required.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Setting OpenBoot PROM Variable `diag-switch` to On for Domain A

```
sc0:sms-user:> setobpparams -d a 'diag-switch?=true'
```

EXAMPLE 2 Setting OpenBoot PROM Variable `security-mode` to Full for Domain A

```
sc0:sms-user:> setobpparams -d a security-mode=full
```

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `setkeyswitch(1M)`, `showobpparams(1M)`

NAME

setpcimode - Forces a PCI-X slot to run in PCI mode.

SYNOPSIS

setpcimode `pci_only | normal` *slot_spec...*

setpcimode `-h`

DESCRIPTION

This command changes the settings for the PCI-X slots on a V2HPCIX I/O board in your server. Use this command to force a PCI-X slot to run in PCI mode, or to return the slot to PCI-X mode.

The slot must be in the disconnected state before you use this command; otherwise, `setpcimode` returns an error. After you use the `setpcimode` command, use the `connect` command to return the slot to a connected state.

When the domain is in HPOST or OBP, the PCI-X adapters are connected. Running `setpcimode` while the domain is in either state will return an error. To use the `setpcimode` command when the domain is in HPOST or OBP, either disconnect the card when Solaris boots, or set the keyswitch position of the domain to `standby`.

If you specify an I/O board that is not a V2HPCIX board, the command returns an error.

Note – You can only run this command from the main SC.

OPTIONS

The following options are supported.

`-pci_only` Forces the PCI-X slot to run in PCI mode.

`-normal` Returns the PCI-X slot to PCI-X mode..

OPERANDS

The following operands are supported:

slot_spec Denotes the slot to be changed. *slot_spec* takes the form `IO#/slot_name`, where:

- `IO#` # is an integer from 0 to 17, corresponding to slots 0 through 17 on the board.
- `slot_name` The slot name: `c3v0`, `c3v1`, or `c3v2`.

EXTENDED DESCRIPTION

Group Privileges Required

You must have domain or platform administrator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

ERROR MESSAGES

`setpcimode` returns error messages when certain error conditions are encountered. The following table lists these error messages and the conditions that cause them to be displayed.

`ERROR: Invalid location: %s`

The location specified in the command line argument is invalid. This error can occur if the location specified is not a PCI-X slot on a V2HPCIX I/O board, or if the location string (`%s`) is not in an acceptable format.

`ERROR: PCI mode cannot be set for %s`

The `setpcimode` command is not applicable to the location specified in the command line argument. This error is only returned when the user specifies slot `c5v0` as the location.

`ERROR: Slot is not on a V2HPCIX board: %s`

This error is returned when the location specified in the command line argument does not belong to a V2HPCIX board.

`ERROR: unable to get domain board info for %s`

This error appears when `setpcimode` is unable to retrieve PCI slot status for the specified location, or when the command fails to find which domain contains the I/O board corresponding to the specified slot. This error should not occur if SMS is functioning normally.

`ERROR: unable to set PCI mode setting for %s`

This error appears when `setpcimode` is unable to set PCI slot status for the specified location. This error should not occur if SMS is functioning normally.

`ERROR: slot is still connected: %s`

This error appears when the `setpcimode` command is issued for a PCI-X slot that still has a connected cassette.

To fix this problem, issue a `cfgadm -c disconnect` command on the domain side for the specified slot, and then retry the `setpcimode` command.

`ERROR: You do not have the appropriate privilege to execute this command for %s`

This error appears when you do not have the appropriate domain/platform privileges to issue the `setpcimode` command for the specified location.

`ERROR: Invalid state: %s`

This error appears if you specified an unrecognizable option (other than `pci_only` or `normal`) for the `setpcimode` command.

`ERROR: an error occurred while processing %s`

This error appears if an internal error occurred for the command. Some examples of errors include running out of memory, or other types of unexpected errors. This error should not occur if the SC is functioning properly.

EXAMPLES

EXAMPLE 1 Setting a PCI-X Slot to Run in PCI Mode

```
sc0:sms-user:> setpcimode -pci_only IO17/C3V2 IO3/C3V1
IO17/C3V2: pci_only
IO3/C3V1: pci_only
```

EXAMPLE 2 Setting a PCI-X Slot Back to PCI-X Mode from PCI Mode

```
sc0:sms-user:> setpcimode -normal IO17/C3V0
IO17/C3V0: normal
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

ATTRIBUTES

See `attributes(5)` for a description of the following attribute:

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`esmd(1M)`, `pcd(1M)`, `showpcimode(1M)`

NAME

setupplatform - set up the available component list and Capacity on Demand (COD) resources used for domains

SYNOPSIS

setupplatform `-p available` [`-d` *domain_indicator* [`-a`|`-r`] *location*...]

setupplatform `-p cod` [*headroom*|`-d` *domain_indicator* *domainRTU*]

setupplatform [`-d` *domain_indicator* -]

setupplatform `-h`

DESCRIPTION

`setupplatform`(1M) sets up the available component list and COD resources to be used for domains. If a *domain_id* or *domain_tag* is specified, a list of boards *must* be specified. An empty board list can be specified as a dash(-). When no *domain_id* or *domain_tag* is specified, current values are displayed in the square brackets ([]) at the command prompt. If no value is specified for a parameter, it retains its current value.

OPTIONS

The following options are supported:

`-a`	Adds the slot(s) to the available component list for the domain.
`-d` *domain_indicator*	Specifies the domain using one of the following:
	domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.
	domain_tag – Name assigned to a domain using `addtag`(1M).
`-h`	Help. Displays usage descriptions.
	Note – Use alone. Any option specified in addition to `-h` is ignored.
`-p available`	Modifies the domain available component list.
`-p cod`	Assigns COD resources.
`-r`	Removes the slots from the available component list for the domain.
`-`	Clears the entire available component list.

OPERANDS

The following operands are supported:

domainRTU — Number of COD right-to-use (RTU) licenses to be reserved for a domain.

headroom — Amount of headroom (processors) to be enabled.

location — Board location separated by a space.

The following *location* forms are accepted:

Sun Fire 12K/E20K:

```
SB(0...8)
IO(0...8)
```

Sun Fire 15K/E25K:

```
SB(0...17)
IO(0...17)
```

EXTENDED DESCRIPTION

If you run the `setupplatform` command without specifying any options, the command prompts you for platform and COD information. You are asked to specify the available component list for all 18 domains, the amount of COD headroom to be used, and the number of COD RTU licenses to be reserved for your domains. When you are prompted for COD information, the maximum values allowed are displayed within parentheses () and default values are displayed within brackets [].

Use the `-p cod` option with the `setupplatform` command to enable COD headroom (processors to be used on demand). Use the `-d` *domain_indicator* with the `-p cod` option to specify the number of domain COD RTU licenses to be reserved.

You can reset the domain available component list and COD RTU reservation values by running the `setupplatform` command with a domain indicator and the '-' option.

Group Privileges Required

You must have platform administrator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

Use `showplatform`(1M) to display the available component list once you have run `setupplatform`.

EXAMPLE 1 Setting Up Available Component List for All Domains

```
sc0:sms-user:>  setupplatform
Available component list for domain domainA [SB3 SB2 SB1 IO5 IO4 IO3]?
-r SB1
Are you sure[no]:  (yes/no)? y
Available for domain DomainB [SB6 SB4 SB1 IO3 IO2 ]?  -
Are you sure[no]:  (yes/no)? y
Available for domain C [SB7 SB5 IO8 IO7]? -a SB17 SB16
Available for domain D [SB9 SB8 SB4 SB2 IO6 IO5 IO1]?
Available for domain E [SB0 IO0]?
Available for domain F []?
Available for domain G []?
Available for domain H []?
Available for domain I []?
Available for domain J []?
Available for domain K []?
Available for domain L []?
Available for domain M []?
Available for domain N []?
Available for domain O []?
Available for domain P []?
Available for domain Q []?
Available for domain R []?
PROC Headroom Quantiy (0 to disable, 8 MAX) [0]? 4
PROC RTUs reserved for domain A (10 MAX) [0]? 3
PROC RTUs reserved for domain B (7  MAX) [0]? 0
PROC RTUs reserved for domain C (7  MAX) [0]? 0
PROC RTUs reserved for domain D (7  MAX) [0]? 0
PROC RTUs reserved for domain E (7  MAX) [0]? 0
PROC RTUs reserved for domain F (7  MAX) [0]? 0
PROC RTUs reserved for domain G (7  MAX) [0]? 0
PROC RTUs reserved for domain H (7  MAX) [0]? 0
PROC RTUs reserved for domain I (7  MAX) [0]? 0
PROC RTUs reserved for domain J (7  MAX) [0]? 0
PROC RTUs reserved for domain K (7  MAX) [0]? 0
PROC RTUs reserved for domain L (7  MAX) [0]? 0
PROC RTUs reserved for domain M (7  MAX) [0]? 0
PROC RTUs reserved for domain N (7  MAX) [0]? 0
PROC RTUs reserved for domain O (7  MAX) [0]? 0
PROC RTUs reserved for domain P (7  MAX) [0]? 0
PROC RTUs reserved for domain Q (7  MAX) [0]? 0
PROC RTUs reserved for domain R (10 MAX) [3]? 3
```

```
sc0:sms-user:>  showplatform -p available
Available for domain DomainA:
 SB3 SB2
 IO0 IO4 IO3
Available for domain DomainB:
 None
 None
Available for domain DomainC:
 SB1 SB6 SB7 SB5
 IO8 IO7
Available for domain D:
 SB9 SB8 SB4
 IO6 IO5 IO1
Available for domain E:
 SB0
 IO0
Available for domain DomainF:
 None
 None
Available for domain DomainG:
 None
 None
Available for domain DomainH:
 None
 None
Available for domain I:
 None
 None
Available for domain J:
 None
 None
Available for domain DomainK:
 None
 None
Available for domain L:
 None
 None
Available for domain M:
 None
 None
Available for domain N:
 None
 None
Available for domain O:
 None
 None
Available for domain P:
 None
 None
Available for domain Q:
 None
 None
Available for domain R:
 None
 None
```

EXAMPLE 2 Setting Up Available Component List for Domain `engB` to Boards at SB0, IO1,

and IO2

```
sc0:sms-user:> setupplatform -p available -d engB SB0 IO1 IO2
```

EXAMPLE 3 Clearing All Boards in `engB` Available Component List and Reserved COD RTUs

```
sc0:sms-user:> setupplatform -d engB -
```

EXAMPLE 4 Adding Boards at SB0 and IO2 to `engB` Available Component List

```
sc0:sms-user:> setupplatform -p available engB -a SB0 IO2
```

EXAMPLE 5 Removing Boards at SB3 and IO3 From `engB` Available Component List

```
sc0:sms-user:> setupplatform -p available -d engB -r SB3 IO3
```

EXAMPLE 6 Setting COD CPU Headroom Quantity and Reserve Domain COD RTU Licenses

```
sc0:sms-user:> setupplatform -p cod
PROC Headroom Quantity (0 to disable, 8 MAX) [0]? 4
PROC RTUs reserved for domain A (10 MAX) [0]? 3
PROC RTUs reserved for domain B (7  MAX) [0]? 0
PROC RTUs reserved for domain C (9  MAX) [2]? 0
.
.
.
PROC RTUs reserved for domain R (7  MAX) [0]? 0
```

EXAMPLE 7 Set the COD Headroom CPUs to 8

```
sc0:sms-user:> setupplatform -p cod 8
```

EXAMPLE 8 Set the number of COD RTUs for Domain `engB` to 6

```
sc0:sms-user:> setupplatform -p cod -d engB 6
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An internal error occurred. For further information, see `/var/opt/SUNWSMS/adm/platform/messages`.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `showplatform(1M)`

NAME

showboards - show the assignment information and status of the boards

SYNOPSIS

showboards [`-d` *domain_indicator*] [`-v`]

showboards [`-d` *domain_indicator*] `-c`

showboards `-h`

DESCRIPTION

`showboards`(1M) displays board assignments and board status, including the clock source and status for all boards . If *domain_id* or *domain_tag* is specified, this command displays which boards are assigned or available to the given domain. The information displayed also indicates whether a board is a Capacity on Demand (COD) board.

If the `-v` option is used, `showboards` displays all components, including *domain configuration units* (DCUs) such as `CPUs`, `MCPUs`, `HPCI`, `HPCI+s`, and `WPCI`; as well as the system controllers (SCs) and other components that are not DCUs.

OPTIONS

The following options are supported:

`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-c`	Clock source. Displays the clock source and status for all system, expander, I/O, and centerplane support boards and the SCs. See Example 6. If a domain is specified (with the -d option), the -c option displays the clock information only for the system and I/O boards accessible by that domain.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-v`	Verbose. Displays all components, including DCUs.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, platform operator, platform service privileges or domain administrator, or domain configurator privileges for the specified domain to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

Platform administrator privileges:

- If no options are specified, `showboards` displays all components including those DCUs that are assigned or available.
- If *domain_id* or *domain_tag* is specified, `showboards` displays information on DCUs that are assigned and available to that domain. DCUs assigned to other domains are not displayed.
- If the `-v` option is provided, `showboards` displays information on all assigned or available DCUs. In addition, `showboards` displays information on all other components.
- If *domain_id* or *domain_tag* and the `-v` option are specified, `showboards` displays information on DCUs that are assigned or available to that domain. In addition, `showboards` displays information on all other components. DCUs assigned to other domains are not displayed.

Domain administrator/configurator privileges:

- If no options are specified, `showboards` displays all boards for all domains for which you have privileges, including those DCUs that are assigned or available.
- If *domain_id* or *domain_tag* is specified, `showboards` displays information on DCUs that are assigned or available to that domain. DCUs assigned to other domains are not displayed. Available DCUs are those boards that are in the domain's available component list. See `setupplatform`(1M) and `showplatform`(1M). You must have domain administrator or configurator privileges for the specified domain.
- The `-v` option is not available to this user.

Status Fields Displayed

This section describes status information displayed in the `showboards` command output.

The Pwr field contains one of five measurements:

`On`	Full voltage detected.
`Off`	No voltage detected.
`Min`	Some voltage detected.
`Unk`	Unknown. Unable to determine board power on state.
—	The slot is empty, so power state is not applicable.

`Min` does not imply that the board can be used at this point, but only that some power was detected on the board. It should not be used until it has been powered on. Conversely, it should not be removed from the system before being powered off.

The Board Status field contains one of four values:

`Active`	The board is assigned to a domain and has passed POST.
`Assigned`	The board is assigned to a domain.
`Available`	The board is available to be assigned to a domain.
—	Domain assignment or activity is not applicable for this board.

The Test status field reflects the recorded entry in the PCD database from the last POST run. The Test Status field contains one of six values:

`Passed`	The board passed POST.
`Degraded`	The board is in a degraded mode.
`Failed`	The board failed POST. Contact your Sun Service representative, who can determine whether or not the board needs to be replaced.
`iPOST`	The board is in POST.
`Unknown`	The board has not been tested.
—	The test status for this board is unavailable.

The Domain field contains one of four values:

domain_id	ID for a domain.
domain_tag	Name assigned to a domain using `addtag`(1M).
`Isolated`	The board is not assigned to any domain.
—	Domain assignment is not applicable for this board.

Status Fields Displayed by the -c option

This section describes status information displayed by the `showboards -c` output.

The Clock Signal Status field provides three indicators:

`Good`	A good clock signal is detected by the hardware.
`Failed`	A bad clock signal has been detected by the hardware.
`Unknown`	The clock status is unknown.

The Clock Source field provides three indicators:

`SC0 Clock`	System controller 0 is providing the clock signal.
`SC1 Clock`	System controller 1 is providing the clock signal.
`Unknown`	The current clock source is unknown.

The SC0 and SC1 Clock Status fields each provide the following indicators:

`Good`	System controller is on and is running.
Failed	System controller has failed.
`SC-Off`	System controller is off.
`No-SC`	There is no system controller in the Clock Status field.

The Auto-Clock Selection field provides three indicators:

`Enabled`	Hardware may automatically attempt to switch clock sources, if necessary.
`Disabled`	Hardware will not switch clock sources automatically.
`Unknown`	The clock select mode is unknown.

EXAMPLES

EXAMPLE 1 Listing boards for Platform Administrators on Sun Fire 15K/E25K System

```
sc0:sms-user:> showboards

Location  Pwr  Type         Board Status  Test Status   Domain
----      ---  ----         ------------  -----------   ------
SB0        On   CPU          Active        Passed        domainC
SB1        On   V3CPU        Active        Passed        A
SB2        On   V3CPU        Active        Passed        D
SB3        On   CPU          Active        Passed        engB
SB4        On   CPU          Active        Passed        engB
SB5        On   CPU          Active        Passed        engB
SB6        On   CPU          Active        Passed        A
SB7        On   CPU          Active        Passed        domainC
SB8        Off  CPU          Available     Unknown       Isolated
SB9        On   CPU          Active        Passed        dmnJ
SB10       Off  CPU          Available     Unknown       Isolated
SB11       Off  CPU          Available     Unknown       Isolated
SB12       Off  CPU          Assigned      Unknown       engB
SB13        -   Empty Slot   Available         -         Isolated
SB14       Off  CPU          Assigned      Failed        domainC
SB15       On   CPU          Active        Passed        P
SB16       On   CPU          Active        Passed        domainC
SB17        -   Empty Slot   Assigned          -         dmnR
IO0         -   Empty Slot   Available         -         Isolated
IO1        On   HPCI         Active        Passed        A
IO2        On   MCPU         Active        Passed        engB
IO3        On   MCPU         Active        Passed        domainC
IO4        On   HPCI+        Available     Degraded      domainC
IO5        Off  HPCI+        Assigned      Unknown       engB
IO6        On   HPCI         Active        Passed        A
IO7        On   HPCI         Active        Passed        dmnJ
IO8        On   WPCI         Active        Passed        Q
IO9        On   HPCI+        Assigned      iPOST         dmnJ
IO10       Off  HPCI         Assigned      Unknown       engB
IO11       Off  HPCI         Assigned      Failed        engB
IO12       Off  HPCI         Assigned      Unknown       engB
IO13        -   Empty Slot   Available         -         Isolated
IO14       Off  HPCI+        Available     Unknown       Isolated
IO15       On   HPCI         Active        Passed        P
IO16       On   HPCI         Assigned      Unknown       Q
IO17        -   Empty Slot   Assigned          -         dmnR
```

The following example illustrates `showboards` output if you have platform administrator privileges and specify a domain on a Sun Fire 15K/E25 system. The output does not include boards that are assigned to other domains.

EXAMPLE 2 Listing boards for Platform Administrators for Domain B

```
sc0:sms-user:> showboards -d b

Location  Pwr  Type         Board Status  Test Status   Domain
----      ---  ----         ------------  -----------   ------
SB3        On   CPU          Active        Passed        engB
SB4        On   CPU          Active        Passed        engB
SB5        On   CPU          Active        Passed        engB
SB8        Off  CPU          Available     Unknown       Isolated
SB10       Off  CPU          Available     Unknown       Isolated
SB11       Off  CPU          Available     Unknown       Isolated
SB12       Off  CPU          Assigned      Unknown       engB
SB13        -   Empty Slot   Available         -         Isolated
IO0         -   Empty Slot   Available         -         Isolated
IO2        On   MCPU         Active        Passed        engB
IO5        Off  HPCI+        Assigned      Unknown       engB
IO10       Off  HPCI         Assigned      Unknown       engB
IO11       Off  HPCI         Assigned      Failed        engB
IO12       On   HPCI         Assigned      Unknown       engB
IO13        -   Empty Slot   Available         -         Isolated
IO14       Off  HPCI+        Available     Unknown       Isolated
```

The following example illustrates `showboards` output if you have platform administrator privileges and use the `-v` option on a Sun Fire 15K/E25K system. The command shows all components. If a board is a COD board, it is specified in the Type of Board field.

EXAMPLE 3 Listing boards for Platform Administrators Using the `-v` Option

```
sc0:sms-user:> showboards -v

Location    Pwr    Type of Board   Board Status  Test Status   Domain

--------    ---    -------------   ------------  -----------   ------
SC0         On     SC              Main              -             -
SC1         On     SC              Spare             -             -
PS0         On     PS                -               -             -
PS1         On     PS                -               -             -
PS2         On     PS                -               -             -
PS3         On     PS                -               -             -
PS4         Off    PS                -               -             -
PS5         On     PS                -               -             -
FT0         On     FANTRAY           -               -             -
FT1         On     FANTRAY           -               -             -
FT2         On     FANTRAY           -               -             -
FT3         On     FANTRAY           -               -             -
FT4         On     FANTRAY           -               -             -
FT5         On     FANTRAY           -               -             -
FT6         On     FANTRAY           -               -             -
FT7         On     FANTRAY           -               -             -
CS0         On     CSB               -               -             -
CS1         On     CSB               -               -             -
EX0          -     EXB               -               -             -
EX1          -     EXB               -               -             -
EX2          -     EXB               -               -             -
EX3          -     EXB               -               -             -
EX4         On     EXB               -               -             -
```

```
EX5          -     EXB              -              -              -
EX6          On    EXB              -              -              -
EX7          -     EXB              -              -              -
EX8          -     EXB              -              -              -
EX9          -     EXB              -              -              -
EX10         -     EXB              -              -              -
EX11         -     EXB              -              -              -
EX12         Off   EXB              -              -              -
EX13         -     EXB              -              -              -
EX14         -     EXB              -              -              -
EX15         -     EXB              -              -              -
EX16         On    EXB              -              -              -
EX17         -     EXB              -              -              -
IO4/C3V0     On    C3V              -              -          domainC
IO4/C5V0     On    C5V              -              -          domainC
IO4/C3V1     On    C3V              -              -          domainC
IO4/C5V1     On    C5V              -              -          domainC
IO6/C3V0     On    C3V              -              -          A
IO6/C5V0     On    C5V              -              -          A
IO6/C3V1     On    C3V              -              -          A
IO6/C5V1     On    C5V              -              -          A
IO9/C3V0     On    C3V              -              -          dmnJ
IO9/C5V0     On    C3V              -              -          dmnJ
IO9/C3V1     On    C3V              -              -          dmnJ
IO9/C3V2     On    C3V              -              -          dmnJ
IO12/C3V0    Off   Unknown          -              -          engB
IO12/C5V0    Off   Unknown          -              -          engB
IO12/C3V1    Off   Unknown          -              -          engB
IO12/C5V1    Off   Unknown          -              -          engB
IO16/C3V0    On    C3V          Assigned       Unknown          Q
IO16/C5V0    On    C5V          Assigned       Unknown          Q
IO16/C3V1    On    C3V          Assigned       Unknown          Q
IO16/C5V1    On    C5V          Assigned       Unknown          Q
SB0          On    CPU          Active         Passed         domainC
SB1          On    V3CPU        Active         Passed         A
SB2          On    V3CPU        Active         Passed         D
SB3          On    CPU          Active         Passed         engB
SB4          On    CPU (COD)    Active         Passed         engB
SB5          On    CPU          Active         Passed         engB
SB6          On    CPU (COD)    Active         Passed         A
SB7          On    CPU          Active         Passed         domainC
SB8          Off   CPU          Available      Unknown        Isolated
SB9          On    CPU          Active         Passed         dmnJ
SB10         Off   CPU          Available      Unknown        Isolated
SB11         Off   CPU          Available      Unknown        Isolated
SB12         Off   CPU (COD)    Assigned       Unknown        engB
SB13         -     Empty Slot   Available          -          Isolated
SB14         Off   CPU          Assigned       Failed         domainC
SB15         On    CPU          Active         Passed         P
SB16         On    CPU (COD)    Active         Passed         domainC
SB17         -     Empty Slot   Assigned           -          dmnR
IO0          -     Empty Slot   Available          -          Isolated
IO1          On    HPCI         Active         Passed         A
IO2          On    MCPU         Active         Passed         engB
IO3          On    MCPU         Active         Passed         domainC
IO4          On    HPCI         Available      Degraded       domainC
IO5          Off   HPCI+        Assigned       Unknown        engB
IO6          On    HPCI         Active         Passed         A
IO7          On    HPCI         Active         Passed         dmnJ
IO8          On    wPCI         Active         Passed         Q
```

```
IO9          On     HPCI+           Assigned      iPOST         dmnJ
IO10         Off    HPCI            Assigned      Unknown       engB
IO11         Off    HPCI            Assigned      Failed        engB
IO12         Off    HPCI            Assigned      Unknown       engB
IO13          -     Empty Slot      Available         -         Isolated
IO14         Off    HPCI+           Available     Unknown       Isolated
IO15         On     HPCI            Active        Passes        Isolated
IO16         On     HPCI            Assigned      Unknown       Q
IO17          -     Empty Slot      Assigned          -         dmnR
```

The following example illustrates `showboards` output if you have domain privileges for domains B, J, and R on a Sun Fire 15K/E25K system. `showboards` displays information for those boards that are assigned or available to domains B, J, and R. Boards that are assigned to other domains or that do not appear in the available component list for domains B, J, or R are not displayed.

EXAMPLE 4 Listing boards for Domain Admin With Privileges on Domains B, J, and R

```
sc0:sms-user:> showboards

Location  Pwr   Type       Board Status  Test Status   Domain
----      ---   ----       ------------  -----------   ------
SB3        On   CPU         Active        Passed        engB
SB4        On   CPU         Active        Passed        engB
SB5        On   CPU         Active        Passed        engB
SB8        Off  CPU         Available     Unknown       Isolated
SB9        On   CPU         Active        Passed        dmnJ
SB10       Off  CPU         Available     Unknown       Isolated
SB11       Off  CPU         Available     Unknown       Isolated
SB12       Off  CPU         Assigned      Unknown       engB
SB13        -   Empty Slot  Available         -         Isolated
SB17        -   Empty Slot  Assigned          -         dmnR
IO0         -   Empty Slot  Available         -         Isolated
IO2        On   MCPU        Active        Passed        engB
IO5        Off  HPCI+        Assigned      Unknown       engB
IO7        On   HPCI        Active        Passed        dmnJ
IO9        On   HPCI+       Assigned      iPOST         dmnJ
IO10       Off  HPCI        Assigned      Unknown       engB
IO11       Off  HPCI        Assigned      Failed        engB
IO12       Off  HPCI        Assigned      Unknown       engB
IO13        -   Empty Slot  Available         -         Isolated
IO14       Off  HPCI+       Available     Unknown       Isolated
IO17        -   Empty Slot  Assigned          -         dmnR
```

In the following example, `showboards` displays output if you have domain privileges on domains B, J, and R on a Sun Fire 15K/E25K system. The command shows board information for those DCUs that are assigned or available to the specified domain. DCUs that are assigned to other domains or that do not appear in the specified domain's available component list are not displayed.

EXAMPLE 5 Listing boards for Domain Administrators for Domain B

```
sc0:sms-user:> showboards -d b

Location  Pwr   Type        Board Status  Test Status   Domain
----      ---   ----        ------------  -----------   ------
SB3        On   CPU          Active        Passed        engB
SB4        On   CPU          Active        Passed        engB
SB5        On   CPU          Active        Passed        engB
SB6         -   Empty Slot   Available         -         Isolated
SB8        Off  CPU          Available     Unknown       Isolated
SB8        Off  CPU          Available     Unknown       Isolated
SB10       Off  CPU          Available     Unknown       Isolated
SB11       Off  CPU          Available     Unknown       Isolated
SB12       Off  CPU          Assigned      Unknown       engB
SB13        -   Empty Slot   Available         -         Isolated
IO0         -   Empty Slot   Available         -         Isolated
IO2        On   MCPU         Active        Passed        engB
IO5        Off  HPCI+        Assigned      Unknown       engB
IO6         -   Empty Slot   Available         -         Isolated
IO10       Off  HPCI         Assigned      Unknown       engB
IO11       Off  HPCI         Assigned      Failed        engB
IO12       Off  HPCI         Assigned      Unknown       engB
IO13        -   Empty Slot   Available         -         Isolated
IO14       Off  HPCI+        Available     Unknown       Isolated
```

EXAMPLE 6 Displaying Clock Source and Status For All Boards

```
sc0:sms-user:> showboards -c

                  Current      SC0 Clock    SC1 Clock   Auto-Clock
Location  Pwr   Clock Source     Status       Status    Selection
--------  ---   ------------   ---------    ----------  -----------
CS0       On    SC0 Clock        Good         Good       Disabled
CS1       On    SCO Clock        Good         Good       Disabled
EX0       On    SC0 Clock        Good         Good       Disabled
.
.
EX15      Off       -              -            -          -
EX16      Off       -              -            -          -
EX17      On    SC0 Clock        Good         Good       Disabled
SB0       On    SC0 Clock        Good         Good       Disabled
SB1       On    SC0 Clock        Good         Good       Disabled
.
.
.
SB17      On    SC0 Clock        Good         Good       Disabled
IO0       On    SC0 Clock        Good         Good       Disabled
.
.
.
IO17      On    SC0 Clock        Good         Good       Disabled
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`1`	An invalid domain was specified.
`2`	An invalid command-line option was specified.
`3`	An incorrect number of domains was specified.
`4`	The user does not have valid privileges.
`5`	An internal error occurred.
`6`	An error occurred getting board information.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop
Interface Stability	Evolving
Command Output	Unstable

SEE ALSO

`addtag(1M)`, `setupplatform(1M)`, `showcodusage(1M)`, `showplatform(1M)`

NAME

showbus - display the bus configuration of expanders in active domains

SYNOPSIS

showbus [`-v`]

showbus `-h`

DESCRIPTION

`showbus`(1M) displays the bus configuration of expanders in active domains. This information defaults to displaying configuration by slot order EX0–EX17.

OPTIONS

The following options are supported:

`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-v`	Verbose. Displays all available command information. In addition to expander configuration, the domain, domain keyswitch position, and slot 0 and slot 1 board assignments are displayed.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, operator, or service privileges to display all sets of communicating expanders (SOCX) in the system.

Domain administrators or configurators can display only the SOCX assigned to the domain(s) in which they have privileges.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Display Bus Configuration for All Domains

This display is the default for platform administrators. A domain administrator/configurator must have privileges on all domains in order to obtain this display. Otherwise only those domains for which the user has privileges are displayed.

```
sc0:sms-user:> showbus

Location  Address    Data       Response    SOCX
-----------------------------------------------
EX0       CS0        CS1        CS0          0x0001
EX1       UNCONF     UNCONF     UNCONF       UNCONF
EX2       UNCONF     UNCONF     UNCONF       UNCONF
EX3       UNCONF     UNCONF     UNCONF       UNCONF
EX4       BOTH       BOTH       BOTH        0x14010
EX5       UNCONF     UNCONF     UNCONF       UNCONF
EX6       UNCONF     UNCONF     UNCONF       UNCONF
EX7       UNCONF     UNCONF     UNCONF       UNCONF
EX8       UNCONF     UNCONF     UNCONF       UNCONF
EX9       UNCONF     UNCONF     UNCONF       UNCONF
EX10      UNCONF     UNCONF     UNCONF       UNCONF
EX11      UNCONF     UNCONF     UNCONF       UNCONF
EX12      UNCONF     UNCONF     UNCONF       UNCONF
EX13      UNCONF     UNCONF     UNCONF       UNCONF
EX14      BOTH       BOTH       BOTH        0x14010
EX15      UNCONF     UNCONF     UNCONF       UNCONF
EX16      BOTH       BOTH       BOTH        0x14010
EX17      UNCONF     UNCONF     UNCONF       UNCONF
```

EXAMPLE 2 Display `Showbus` Information for All Domains Using `-v`

```
sc0:sms-user:> showbus -v
--------------
SOCX: 0x00002
--------------
Address:  BOTH
   Data:  BOTH
Response: BOTH
--------------
Domain:A - - ON/Running OBP
  Location:  EX1   SB1: On/active           IO1:  On/active
---------------
UNCONFIGURED
--------------
Domain: B - - OFF/Powered Off
  Location: EX12  SB12: Off/assigned       I012: Off/assigned
Domain: C - - OFF/Powered Off
  Location: EX10  SB10: Off/assigned       IO10: Off/assigned
--------------
UNASSIGNED
--------------
  Location:  EX0   SB0:  On/unassigned     IO0: Off/unassigned
  Location:  EX2   SB2: Off/unassigned     IO2:  On/unassigned
  Location:  EX6   SB6: Off/unassigned     IO6:  On/unassigned
  Location:  EX9   SB9:  On/unassigned     IO0: Off/unassigned
  Location: EX11  SB11: Off/unassigned    IO11: Off/unassigned
```

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`setbus(1M)`

NAME

showcmdsync - display the current command synchronization list

SYNOPSIS

showcmdsync [-v]

showcmdsync -h

DESCRIPTION

showcmdsync displays the command synchronization list to be used by the spare system controller (SC) to determine which commands or scripts need to be restarted after an SC failover.

The command synchronization list is displayed in the format *Descriptor, Identifier, Cmd*, where:

Descriptor	Specifies the command synchronization descriptor that represents a particular script.
Identifier	Identifies a marker point in the script from which the script can be resumed on the new main SC after an automatic failover occurs. The identifier -1 indicates that the script does not have any marker points.
Cmd	Indicates the name of the script to be restarted.

OPTIONS

The following options are supported:

-h	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to -h is ignored.
-v	Verbose. Displays all available command information.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or domain configurator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying a Command Synchronization List

```
sc0:sms-user:> showcmdsync
DESCRIPTOR     IDENTIFIER   CMD
         0             -1   c1 a1 a2
```

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`cancelcmdsync(1M)`, `initcmdsync(1M)`, `runcmdsync(1M)`, `savecmdsync(1M)`

NAME

showcodlicense - display the current Capacity on Demand (COD) right-to-use (RTU) licenses stored in the COD license database

SYNOPSIS

showcodlicense [`-r`] [`-v`]

showcodlicense `-h`

DESCRIPTION

`showcodlicense`(1M) displays COD license information stored in the COD license database.

OPTIONS

The following options are supported:

`-h`	Help. Displays usage descriptions.
	Note – Use alone. Any option specified in addition to `-h` is ignored.
`-r`	Displays the license information in the raw *license-signature* format, as stored in the COD license database.
`-v`	Verbose. Displays both the formatted license information and raw *license-signature* data.

EXTENDED DESCRIPTION

The `showcodlicense` command displays the following COD information:

Description	Type of resource (processor).
Lic Ver	Version number of the license, which is always set to 01.
Expiration	None.
Count	Number of right-to-use licenses granted for the given resource.
Status	GOOD, which indicates that the given resource is valid, or EXPIRED, which indicates that the resource license is no longer valid.
Cls	Not supported. Tier class value is always set to 1.
Tier Num	Not supported. Tier number value is always set to 1.
Req	Not supported. Required number of lower-tier licenses is always set to 0.

Group Privileges Required

You must have platform administrator or platform operator group privileges to run this command.

Refer to Chapter 2, "SMS Security Options and Administrative Privileges" in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

The following examples show the COD license information displayed:

EXAMPLE 1 Displaying Formatted License Data

```
sc0:sms-user:> showcodlicense

              Lic                                      Tier
 Description  Ver   Expiration  Count   Status  Cls  Num  Req
 -----------  ---  -----------  -----  -------  ---  ---  ---
 PROC          01         NONE     16     GOOD    1    1    0
```

EXAMPLE 2 Displaying Raw License Data

```
sc0:sms-user:> showcodlicense -r

01:5014936C37048:03001:0201010100:16:00000000:RKQhd8zKNnTwvxT5DJlZNQ
```

EXAMPLE 3 Displaying Formatted and Raw License Data

```
sc0:sms-user:> showcodlicense -v

              Lic                                      Tier
 Description  Ver   Expiration  Count   Status  Cls  Num  Req
 -----------  ---  -----------  -----  -------  ---  ---  ---
 PROC          01         NONE     16     GOOD    1    1    0
 01:5014936C37048:03001:0201010100:16:00000000:RKQhd8zKNnTwvxT5DJlZNQ
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	Invalid usage.
2	The user does not have valid privileges.
>2	An internal error occurred. For further information see `/var/opt/SUNWSMS/adm/platform/messages`.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addcodlicense(1M)`, `codd(1M)`, `deletecodlicense(1M)`, `showcodusage(1M)`

NAME

showcodusage - display the current usage statistics for Capacity on Demand (COD) resources

SYNOPSIS

showcodusage [`-v`] [`-p resource|domains`]

showcodusage `-h`

DESCRIPTION

`showcodusage`(1M) shows current information about COD right-to-use (RTU) licenses in use. By default, this command displays a summary of COD RTU licenses used and installed, along with the current state of each resource.

OPTIONS

The following options are supported:

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-p domains` Displays the license usage for each domain. The statistics reported include the number of COD RTU licenses used by the domain, resources assigned to the domain, and COD RTU licenses reserved for the domain.

`-p resource` Displays license usage by resource type.

`-v` Verbose. Displays all available COD usage information, including COD RTU license use for both the system and its domains.

EXTENDED DESCRIPTION

The `showcodusage -p resource` command displays the following COD usage information for the system:

Resource Identifies the type of COD resources available (processors).

In Use Specifies the number of COD CPUs currently used in the system.

Installed	Specifies the number of COD CPUs installed in the system.	
License	Specifies the number of COD RTU licenses installed.	
Status	Specifies one of the following COD attributes:	
	`OK`	Indicates that there are sufficient licenses for the COD CPUs in use. Also specifies the number of remaining COD resources available and the number of any instant access CPUs (headroom) available.
	`HEADROOM`	The number of instant access COD CPUs in use.
	`Violation`	Indicates a COD RTU license violation exists. Specifies the number of COD CPUs in use that exceeds the number of COD RTU licenses available. This situation can occur when you force the deletion of a COD RTU license key from the COD RTU license database, but the COD CPU associated with the license key is still in use.

The `showcodusage -p domain` command displays the following COD usage information for each domain:

Domain/Resource	Identifies COD RTU resource (processor) for each domain. An Unused processor is a COD CPU that has not yet been assigned to a domain.
In Use	Specifies the number of COD CPUs currently used in the domain.

Installed	Specifies the number of COD CPU resources installed in the domain.
Reserved	Specifies the number of COD RTU licenses allocated to the domain.
Status	Contains one of the following when the `-v` option is specified:
`Licensed`	The domain COD CPU has a COD RTU license and is in use.
`Unlicensed`	A COD RTU license for the domain COD CPU could not be obtained and it is not in use.
`Unused`	The COD CPU is not in use.

Group Privileges Required

You must have platform administrator, platform operator, or domain administrator group privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

Users with platform administrator privileges can view both resource and domain usage summaries. Users with domain administrator privileges can view only the domain usage summaries for which they have privileges, and a report of unused licenses.

EXAMPLE 1 Displaying COD Usage by Resource

```
sc0:sms-user:> showcodusage -p resource
Resource  In Use  Installed  Licensed  Status
--------  ------  ---------  --------  ------
PROC           4          4        16  OK: 12 available
```

EXAMPLE 2 Displaying COD Usage by Domain

```
sc0:sms-user:> showcodusage -p domains
Domain/Resource  In Use  Installed  Reserved
---------------  ------  ---------  --------
A - PROC              0          0         0
B - PROC              0          0         0
C - PROC              0          0         0
D - PROC              4          4         0
E - PROC              0          0         0
F - PROC              0          0         0
G - PROC              0          0         0
H - PROC              0          0         0
I - PROC              0          0         0
J - PROC              0          0         0
K - PROC              0          0         0
L - PROC              0          0         0
M - PROC              0          0         0
N - PROC              0          0         0
O - PROC              0          0         0
P - PROC              0          0         0
Q - PROC              0          0         0
R - PROC              0          0         0
Unused - PROC         0          0        12
```

EXAMPLE 3 Displaying COD Usage by Resource and Domain

```
sc0:sms-user:> showcodusage -v
Resource  In Use  Installed  Licensed  Status
--------  ------  ---------  --------  ------
PROC           4          4        16  OK: 12 available

Domain/Resource  In Use  Installed  Reserved  Status
---------------  ------  ---------  --------  ------
A - PROC              0          0         0
B - PROC              0          0         0
     SB6 - PROC       0          0
         SB6/P0                                Unused
         SB6/P1                                Unused
         SB6/P2                                Unused
         SB6/P3                                Unused
C - PROC              0          0         0
    SB12 - PROC       0          0
        SB12/P0                                Unused
        SB12/P1                                Unused
        SB12/P2                                Unused
        SB12/P3                                Unused
D - PROC              4          4         0
     SB4 - PROC       4          4
         SB4/P0                                Licensed
         SB4/P1                                Licensed
         SB4/P2                                Licensed
         SB4/P3                                Licensed
    SB16 - PROC       4          4
        SB16/P0                                Unused
        SB16/P1                                Unused
        SB16/P2                                Unused
        SB16/P3                                Unused
E - PROC              0          0         0
F - PROC              0          0         0
G - PROC              0          0         0
H - PROC              0          0         0
I - PROC              0          0         0
J - PROC              0          0         0
K - PROC              0          0         0
L - PROC              0          0         0
M - PROC              0          0         0
N - PROC              0          0         0
O - PROC              0          0         0
P - PROC              0          0         0
Q - PROC              0          0         0
R - PROC              0          0         0
Unused - PROC         0          0        12
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	User cancel.

2	Invalid usage.
3	User does not have valid privileges.
>3	An internal error occurred. For further information see `/var/opt/SUNWSMS/adm/platform/messages`.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`showcodlicense(1M)`, `codd(1M)`

NAME

showcomponent - display the blacklist status for a component

SYNOPSIS

showcomponent [`-a|-d` *domain_indicator*] [`-v`] [*location*]...

showcomponent `-h`

DESCRIPTION

`showcomponent`(1M) indicates whether the specified component is listed in the platform, domain, or ASR blacklist file.

If neither the `-a` nor the `-d` option is specified, `showcomponent` displays the platform blacklist. If no *location* is specified, `showcomponent` displays all components in the specified blacklist.

The *blacklist* is an internal file that lists components that POST cannot use at boot time. POST reads the blacklist file(s) before preparing the system for booting, and passes along to OpenBoot PROM a list of only those components that have been successfully tested; those on the blacklist are excluded.

SMS supports three blacklists: one for domain boards, one for platform boards, and the internal automatic system recovery (ASR) blacklist.

For more information on the use and editing of platform and domain blacklists refer to the *System Management Services (SMS) 1.6 Administrator Guide*.

OPTIONS

The following options are supported:

`-a`	Specifies the ASR blacklist.
`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-v`	Verbose. Displays all available command information.

OPERANDS The following operands are supported:

location List of component locations, separated by forward slashes and comprised of:

board_loc/proc/bank/logical_bank

board_loc/proc/bank/all_dimms_on_that_bank

board_loc/proc/all_banks_on_that_proc

board_loc/all_banks_on_that_board

board_loc/proc

board_loc

board_loc/procs

board_loc/cassette

board_loc/bus

board_loc/paroli_link

Multiple *location* arguments are permitted, separated by a space.

The *location* forms are optional and are used to specify particular components on boards in specific locations.

For example, the location SB5/P0/B1/L1 indicates

Logical Bank 1 of Bank 1 on Processor 0 at SB5.

The SB0/PP1 *location* indicates Processor Pair 1 at SB0.

The CS0/ABUS1 *location* indicates address bus 1 at CS0.

The following *board_loc* forms are accepted:

Sun Fire 12K/E20K:

```
SB(0...8)

IO(0...8)

CS(0|1)

EX(0...8)
```

Sun Fire 15K/E25K:

```
SB(0...17)

IO(0...17)

CS(0|1)

EX(0...17)
```

Processor locations indicate single processors or processor pairs.

There are four possible processors on a CPU/memory board. Processor pairs on that board are procs 0 and 1, and procs 2 and 3.

The MaxCPU has two processors: procs 0 and 1, and only one proc pair (PP0). Using PP1 for this board causes `showcomponent` to exit and display an error message.

The following *proc* forms are accepted:

```
P(0...3)    PP(0|1)
```

The following *bank* forms are accepted:

```
B(0|1)
```

The following *logical_bank* forms are accepted:

```
L(0|1)
```

The following *all_dimms_on_that_bank* forms are accepted:

```
D
```

The following *all_banks_on_that_proc* forms are accepted:

```
B
```

The following *all_banks_on_that_board* forms are accepted:

```
B
```

The following *paroli_link* forms are accepted:

```
PAR(0|1)
```

The hsPCI assemblies contain hot-swappable cassettes.

The following *hsPCI* forms are accepted:

```
C(3|5)V(0|1)
```

The hsPCI+ assemblies contain hot-swappable cassettes.

The following *hsPCI+* forms are accepted:

```
C3V(0|1|2) and C5V0
```

There are three bus locations: address, data, and response.

The following *bus* forms are accepted:

```
ABUS|DBUS|RBUS (0|1)
```

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or domain configurator privileges to run this command. If you have domain privileges you can run this command only on the domain for which you have privileges.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying Whether SB0 is ASR Blacklisted

```
sc0:sms-user:>  showcomponent -a SB0
Component SB0 is disabled: #High Voltage
```

EXAMPLE 2 Displaying Whether SB15 (“V3CPU”) is ASR Blacklisted

```
sc0:sms-user:>  showcomponent -a SB15
Component V3CPU at SB15 is NOT disabled in the specified blacklist
```

EXAMPLE 3 Displaying Whether SB11 (“CPU”) is ASR Blacklisted

```
sc0:sms-user:>  showcomponent -a SB11
Component CPU at SB11 is disabled in the specified blacklist: # ESMD sensor
read failure 0528.1306.24
```

EXAMPLE 4 Displaying Whether Four Boards/Components in Domain B Are Blacklisted

```
sc0:sms-user:>  showcomponent -dB IO4/PP0 SB5 IO6/C5V0 EX7/ABUS0
Component IO4/PP0 is disabled: #High temp
Component SB5 is disabled: <no reason given>
Component IO6/C5V0 is NOT disabled.
Component EX7/ABUS0 is NOT disabled.
```

EXAMPLE 5 Displaying Whether the Logical Bank on IO7 in Domain B Is Blacklisted

```
sc0:sms-user:>  showcomponent -dB IO7/P0/B1/L0
Component IO7/P0/B1/L0 is disabled: <no reason given>
```

EXAMPLE 6 Displaying All Platform-Blacklisted Components

```
sc0:sms-user:>  showcomponent
Component SB0 is disabled: #High temp
Component SB3 is disabled:
Component IO2 is disabled. <no reason given>
```

EXAMPLE 7 Displaying All Domain B Blacklisted Components

```
sc0:sms-user:>  showcomponent -dB
Component IO4/PP0 is disabled: #High temp
Component SB5 is disabled: <no reason given>
```

EXAMPLE 8 Displaying All ASR-Blacklisted Components

```
sc0:sms-user:>  showcomponent -a
Component SB0 is disabled: #High temp
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

FILES

The following files are used by this command.

`/etc/opt/SUNWSMS/config/asr/blacklist`	List of components excluded by `esmd`.
`/etc/opt/SUNWSMS/config/platform/blacklist`	List of platform components excluded.
`/etc/opt/SUNWSMS/config/`*domain_id*`/blacklist`	List of domain components excluded.

Note – The ASR blacklist file is created and used internally and should *not* be edited manually.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`enablecomponent(1M)`, `disablecomponent(1M)`, `esmd(1M)`

NAME

showdatasync - display the status of system controller (SC) data synchronization for failover

SYNOPSIS

showdatasync [`-l` | `-Q`] [`-v`]

showdatasync `-h`

DESCRIPTION

`showdatasync` provides the current status of files propagated (copied) from the main SC to its spare. Data propagation synchronizes data on the spare SC with data on the main SC, so that the spare SC is current with the main SC if an SC failover occurs.

OPTIONS

The following options are supported:

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-l` Lists the files in the current data propagation list. For details on the information displayed see the EXTENDED DESCRIPTION section.

`-Q` Lists the files queued for propagation. Each file name includes the absolute path to the file.

`-v` Verbose. Displays all available command information.

EXTENDED DESCRIPTION

This section describes the information displayed by the `showdatasync` command.

If you do not specify an option with the `showdatasync` command, the following information is displayed:

```
File Propagation Status:
Active File:
Queued files:
```

where:

`File Propagation Status`	Displays the current status of data synchronization:	
	`Active`	Indicates that the data synchronization process is enabled and functioning normally.
	`Disabled`	Indicates that the data synchronization process has been disabled because SC failover was disabled.
	`Failed`	Indicates that the data synchronization process cannot currently propagate files to the spare SC.
`Active File`	Displays either the absolute path of the file currently being propagated or a dash (-) indicating that the link is idle.	
`Queued files`	Specifies the number of files to be propagated but not yet processed.	

If you specify the `-l` option with the `showdatasync` command, each entry in the data propagation list is displayed in the format:

```
TIME PROPAGATED          INTERVAL     FILE
time                     interval     filename
```

where:

time	Indicates the last time that the file was propagated from the main SC to the spare.
interval	Specifies the interval, in minutes, between checks for file modification. The default interval is 60 minutes.
filename	Provides the absolute path and name of the propagated file.

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or domain configurator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying Data Synchronization Status

```
sc0:sms-user:> showdatasync
File Propagation State:  ACTIVE
Active File:             -
Queued files:            0
```

EXAMPLE 2 Displaying Data Synchronization List

```
sc0:sms-user:> showdatasync -l
TIME PROPAGATED          INTERVAL     FILE
Mar 23 16:00:00                60     /tmp/t1
```

EXAMPLE 3 Displaying Data Synchronization Queue

```
sc0:sms-user:> showdatasync -Q
FILE
/tmp/t1
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`setdatasync(1M)`

NAME

showdate - display the date and time for the system controller (SC) or a domain

SYNOPSIS

showdate [`-d` *domain_indicator*] [`-u`] [`-v`]

showdate `-h`

DESCRIPTION

`showdate` (1M) displays the SC's current date and time. Optionally, `showdate` displays domain time of day.

OPTIONS

The following options are supported:

`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-u`	Interprets and displays the time using Greenwich Mean Time (GMT). The default is the local time zone.
`-v`	Verbose. Displays all available command information.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, platform operator, or platform service privileges to display the date on the SC. You must have domain administrator or domain configurator privileges for the specified domain to display the domain date.

Refer to Chapter 2, "SMS Security Options and Administrative Privileges" in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Showing the Current Local Date in Pacific Standard Time

```
sc0:sms-user:> showdate
System Controller: Sat Feb 2 15:23:21 PST 2002
```

EXAMPLE 2 Showing the Current Date Using GMT

```
sc0:sms-user:> showdate -u
System Controller: Sat Feb 2 23:23:21 GMT 2002
```

EXAMPLE 3 Showing the Current Local Date on Domain A in Pacific Standard Time

```
sc0:sms-user:> showdate -d a
Domain a: Sat Feb 2 15:33:20 PST 2002
```

EXAMPLE 4 Showing the Current Date on Domain A Using GMT

```
sc0:sms-user:> showdate -d a -u
Domain a: Sat Feb 2 23:33:20 GMT 2002
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `setdate(1M)`

NAME

showdevices - display system board devices and resource usage information

SYNOPSIS

showdevices [`-v`] [`-p` bydevice | byboard | query | force] *location...*

showdevices [`-v`] [`-p` bydevice | byboard] `-d` *domain_indicator*

showdevices `-h`

DESCRIPTION

`showdevices`(1M) displays the configured physical devices on system boards and the resources made available by these devices. Usage information is provided by applications and subsystems that are actively managing system resources. You can perform offline queries of managed resources to display the predicted impact of a system board DR operation. Unmanaged devices are not displayed by default; you must use the `-v` option.

`showdevices` gathers device information from one or more Sun Fire high-end system domains. The command uses `dca`(1M) as a proxy to gather the information from the domains.

OPTIONS The following options are supported.

`-d` *domain_indicator* Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-p` Displays specific reports.

Valid arguments for `-p` are:

`bydevice`	List output is grouped by device type (CPU, memory, IO). This is the default.
`byboard`	List output is grouped by system board. Default output is in tabular format grouped by device type (CPU, memory, IO).
`query`	List output is the result of a query to predict the effect of removing a system board
`force`	Forced offline query. Resource consumers are requested to apply force semantics in predicting whether they are able to relinquish usage of the system resources. (see `cfgadm`(1M)).

Note – The `query` and `force` arguments are *not* valid with the `-d` option.

`-v` Displays all I/O devices. Includes both managed and unmanaged I/O devices. Managed devices export actively managed resources. Unmanaged devices are physically configured but do not export actively managed resources. No usage information is available for unmanaged devices.

OPERANDS

The following operands are supported:

location — List of board locations, separated by a space. Multiple *location* arguments are permitted.

The following *location* forms are accepted:

Sun Fire 12K/E20K:

```
SB(0...8)
IO(0...8)
```

Sun Fire 15K/E25K:

```
SB(0...17)
IO(0...17)
```

EXTENDED DESCRIPTION

`Showdevices` fields:

`domain`	Tag or identifier
`board`	Board identifier

CPU:

`id`	Processor ID
`state`	Processor state
`speed`	CPU frequency in MHz
`ecache`	CPU ecache size in MB

Memory:

`board mem`	Board memory size in MB
`perm mem`	Amount of nonrelocatable memory on board in MB
`base address`	Base physical address of memory on board
`domain mem`	System memory size in MB
`board`	Board identifier

If a memory drain is in progress, the following is available:

`target board`	Target board identifier
`deleted`	Amount of memory already deleted in MB
`remaining`	Amount of memory remaining to be deleted in MB

I/O devices:

`device`	I/O device instance name
`resource`	Managed resource name
`usage`	Description of resource usage instance
`query`	Result of offline query of resources

Group Privileges Required

You must have domain administrator/configurator privileges on all boards specified to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying devices for Board IO1

```
sc0:sms-user:> showdevices IO1
IO Devices
----------
domain board   device     resource           usage
A      IO1      sd3     /dev/dsk/c0t3d0s0  mounted filesystem "/"
A      IO1      sd3     /dev/dsk/c0t3s0s1  dump device (swap)
A      IO1      sd3     /dev/dsk/c0t3s0s1  swap area
A      IO1      sd3     /dev/dsk/c0t3d0s3  mounted filesystem "/var"
A      IO1      sd3     /var/run           mounted filesystem "/var/run"
```

EXAMPLE 2 Displaying devices for SB1 and IO1

```
sc0:sms-user> showdevices sb1 io1
CPU
---
domain    board  id     state        speed  ecache  usage
B         SB1    32     online       1050   8
B         SB1    33     online       1050   8
B         SB1    34     online       1050   8
B         SB1    35     online       1050   8
B         SB1    36     online       1050   8
B         SB1    37     online       1050   8
B         SB1    38     online       1050   8
B         SB1    39     online       1050   8


Memory
------
                  board   perm    base           domain
domain    board   mem MB  mem MB  address        mem MB
B         SB1     16384   934     0x2000000000   16384


IO Devices
----------
domain    board  device  resource                    usage
B         IO1    ce0     SUNW_network/ce0            ce0 hosts IP
addresses: 10.1.134.133
B         IO1    sd0     /dev/dsk/c0t0d0s0          mounted filesystem "/"
B         IO1    sd0      /dev/dsk/c0t0d0s1           swap area
B         IO1    sd0      /dev/dsk/c0t0d0s1           dump device (swap)
```

EXAMPLE 3 Displaying Offline Query Result for System Board IO1

```
sc0:sms-user:> showdevices -p query IO1

Location IO1 - Domain A
====================
IO Devices
----------
device  resource               query  usage/reason
sd3     /dev/dsk/c0t3d0s0      fail   mounted filesystem "/"
sd3     /dev/dsk/c0t3s0s1      fail   dump device (swap)
sd3     /dev/dsk/c0t3s0s1      fail   swap area
sd3     /dev/dsk/c0t3d0s3      fail   mounted filesystem "/var"
sd3     /var/run                -     mounted filesystem "/var/run"
```

The query field shows the predicted result of removing the resource. The failure of the mounted file system `/var` to offline prevents the query from reaching the layered mount point `/var/run`.

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	An invalid domain was specified.
2	A command line error such as an invalid option was detected.
3	More than one domain was specified.
4	An error occurred while communicating with `pcd`.
5	An error occurred while communicating with a domain.
6	An error occurred while handling device information.
7	An internal error, such as failed memory allocation, occurred.

ATTRIBUTES

See `attributes(5)` for a description of the following attribute.

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `dca(1M)`, `pcd(1M)`

NAME

showenvironment - display the environmental data

SYNOPSIS

showenvironment [`-d` *domain_indicator* [, *domain_indicator*]...]...[`-p` temps | volts | currents | fans | powers [,temps | volts | currents | fans | powers]..] [`-v`]

showenvironment [`-d` *domain_indicator*[, *domain_indicator*]...]... [`-p` faults] [`-v`]

showenvironment `-h`

DESCRIPTION

`showenvironment`(1M) displays the environmental data (temperatures, voltages, and so on). If a domain *domain_id* or *domain_tag* is specified, environmental data relating to the domain is displayed, provided that the user has domain privileges for that domain. If a domain is not specified, all domain data permissible to the user is displayed.

Note – Only *domain configuration units* (DCUs) (for example, CPU or I/O) belong to a domain. Displaying environmental data relating to such things as fan trays, bulk power, or other boards (`exb`, `csb`) requires platform privileges. You can also specify individual reports for temperatures, voltages, currents, bulk power status, and fan tray status; or specify faults with the `-p` option. If the `-p` option is not present, all reports are shown.

OPTIONS

The following options are supported:

`-d` *domain_indicator* — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-p` Displays specific reports. Multiple report arguments are separated by commas.

Valid reports are:

`temps`	List output is grouped by temperature
`volts`	List output is grouped by voltage
`currents`	List output is grouped by current
`fans`	List output is grouped by fans
`powers`	List output is grouped by bulk power supplies

`-p faults` List output is of all component readings not within the optimum thresholds.

Note – The `faults` argument cannot be used in conjunction with any other report argument.

`-v` Verbose. Displays all available command information.

EXTENDED DESCRIPTION

The Unit field contains one of three measurements:

`C`	Degrees Celsius
`V`	Volts
`A`	Amperes

The Status field can contain one of 16 states.

Temperature readings:

`OVERLIMIT`	Over limit
`HIGH_CRIT`	High critical
`HIGH_WARN`	High warning
`LOW_CRIT`	Low critical
`LOW_WARN`	Low warning
`OK`	Optimum
`INVALID`	Reading failure

Voltage readings:

HIGH_MAX	High maximum
LOW_MIN	Low minimum
OK	Acceptable
INVALID	Reading failure

Current readings:

OK	The difference between both companion component readings is within tolerance.
BAD	The difference between both companion component readings is out of tolerance.
INVALID	Reading failure.

Miscellaneous:

ON	Power on.
OFF	Power off.
PRESENCE	A HotPlug card is present in slot 1.
FAIL	Failure state.
HIGH	Set to high speed.
NORMAL	Set to normal speed.
INVALID	Reading failure.
AGE	Age of the reading.
UNKNOWN	Unknown power/board type.

Group Privileges Required

Only domain information for which you have domain administrator or configurator privileges for is displayed. Otherwise, you must have platform administrator, operator, or service privileges.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 showenvironment Output

```
LOCATION          SENSOR            VALUE   UNIT  AGE         STATUS
----------        ------------      -----   ----  ------      ------
SC at SC0         RIO Temp          35      C     41.3  sec   OK
SC at SC0         IOA Temp          28      C     41.3  sec   OK
SC at SC0         PS0 Temp          43      C     41.3  sec   OK
SC at SC0         PS1 Temp          37      C     41.3  sec   OK
SCPER at SCPER0   AMB 0 Temp        21      C     43.8  sec   OK
SCPER at SCPER0   AMB 1 Temp        22      C     43.8  sec   OK
SCPER at SCPER0   AMB 2 Temp        22      C     43.8  sec   OK
SC at SC0         1.5 VDC           1.48    V     19.8  sec   OK
SC at SC0         3.3 VDC           3.28    V     19.8  sec   OK
SC at SC0         3.3 VDC HK        3.28    V     19.8  sec   OK
SC at SC0         5.0 VDC           4.99    V     19.8  sec   OK
SC at SC0         +12.0 VDC         12.03   V     19.8  sec   OK
SC at SC0         -12.0 VDC         -12.01  V     19.8  sec   OK
SCPER at SCPER0   3.3 VDC HK        3.30    V     19.2  sec   OK
SCPER at SCPER0   5.0 VDC           5.02    V     19.2  sec   OK
SCPER at SCPER0   +12.0 VDC         12.48   V     19.2  sec   OK
SC at SC0         1.5 CVT0 VDC      1.58    A     19.8  sec   OK
SC at SC0         1.5 CVT1 VDC      1.66    A     19.8  sec   OK
SC at SC0         3.3 V_PS0         6.29    A     19.8  sec   OK
SC at SC0         3.3 V_PS1         6.44    A     19.8  sec   OK
SC at SC0         5.0 V_PS0         6.34    A     19.8  sec   OK
SC at SC0         5.0 V_PS1         6.46    A     19.8  sec   OK
SC at SC1         RIO Temp          33      C     41.3  sec   OK
SC at SC1         IOA Temp          27      C     41.3  sec   OK
SC at SC1         PS0 Temp          37      C     41.3  sec   OK
SC at SC1         PS1 Temp          48      C     41.3  sec   OK
SC at SC1         SBBC Temp         40      C     41.3  sec   OK
SC at SC1         CBH Temp          45      C     41.3  sec   OK
SCPER at SCPER1   AMB 0 Temp        20      C     43.8  sec   OK
SCPER at SCPER1   AMB 1 Temp        20      C     43.8  sec   OK
SCPER at SCPER1   AMB 2 Temp        20      C     43.8  sec   OK
SC at SC1         1.5 VDC           1.50    V     19.4  sec   OK
SC at SC1         3.3 VDC           3.26    V     19.4  sec   OK
SC at SC1         3.3 VDC HK        3.28    V     19.4  sec   OK
SC at SC1         5.0 VDC           5.12    V     19.4  sec   OK
SC at SC1         +12.0 VDC         12.03   V     19.4  sec   OK
SC at SC1         -12.0 VDC         -12.01  V     19.4  sec   OK
SCPER at SCPER1   3.3 VDC HK        3.30    V     19.1  sec   OK
SCPER at SCPER1   5.0 VDC           4.99    V     19.1  sec   OK
SCPER at SCPER1   +12.0 VDC         12.03   V     19.1  sec   OK
SC at SC1         1.5 CVT0 VDC      0.98    A     19.5  sec   OK
SC at SC1         1.5 CVT1 VDC      0.97    A     19.5  sec   OK
SC at SC1         3.3 V_PS0         5.27    A     19.5  sec   OK
SC at SC1         3.3 V_PS1         10.69   A     19.5  sec   OK
SC at SC1         5.0 V_PS0         4.51    A     19.5  sec   OK
SC at SC1         5.0 V_PS1         9.27    A     19.5  sec   OK
CSB at CS0        AMB Top Temp      27      C     41.4  sec   OK
CSB at CS0        AMB Bot Temp      28      C     41.4  sec   OK
CSB at CS0        SBBC Temp         39      C     41.4  sec   OK
CSB at CS0        1.5 VDC           1.51    V     20.0  sec   OK
CSB at CS0        3.3 VDC           3.28    V     20.0  sec   OK
CSB at CS0        2.5 VDC           2.54    V     20.0  sec   OK
CSB at CS0        3.3 VDC HK        3.28    V     20.0  sec   OK
CSB at CS1        AMB Top Temp      25      C     41.3  sec   OK
CSB at CS1        AMB Bot Temp      26      C     41.3  sec   OK
CSB at CS1        1.5 VDC           1.51    V     19.9  sec   OK
```

```
CSB at CS1      3.3 VDC          3.30    V     19.9  sec  OK
CSB at CS1      2.5 VDC          2.54    V     19.9  sec  OK
CSB at CS1      3.3 VDC HK       3.28    V     19.9  sec  OK
CP at CP0       DMX0 Temp        32      C     41.5  sec  OK
CP at CP0       DMX1 Temp        30      C     41.5  sec  OK
CP at CP0       DMX3 Temp        28      C     41.5  sec  OK
CP at CP0       DMX5 Temp        24      C     41.5  sec  OK
CP at CP0       AMX0 Temp        32      C     41.5  sec  OK
CP at CP0       AMX1 Temp        32      C     41.5  sec  OK
CP at CP0       RMX Temp         32      C     41.5  sec  OK
CP at CP0       DARB Temp        28      C     41.5  sec  OK
CP at CP1       DMX0 Temp        30      C     41.4  sec  OK
CP at CP1       DMX1 Temp        28      C     41.4  sec  OK
CP at CP1       DMX3 Temp        28      C     41.4  sec  OK
CP at CP1       DMX5 Temp        26      C     41.4  sec  OK
CP at CP1       AMX0 Temp        34      C     41.4  sec  OK
CP at CP1       AMX1 Temp        34      C     41.4  sec  OK
CP at CP1       RMX Temp         35      C     41.4  sec  OK
CP at CP1       DARB Temp        30      C     41.4  sec  OK
EXB at EX0      AMB Top Temp     33      C     41.9  sec  OK
EXB at EX0      AMB Bot Temp     30      C     41.9  sec  OK
EXB at EX0      SBBC Temp        44      C     41.9  sec  OK
EXB at EX0      AXQ Temp         37      C     41.9  sec  OK
EXB at EX0      SDIM Temp        43      C     41.9  sec  OK
EXB at EX0      SDISC Temp       43      C     41.9  sec  OK
EXB at EX0      SDISE Temp       37      C     41.9  sec  OK
EXB at EX0      1.5 VDC          1.50    V     21.0  sec  OK
EXB at EX0      3.3 VDC          3.28    V     21.0  sec  OK
EXB at EX0      2.5 VDC          2.49    V     21.0  sec  OK
EXB at EX0      3.3 VDC HK       3.30    V     21.0  sec  OK
CPU at SB0      PROC 0 Temp      50      C     0.1   sec  OK
CPU at SB0      PROC 1 Temp      53      C     0.1   sec  OK
CPU at SB0      PROC 2 Temp      54      C     0.1   sec  OK
CPU at SB0      PROC 3 Temp      50      C     0.1   sec  OK
CPU at SB0      SDC0 Temp        66      C     0.1   sec  OK
CPU at SB0      AR0 Temp         60      C     0.1   sec  OK
CPU at SB0      DX0 Temp         59      C     0.1   sec  OK
CPU at SB0      DX1 Temp         58      C     0.1   sec  OK
CPU at SB0      DX2 Temp         54      C     0.1   sec  OK
CPU at SB0      DX3 Temp         50      C     0.1   sec  OK
CPU at SB0      SBBC0 Temp       55      C     0.1   sec  OK
CPU at SB0      SBBC1 Temp       56      C     0.1   sec  OK
CPU at SB0      1.5 VDC          1.51    V     23.1  sec  OK
CPU at SB0      3.3 VDC          3.30    V     23.1  sec  OK
CPU at SB0      Core 0 Volt      1.39    V     23.1  sec  OK
CPU at SB0      Core 1 Volt      1.40    V     23.1  sec  OK
CPU at SB0      Core 2 Volt      1.39    V     23.1  sec  OK
CPU at SB0      Core 3 Volt      1.38    V     23.1  sec  OK
HPCI at IO0     PS0 Temp         42      C     43.8  sec  OK
HPCI at IO0     PS1 Temp         34      C     43.8  sec  OK
HPCI at IO0     SDC0 Temp        62      C     43.8  sec  OK
HPCI at IO0     AR0 Temp         57      C     43.8  sec  OK
HPCI at IO0     DX0 Temp         53      C     43.8  sec  OK
HPCI at IO0     DX1 Temp         46      C     43.8  sec  OK
HPCI at IO0     SBBC Temp        35      C     43.8  sec  OK
HPCI at IO0     IOA0 Temp        45      C     43.8  sec  OK
HPCI at IO0     IOA1 Temp        46      C     43.8  sec  OK
HPCI at IO0     1.5 VDC          1.49    V     22.7  sec  OK
HPCI at IO0     3.3 VDC          3.30    V     22.7  sec  OK
HPCI at IO0     5.0 VDC          4.99    V     22.7  sec  OK
```

```
HPCI at IO0       +12.0 VDC         11.88   V     22.7  sec  OK
HPCI at IO0       -12.0 VDC         -12.01  V     22.7  sec  OK
HPCI at IO0       3.3 VDC HK        3.30    V     22.7  sec  OK
HPCI at IO0       1.5 CVT0 VDC      1.49    A     22.7  sec  OK
HPCI at IO0       1.5 CVT1 VDC      0.00    A     22.7  sec  OK
HPCI at IO0       3.3 V_PS0         10.25   A     22.7  sec  OK
HPCI at IO0       3.3 V_PS1         10.25   A     22.7  sec  OK
HPCI at IO0       5.0 V_PS0         3.78    A     22.7  sec  OK
HPCI at IO0       5.0 V_PS1         3.90    A     22.7  sec  OK
 Schizo0.0      Schizo 0 Slot 0  N/A     N/A   N/A         PRESENCE
 Schizo1.0      Schizo 1 Slot 0  N/A     N/A   N/A         PRESENCE
 Schizo0.1      Schizo 0 Slot 1  N/A     N/A   N/A         PRESENCE
 Schizo1.1      Schizo 1 Slot 1  N/A     N/A   N/A         PRESENCE
EXB at EX1        --                --      --    --          OFF
CPU at SB1        --                --      --    --          OFF
HPCI at IO1       --                --      --    --          OFF
EXB at EX3        --                --      --    --          OFF
V3CPU at SB3      --                --      --    --          OFF
HPCI at IO3       --                --      --    --          OFF
EXB at EX6        AMB Top Temp      22      C     42.1  sec  OK
EXB at EX6        AMB Bot Temp      23      C     42.1  sec  OK
EXB at EX6        SBBC Temp         36      C     42.1  sec  OK
EXB at EX6        AXQ Temp          29      C     42.1  sec  OK
EXB at EX6        SDIM Temp         33      C     42.1  sec  OK
EXB at EX6        SDISC Temp        35      C     42.1  sec  OK
EXB at EX6        SDISE Temp        29      C     42.1  sec  OK
EXB at EX6        1.5 VDC           1.51    V     21.1  sec  OK
EXB at EX6        3.3 VDC           3.32    V     21.1  sec  OK
EXB at EX6        2.5 VDC           2.51    V     21.1  sec  OK
EXB at EX6        3.3 VDC HK        3.30    V     21.1  sec  OK
CPU at SB6        --                --      --    --          OFF
MCPU at IO6       PROC 0 Temp       31      C     3.5   sec  OK
MCPU at IO6       PROC 1 Temp       31      C     3.5   sec  OK
MCPU at IO6       SBBC0 Temp        32      C     3.5   sec  OK
MCPU at IO6       SDC0 Temp         44      C     3.5   sec  OK
MCPU at IO6       AR0 Temp          39      C     3.5   sec  OK
MCPU at IO6       DX0  Temp         39      C     3.5   sec  OK
MCPU at IO6       DX1  Temp         39      C     3.5   sec  OK
MCPU at IO6       DX2  Temp         38      C     3.5   sec  OK
MCPU at IO6       DX3  Temp         37      C     3.5   sec  OK
MCPU at IO6       1.5 VDC           1.49    V     22.8  sec  OK
MCPU at IO6       3.3 VDC           3.28    V     22.8  sec  OK
MCPU at IO6       3.3 VDC HK        3.28    V     22.8  sec  OK
MCPU at IO6       Core 0 Volt       1.38    V     22.8  sec  OK
MCPU at IO6       Core 1 Volt       1.37    V     22.8  sec  OK
EXB at EX7        --                --      --    --          OFF
MCPU at IO7       --                --      --    --          OFF
EXB at EX8        --                --      --    --          OFF
MCPU at IO8       --                --      --    --          OFF
EXB at EX9        AMB Top Temp      34      C     4.2   sec  OK
EXB at EX9        AMB Bot Temp      31      C     4.2   sec  OK
EXB at EX9        SBBC Temp         46      C     4.2   sec  OK
EXB at EX9        AXQ Temp          36      C     4.2   sec  OK
EXB at EX9        SDIM Temp         40      C     4.2   sec  OK
EXB at EX9        SDISC Temp        42      C     4.2   sec  OK
EXB at EX9        SDISE Temp        36      C     4.2   sec  OK
EXB at EX9        1.5 VDC           1.51    V     20.9  sec  OK
EXB at EX9        3.3 VDC           3.30    V     20.9  sec  OK
EXB at EX9        2.5 VDC           2.49    V     20.9  sec  OK
EXB at EX9        3.3 VDC HK        3.28    V     20.9  sec  OK
```

```
CPU at SB9       PROC 0 Temp      39       C      1.0   sec  OK
CPU at SB9       PROC 1 Temp      41       C      1.0   sec  OK
CPU at SB9       PROC 2 Temp      43       C      1.0   sec  OK
CPU at SB9       PROC 3 Temp      41       C      1.0   sec  OK
CPU at SB9       SDC0 Temp        62       C      1.0   sec  OK
CPU at SB9       AR0 Temp         56       C      1.0   sec  OK
CPU at SB9       DX0 Temp         56       C      1.0   sec  OK
CPU at SB9       DX1 Temp         56       C      1.0   sec  OK
CPU at SB9       DX2 Temp         53       C      1.0   sec  OK
CPU at SB9       DX3 Temp         47       C      1.0   sec  OK
CPU at SB9       SBBC0 Temp       49       C      1.0   sec  OK
CPU at SB9       SBBC1 Temp       54       C      1.0   sec  OK
CPU at SB9       1.5 VDC          1.51     V      43.7  sec  OK
CPU at SB9       3.3 VDC          3.34     V      43.7  sec  OK
CPU at SB9       Core 0 Volt      1.63     V      43.7  sec  OK
CPU at SB9       Core 1 Volt      1.63     V      43.7  sec  OK
CPU at SB9       Core 2 Volt      1.62     V      43.7  sec  OK
CPU at SB9       Core 3 Volt      1.62     V      43.7  sec  OK
HPCI+ at IO9     D147#0_0         34       C      9.3   sec  OK
HPCI+ at IO9     D147#0_1         41       C      9.3   sec  OK
HPCI+ at IO9     D147#0_2         33       C      9.3   sec  OK
HPCI+ at IO9     D147#1_0         36       C      9.3   sec  OK
HPCI+ at IO9     D147#1_1         42       C      9.3   sec  OK
HPCI+ at IO9     D147#1_2         39       C      9.3   sec  OK
HPCI+ at IO9     XMITS0 Temp      37       C      9.3   sec  OK
HPCI+ at IO9     XMITS1 Temp      49       C      9.3   sec  OK
HPCI+ at IO9     SBBC Temp        35       C      9.3   sec  OK
HPCI+ at IO9     SDC0 Temp        58       C      9.3   sec  OK
HPCI+ at IO9     AR0 Temp         57       C      9.3   sec  OK
HPCI+ at IO9     DX0 Temp         55       C      9.3   sec  OK
HPCI+ at IO9     DX1 Temp         48       C      9.3   sec  OK
HPCI+ at IO9     1.5 VDC0         1.51     V      28.3  sec  OK
HPCI+ at IO9     2.5 VDC0         2.48     V      28.3  sec  OK
HPCI+ at IO9     3.3 VDC0         3.28     V      28.3  sec  OK
HPCI+ at IO9     5.0 VDC0         5.05     V      28.3  sec  OK
HPCI+ at IO9     +12.0 VDC0       12.34    V      28.3  sec  OK
HPCI+ at IO9     -12.0 VDC0       -12.18   V      28.3  sec  OK
HPCI+ at IO9     3.3 VDC HK0      3.32     V      28.3  sec  OK
HPCI+ at IO9     1.5 VDC1         1.52     V      28.3  sec  OK
HPCI+ at IO9     2.5 VDC1         2.48     V      28.3  sec  OK
HPCI+ at IO9     3.3 VDC1         3.28     V      28.3  sec  OK
HPCI+ at IO9     5.0 VDC1         5.05     V      28.3  sec  OK
HPCI+ at IO9     +12.0 VDC1       12.34    V      28.3  sec  OK
HPCI+ at IO9     -12.0 VDC1       -12.10   V      28.3  sec  OK
HPCI+ at IO9     3.3 VDC HK1      3.32     V      28.3  sec  OK
 XMITS0.0       XMITS 0 Slot 0   N/A      N/A    N/A         PRESENCE
 XMITS1.0       XMITS 1 Slot 0   N/A      N/A    N/A         PRESENCE
 XMITS0.1       XMITS 1 Slot 1   N/A      N/A    N/A         PRESENCE
 XMITS1.1       XMITS 1 Slot 1   N/A      N/A    N/A         PRESENCE
EXB at EX13      AMB Top Temp     22       C      42.0  sec  OK
EXB at EX13      AMB Bot Temp     23       C      42.0  sec  OK
EXB at EX13      SBBC Temp        37       C      42.0  sec  OK
EXB at EX13      AXQ Temp         29       C      42.0  sec  OK
EXB at EX13      SDIM Temp        33       C      42.0  sec  OK
EXB at EX13      SDISC Temp       33       C      42.0  sec  OK
EXB at EX13      SDISE Temp       29       C      42.0  sec  OK
EXB at EX13      1.5 VDC          1.50     V      20.8  sec  OK
EXB at EX13      3.3 VDC          3.30     V      20.8  sec  OK
EXB at EX13      2.5 VDC          2.49     V      20.8  sec  OK
EXB at EX13      3.3 VDC HK       3.28     V      20.8  sec  OK
```

```
CPU at SB13      --                 --      --    --          OFF
WPCI at IO13     IOA0 Temp          26      C     43.2  sec   OK
WPCI at IO13     DX0 Temp           47      C     43.2  sec   OK
WPCI at IO13     DX1 Temp           42      C     43.2  sec   OK
WPCI at IO13     SDC Temp           46      C     43.2  sec   OK
WPCI at IO13     SBBC Temp          33      C     43.2  sec   OK
WPCI at IO13     AR Temp            42      C     43.2  sec   OK
WPCI at IO13     WCI0 Temp          26      C     43.2  sec   OK
WPCI at IO13     WCI1 Temp          26      C     43.2  sec   OK
WPCI at IO13     +12 VDC            12.03   V     22.4  sec   OK
WPCI at IO13     -12 VDC            -11.92  V     22.4  sec   OK
WPCI at IO13     3.3 HK             3.28    V     22.4  sec   OK
WPCI at IO13     3.3 VDC            3.28    V     22.4  sec   OK
WPCI at IO13     1.5 VDC            1.49    V     22.4  sec   OK
WPCI at IO13     2.5 VDC            2.47    V     22.4  sec   OK
WPCI at IO13     5.0 VDC            4.96    V     22.4  sec   OK
WPCI at IO13     1.5 VDC PAR0       1.51    V     22.4  sec   OK
WPCI at IO13     1.5 VDC PAR1       1.52    V     22.4  sec   OK
WPCI at IO13     3.3 VDC PAR0       3.26    V     22.4  sec   OK
WPCI at IO13     3.3 VDC PAR1       3.24    V     22.4  sec   OK
 Schizo0.0       Schizo 0 Slot 0    N/A     N/A   N/A         PRESENCE
 Schizo0.1       Schizo 0 Slot 1    N/A     N/A   N/A         PRESENCE
EXB at EX15      AMB Top Temp       23      C     42.1  sec   OK
EXB at EX15      AMB Bot Temp       24      C     42.1  sec   OK
EXB at EX15      SBBC Temp          37      C     42.1  sec   OK
EXB at EX15      AXQ Temp           29      C     42.1  sec   OK
EXB at EX15      SDIM Temp          31      C     42.1  sec   OK
EXB at EX15      SDISC Temp         35      C     42.1  sec   OK
EXB at EX15      SDISE Temp         31      C     42.1  sec   OK
EXB at EX15      1.5 VDC            1.50    V     20.8  sec   OK
EXB at EX15      3.3 VDC            3.28    V     20.8  sec   OK
EXB at EX15      2.5 VDC            2.49    V     20.8  sec   OK
EXB at EX15      3.3 VDC HK         3.36    V     20.8  sec   OK
CPU at SB15      --                 --      --    --          OFF
HPCI+ at IO15    D147#0_0           25      C     43.2  sec   OK
HPCI+ at IO15    D147#0_1           29      C     43.2  sec   OK
HPCI+ at IO15    D147#0_2           29      C     43.2  sec   OK
HPCI+ at IO15    D147#1_0           28      C     43.2  sec   OK
HPCI+ at IO15    D147#1_1           31      C     43.2  sec   OK
HPCI+ at IO15    D147#1_2           37      C     43.2  sec   OK
HPCI+ at IO15    XMITS0 Temp        32      C     43.2  sec   OK
HPCI+ at IO15    XMITS1 Temp        53      C     43.2  sec   OK
HPCI+ at IO15    SBBC Temp          30      C     43.2  sec   OK
HPCI+ at IO15    SDC0 Temp          40      C     43.2  sec   OK
HPCI+ at IO15    AR0 Temp           36      C     43.2  sec   OK
HPCI+ at IO15    DX0 Temp           36      C     43.2  sec   OK
HPCI+ at IO15    DX1 Temp           36      C     43.2  sec   OK
HPCI+ at IO15    1.5 VDC0           1.51    V     22.3  sec   OK
HPCI+ at IO15    2.5 VDC0           2.49    V     22.3  sec   OK
HPCI+ at IO15    3.3 VDC0           3.28    V     22.3  sec   OK
HPCI+ at IO15    5.0 VDC0           5.02    V     22.3  sec   OK
HPCI+ at IO15    +12.0 VDC0         12.34   V     22.3  sec   OK
HPCI+ at IO15    -12.0 VDC0         -12.26  V     22.3  sec   OK
HPCI+ at IO15    3.3 VDC HK0        3.35    V     22.3  sec   OK
HPCI+ at IO15    1.5 VDC1           1.52    V     22.3  sec   OK
HPCI+ at IO15    2.5 VDC1           2.49    V     22.3  sec   OK
HPCI+ at IO15    3.3 VDC1           3.28    V     22.3  sec   OK
HPCI+ at IO15    5.0 VDC1           4.99    V     22.3  sec   OK
HPCI+ at IO15    +12.0 VDC1         12.34   V     22.3  sec   OK
HPCI+ at IO15    -12.0 VDC1         -12.18  V     22.3  sec   OK
```

```
HPCI+ at IO15     3.3 VDC HK1       3.35    V      22.3  sec  OK
 XMITS1.0        XMITS 1 Slot 0    N/A     N/A    N/A         PRESENCE
 XMITS0.1        XMITS 0 Slot 1    N/A     N/A    N/A         PRESENCE
 XMITS1.1        XMITS 1 Slot 1    N/A     N/A    N/A         PRESENCE
EXB at EX16       AMB Top Temp      31      C      42.1  sec  OK
EXB at EX16       AMB Bot Temp      30      C      42.1  sec  OK
EXB at EX16       SBBC Temp         42      C      42.1  sec  OK
EXB at EX16       AXQ Temp          37      C      42.1  sec  OK
EXB at EX16       SDIM Temp         39      C      42.1  sec  OK
EXB at EX16       SDISC Temp        39      C      42.1  sec  OK
EXB at EX16       SDISE Temp        39      C      42.1  sec  OK
EXB at EX16       1.5 VDC           1.51    V      20.7  sec  OK
EXB at EX16       3.3 VDC           3.30    V      20.7  sec  OK
EXB at EX16       2.5 VDC           2.51    V      20.7  sec  OK
EXB at EX16       3.3 VDC HK        3.28    V      20.7  sec  OK
V3CPU at SB16     PROC 0 Temp       51      C      10.7  sec  OK
V3CPU at SB16     PROC 1 Temp       54      C      10.7  sec  OK
V3CPU at SB16     PROC 2 Temp       56      C      10.7  sec  OK
V3CPU at SB16     PROC 3 Temp       53      C      10.7  sec  OK
V3CPU at SB16     SDC0 Temp         62      C      10.7  sec  OK
V3CPU at SB16     AR0 Temp          56      C      10.7  sec  OK
V3CPU at SB16     DX0 Temp          56      C      10.7  sec  OK
V3CPU at SB16     DX1 Temp          54      C      10.7  sec  OK
V3CPU at SB16     DX2 Temp          54      C      10.7  sec  OK
V3CPU at SB16     DX3 Temp          49      C      10.7  sec  OK
V3CPU at SB16     SBBC0 Temp        54      C      10.7  sec  OK
V3CPU at SB16     SBBC1 Temp        52      C      10.7  sec  OK
V3CPU at SB16     1.5 VDC           1.51    V      23.4  sec  OK
V3CPU at SB16     3.3 VDC           3.30    V      23.4  sec  OK
V3CPU at SB16     Core 0 Volt       1.20    V      23.4  sec  OK
V3CPU at SB16     Core 1 Volt       1.20    V      23.4  sec  OK
V3CPU at SB16     Core 2 Volt       1.20    V      23.4  sec  OK
V3CPU at SB16     Core 3 Volt       1.20    V      23.4  sec  OK
HPCI at IO16      PS0 Temp          51      C      42.4  sec  OK
HPCI at IO16      PS1 Temp          37      C      42.4  sec  OK
HPCI at IO16      SDC0 Temp         68      C      42.4  sec  OK
HPCI at IO16      AR0 Temp          64      C      42.4  sec  OK
HPCI at IO16      DX0 Temp          56      C      42.4  sec  OK
HPCI at IO16      DX1 Temp          47      C      42.4  sec  OK
HPCI at IO16      SBBC Temp         38      C      42.4  sec  OK
HPCI at IO16      IOA0 Temp         56      C      42.4  sec  OK
HPCI at IO16      IOA1 Temp         49      C      42.4  sec  OK
HPCI at IO16      1.5 VDC           1.49    V      22.0  sec  OK
HPCI at IO16      3.3 VDC           3.28    V      22.0  sec  OK
HPCI at IO16      5.0 VDC           4.96    V      22.0  sec  OK
HPCI at IO16      +12.0 VDC         11.88   V      22.0  sec  OK
HPCI at IO16      -12.0 VDC         -12.01  V      22.0  sec  OK
HPCI at IO16      3.3 VDC HK        3.30    V      22.0  sec  OK
HPCI at IO16      1.5 CVT0 VDC      1.49    A      22.0  sec  OK
HPCI at IO16      1.5 CVT1 VDC      0.00    A      22.0  sec  OK
HPCI at IO16      3.3 V_PS0         10.54   A      22.0  sec  OK
HPCI at IO16      3.3 V_PS1         10.10   A      22.0  sec  OK
HPCI at IO16      5.0 V_PS0         4.27    A      22.0  sec  OK
HPCI at IO16      5.0 V_PS1         4.27    A      22.0  sec  OK
 Schizo0.0       Schizo 0 Slot 0   N/A     N/A    N/A         PRESENCE
 Schizo1.0       Schizo 1 Slot 0   N/A     N/A    N/A         PRESENCE
 Schizo0.1       Schizo 0 Slot 1   N/A     N/A    N/A         PRESENCE


FANTRAY   POWER    SPEED     FAN0  FAN1  FAN2  FAN3  FAN4  FAN5
```

```
------    ------   -----     ----  ----  ----  ----  ----  ----
FT0       ON       NORMAL    OK    OK    OK    OK    OK    OK
FT1       ON       NORMAL    OK    OK    OK    OK    OK    OK
FT2       ON       NORMAL    OK    OK    OK    OK    OK    OK
FT3       ON       NORMAL    OK    OK    OK    OK    OK    OK
FT4       ON       NORMAL    OK    OK    OK    OK    OK    OK
FT5       ON       NORMAL    OK    OK    OK    OK    OK    OK
FT6       ON       NORMAL    OK    OK    OK    OK    OK    OK
FT7       ON       NORMAL    OK    OK    OK    OK    OK    OK


POWER          UNIT     AC0      AC1      DC0      DC1      FAN0     FAN1
-----          ----     ---      ---      ---      ---      ----     ----
PS at PS0       OK       OK       FAIL     ON       ON       OK       OK
PS at PS1       OK       OK       FAIL     ON       ON       OK       OK
PS at PS2       OK       OK       FAIL     ON       ON       OK       OK
PS at PS3       OK       OK       FAIL     ON       ON       OK       OK
PS at PS4       OK       OK       FAIL     ON       ON       OK       OK
PS at PS5       OK       OK       FAIL     ON       ON       OK       OK

POWER           VALUE     UNIT    STATUS
---------       -----     ----    ------
PS at PS0
 Current0       13.94     A       N/A
 Current1       0.00      A       N/A
 48VDC          48.60     V       N/A
 Power          677.48    W       N/A
PS at PS1
 Current0       12.35     A       N/A
 Current1       0.00      A       N/A
 48VDC          50.20     V       N/A
 Power          619.97    W       N/A
PS at PS2
 Current0       14.34     A       N/A
 Current1       0.00      A       N/A
 48VDC          49.40     V       N/A
 Power          708.40    W       N/A
PS at PS3
 Current0       12.35     A       N/A
 Current1       0.00      A       N/A
 48VDC          50.20     V       N/A
 Power          619.97    W       N/A
PS at PS4
 Current0       11.95     A       N/A
 Current1       0.00      A       N/A
 48VDC          50.60     V       N/A
 Power          604.67    W       N/A
PS at PS5
 Current0       13.54     A       N/A
 Current1       0.00      A       N/A
 48VDC          50.80     V       N/A
 Power          687.83    W       N/A
Total Power     3918.32   W       N/A
```

EXAMPLE 2 Showing Power Output

```
POWER          UNIT   AC0      AC1      DC0      DC1      FAN0     FAN1
-----          ----   ---      ---      ---      ---      ----     ----
PS at PS0      OK     OK       FAIL     ON       ON       OK       OK
PS at PS1      OK     OK       FAIL     ON       ON       OK       OK
PS at PS2      OK     OK       FAIL     ON       ON       OK       OK
PS at PS3      OK     OK       FAIL     ON       ON       OK       OK
PS at PS4      OK     FAIL     OK       ON       ON       OK       OK
PS at PS5      OK     OK       FAIL     ON       ON       OK       OK

POWER            VALUE     UNIT    STATUS
---------        -----     ----    ------
PS at PS0
 Current0        41.83     A       N/A
 Current1        0.00      A       N/A
 48VDC           49.20     V       N/A
 Power           2058.04   W       N/A
PS at PS1
 Current0        39.84     A       N/A
 Current1        0.00      A       N/A
 48VDC           47.01     V       N/A
 Power           1872.88   W       N/A
PS at PS2
 Current0        39.84     A       N/A
 Current1        0.00      A       N/A
 48VDC           48.21     V       N/A
 Power           1920.69   W       N/A
PS at PS3
 Current0        38.25     A       N/A
 Current1        0.00      A       N/A
 48VDC           48.41     V       N/A
 Power           1851.68   W       N/A
PS at PS4
 Current0        0.00      A       N/A
 Current1        39.84     A       N/A
 48VDC           48.60     V       N/A
 Power           1936.22   W       N/A
PS at PS5
 Current0        41.03     A       N/A
 Current1        0.00      A       N/A
 48VDC           48.60     V       N/A
 Power           1994.06   W       N/A
Total Power      11633.57  W       N/A
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	An invalid domain was used.
2	An invalid command-line option was used.
3	Invalid permission.
4	An internal error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop
Interface Stability	Evolving
Command Output	Unstable

SEE ALSO

`addtag(1M)`

NAME

showfailover - display system controller (SC) failover status or role

SYNOPSIS

showfailover [`-r`] [`-v`]

showfailover `-h`

DESCRIPTION

`showfailover`(1M) enables you to monitor the state of the SC failover mechanism. This command displays the current status of the failover mechanisms. If you do not specify a `-r` option, then the following information is displayed:

```
SC Failover Status: state
```

The failover mechanism can be in one of four states: ACTIVATING, ACTIVE, DISABLED, or FAILED. See the EXTENDED DESCRIPTION section.

OPTIONS

The following options are supported:

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-r` Displays the SC's role as either MAIN, SPARE, or UNKNOWN.

`-v` Verbose. Displays all available command information.

EXTENDED DESCRIPTION

The failover mechanism states are as follows:

`ACTIVATING` Indicates that the failover mechanism is preparing to transition to the ACTIVE state. Failover becomes active when all tests have passed and files have been synchronized.

`ACTIVE` Indicates that the failover mechanism is enabled and is functioning normally.

`DISABLED` Indicates that the failover mechanism has been disabled due to a failover or an operator request (for example, `setfailover off`).

`FAILED` Indicates that the failover mechanism has detected a failure that prevents a failover.

In addition, if the external network has been configured, `showfailover` displays the state of each of the external network interface links monitored by the failover processes. The display format is as follows:

external community name: [UP | DOWN]

A failure string is returned, describing the failure condition. Each failure string has a code associated with it. The codes and associated failure strings are defined in the following table.

String	Explanation
None	No failure.
S-SC EXT NET	The spare SC external network interface has failed.
S-SC CONSOLE BUS	A fault has been detected on the spare SC's console bus path.
S-SC LOC CLK	The spare SC's local clock has failed.
S-SC DISK FULL	The spare SC's system is full.
S-SC IS DOWN	The spare SC is down and/or unresponsive. If this message results from the I2 network/HASRAMs being down then the spare SC could still be running. Login to the spare SC to verify.
S-SC MEM EXHAUSTED	The spare SC's memory/swap space has been exhausted.

String	Explanation
S-SC SMS DAEMON	At least one SMS daemon could not be started/ restarted on the spare SC.
S-SC INCOMPATIBLE SMS VERSION	The spare SC is running a different version of SMS software. Both SCs must be running the same version.
I2 NETWORK/HASRAMS DOWN	Both interfaces for communication between the SCs are down. The main cannot tell what version of SMS is running on the spare nor what its state is. It declares the spare SC down and logs a message to that effect. Dependent services, including file propagation, are unavailable.

Group Privileges Required

You must have platform administrator, platform operator, or platform service privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying a Failover Status That Indicates That Everything is OK

```
sc0:sms-user:> showfailover
SC Failover Status: ACTIVE
C1: UP
```

EXAMPLE 2 Displaying a Failover Status That Indicates That the Spare SC System is Full

```
sc0:sms-user:> showfailover
SC Failover Status: FAILED
S-SC DISK FULL
C1: UP
```

EXAMPLE 3 Displaying the SC Role

```
sc0:sms-user:>  showfailover -r
SPARE
```

EXAMPLE 4 Displaying the Status of All Monitored Components

```
sc0:sms-user:>  showfailover -v
SC Failover Status:     ACTIVE
Status of Shared Memory:
    HASRAM (CSB at CS0): .....................................Good
    HASRAM (CSB at CS1): .....................................Good

Status of xc30p13-sc0:
    Role: ....................................................MAIN
    SMS Daemons: ..........................  ..................Good
    System Clock: ............................................Good
    Private I2 Network: ......................................Good
```

```
    Private HASRAM Network:.................................Good
    Public Network.....................................NOT TESTED
    System Memory: .........................................38.9%
    Disk Status:
        /: .................................................17.4%
Console Bus Status:
        EXB at EX1: ........................................Good
        EXB at EX2: ........................................Good
        EXB at EX4: ........................................Good

Status of xc30p13-sc1:
    Role: ..................................................SPARE
    SMS Daemons: ...........................................Good
    System Clock: ..........................................Good
    Private I2 Network: ....................................Good
    Private HASRAM Network:.................................Good
    Public Network: ...................................NOT TESTED
    System Memory: .........................................34.2%
    Disk Status:
        /: .................................................17.1%
    Console Bus Status:
        EXB at EX1: ........................................Good
        EXB at EX2: ........................................Good
        EXB at EX4: ........................................Good
```

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes:

Attribute Types	Attribute Values
Availability	SUNWSMSop
Interface Stability	Evolving
Command Output	Unstable

SEE ALSO

`setfailover(1M)`

NAME

showkeyswitch - display the position of the virtual keyswitch

SYNOPSIS

showkeyswitch `-d` *domain_indicator* [`-v`]

showkeyswitch `-h`

DESCRIPTION

`showkeyswitch`(1M) displays the position of the virtual keyswitch of the specified domain. The state of each virtual keyswitch is maintained between power cycles of the system controller (SC) or physical power cycling of the power supplies by the `pcd`(1M).

OPTIONS

The following options are supported:

`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* –ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-v`	Verbose. Displays all available command information.

EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator, platform operator, platform service, domain administrator, or configurator privileges for the specified domain to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying the Keyswitch Status for Domain A

```
sc0:sms-user:> showkeyswitch -d A
Virtual keyswitch position: ON
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `setkeyswitch(1M)`, `pcd(1M)`

NAME

showlogs - display message log files or the event logs.

SYNOPSIS

showlogs [-F] [-f *filename*] [-d *domain_indicator*] [-p m|c|s] [-v]

showlogs [-F] [-f *filename*] [-d *domain_indicator*] [-E] [-p e [*event_class*|list|ereport|ena0x<*ena_value*>|uuid<*uuid_value*>|*event_code*] [*number*]]

showlogs -h

DESCRIPTION

showlogs(1M) displays the following:

- Message logs, console logs, or syslog for the platform or a specified domain, depending on the options specified. The default is the platform message log. You must have platform group privileges to run the default; otherwise you receive an error message.
- Portions of the event log, depending on the options specified.

OPTIONS

The following options are supported:

-d *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using addtag(1M). If you specify a domain with the -p e options, the event logs for only that domain are displayed.
-F	Displays only lines that have been appended to the log file since the showlogs command was executed. Similar to the tail -f command. Output continues until interrupted by Ctrl-c.
-f *filename*	Places the output of the showlogs command into a specified file.
-E	Formats and condenses the event log information displayed when specified with the -p e options.
-h	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to -h is ignored.

`-p`

Displays a specific log type: the platform message log, domain console log, domain syslog, or event log.

Valid arguments for `-p` are:

- `m`

  Displays the platform message log. Displaying the platform message log requires platform group privileges.

- `c`

  Displays the domain console log. Displaying the domain console log requires the `-d` option and domain privileges for that domain.

- `s`

  Displays the domain syslog. Displaying the domain syslog requires domain privileges for that domain. `syslog`s forwarded to the system controller (SC) are stored in `/var/opt/SUSNWSMS/adm/anonymous`.

- `e [`*event_class*`|list|ereport|ena0x<`*ena_value*`>|uuid<`*uuid_value*`>|`*event_code*`][`*number*`]]`

  Displays information from the event log. The `-p e` option displays the last entry in the event log by default, unless you specify a *number* of events to be displayed. Displaying event log information requires platform administration or service privileges.

  You can use the following arguments to display certain event log information. This information can be used for additional troubleshooting purposes by your service provider.

`-p`

Displays a specific log type: the platform message log, domain console log, domain syslog, or event log.

Valid arguments for `-p` are:

- `m`

  Displays the platform message log. Displaying the platform message log requires platform group privileges.

- `c`

  Displays the domain console log. Displaying the domain console log requires the `-d` option and domain privileges for that domain.

- `s`

  Displays the domain syslog. Displaying the domain syslog requires domain privileges for that domain. `syslog`s forwarded to the system controller (SC) are stored in `/var/opt/SUSNWSMS/adm/anonymous`.

- `e [`*event_class*`|list|ereport|ena0x<`*ena_value*`>|uuid<`*uuid_value*`>|`*event_code*`][`*number*`]]`

  Displays information from the event log. The `-p e` option displays the last entry in the event log by default, unless you specify a *number* of events to be displayed. Displaying event log information requires platform administration or service privileges.

  You can use the following arguments to display certain event log information. This information can be used for additional troubleshooting purposes by your service provider.

- *event_class* – Displays the last event in the event log that matches a specified ereport event class. The *event_class* is a dot-separated string that identifies the error report event class, for example: `ereport.asic.sdc.porterr.parity_bidi_error`
- `list` – Displays the last list event in the event log. A list event provides a list of faults associated with the hardware error.
- `ereport` – Displays the last error event in the event log. An error report contains data on an unexpected condition or behavior.
- `ena0x`<*ena_value*> – Displays the error event in the event log that matches the Error Numeric Association (ENA) hex value specified, where <*ena_value*> is a hex value. For example, in the ENA specification `ena0xc4fc168cfe77b402`, the hex value is `c4fc168cfe77b402`. The ENA differentiates multiple instances of the same error event.
- `uuid`<*uuid_value*> – Displays the list event in the event log that matches the Universal Unique Identifier (UUID) value specified, where <*uuid_value*> is the UUID value. For example, in the UUID specification `uuid042c2762-982f-11d7-800a-080020fa6556`, the UUID value is `042c2762-982f-11d7-800a-080020fa6556`. The UUID is used by the system to track fault management activity.
- *event_code* – Displays the last list event that matches the specified event code, which is a dash-separated alphanumeric text string that uniquely identifies an event type, for example: `SF15000-8000-A1`. The event code summarizes the fault classes involved in the list events and is used by your service provider to obtain further information concerning the event.
- *number* – An integer that indicates a specific number of events to be displayed. The events displayed match any other arguments specified. If a number is specified with the `-p e` option, the specified number of the events in the event log is displayed. For example, `-p e 5` displays the last five events in the event log.

`-v` Verbose. Displays all available command information.

EXTENDED DESCRIPTION

Group Privileges Required

The group privileges determines the type of `showlogs` output that you can view:

- If you have platform administrator, operator, or service privileges, you can display the platform messages log file.
- If you have platform administrator or service privileges, you can display event log information.
- If you have domain administrator/configurator privileges, you can display only those log files for domains for which you have privileges.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying Platform Message Log to Standard Out

```
sc0:sms-user:> showlogs
Aug 24 14:30:53 2000 xc8-sc0 hwad[104609]: [0 5751139758216 ERR SCCSR.cc
1347] getCrt - Client: 104621.14 has locked - 167
Aug 24 14:30:53 2000 xc8-sc0 hwad[104609]: [0 5751170721148 ERR SCCSR.cc
1362] getCrt - Client: 104621.14 about to unlock - 167.....
```

EXAMPLE 2 Displaying Domain A Message Log to Standard Out

```
sc0:sms-user:> showlogs -d A
Aug 15 14:28:05 2000 xc8-sc0 dsmd[106850]-A(): [0 8500962546702 INFO
Observers.cc 125]  DOMAIN_UP A event has been sent to SYMON, rc = 0.
Aug 15 14:28:05 2000 xc8-sc0 dsmd[106850]-A(): [0 8500963756755 INFO
DomainMon.cc 183]  Start monitoring domain A every 5 second....
```

EXAMPLE 3 Displaying Newly Appended Lines to Domain A Message Log to Standard Out

```
sc0:sms-user:> showlogs -d A -F
Aug 25 14:28:05 2000 xc8-sc0 dsmd[106850]-A(): [0 8500960648900 INFO
Observers.c c 193] DOMAIN_UP A event has been sent to DXS, rc = 0.....
```

EXAMPLE 4 Displaying Domain A Console Log to Standard Out

```
sc0:sms-user:> showlogs -d A -p c
** Domain Server Shutting Down - disconnecting
** Domain Server Shutting Down - disconnecting
Sun Fire 15K/E25K system, using IOSRAM based Console OpenBoot 4.0, 2048
MB memory installed, Serial #10000000. Ethernet address 8:0:20:b8:2d:b1,
Host ID: 80a3e446.
```

EXAMPLE 5 Displaying Domain sms2 Syslog to Standard Out

```
sc0:sms-user:> showlogs -d sms2  -p s
Sep  7 13:51:49 sms2 agent[6629]: [ID 240586 daemon.alert] syslog
Sep 07 13:51:49 agent   {received software termination signal)
Sep  7 13:51:49 sms2 agent[6629]: [ID 985882 daemon.alert] syslog
Sep 07 13:51:49 agent  *** terminating execution ***
Sep  7 13:51:50 sms2 platform[22481]: [ID 345917 daemon.alert] syslog
Sep 07 13:51:50 platform      *** terminating execution ***
Sep  7 14:49:07 sms2 platform[4309]: [ID 745356 daemon.alert] syslog
Sep 07 14:49:07 platform    general parsing error
Sep  7 14:49:07 sms2 platform[4309]: [ID 334248 daemon.alert] syslog
Sep 07 14:49:07 platform    file://localhost/scmonitor-d.x;flags=ro
Sep  7 14:49:07 sms2 platform[4309]: [ID 449452 daemon.alert] syslog
Sep 07 14:49:07 platform    couldn't load file  ...
```

EXAMPLE 6 Displaying the Last Event in the Event Log

```
sc0:sms-user:> showlogs -p e
version: 1
class: list.suspects
fault-diag-time: Mon Jun  6 17:20:15 MDT 2005
DE:
    scheme: diag-engine
    authority:
               product-id: SF15000
               chassis-id: 301AFFFFF
               domain-id: D
    name: sms-de
    version: 1.5
uuid: 0x147e965ebb5ec
code: SF15000-8000-A1
list-sz: 1
fault-events:
              version: 1
              class: fault.board.io.proc
              fault-diag-time: Mon Jun  6 17:20:15 MDT 2005
              DE:
                  scheme: diag-engine
                  authority:
                            product-id: SF15000
                            chassis-id: 301AFFFFF
                            domain-id: D
                  name: sms-de
                  version: 1.5
              ENA-list-sz: 4
              ENA-list: 0x62a6680600000002
                        0x636a74de00000402
                        0x6400cd9800000802
                        0x64be80d400000c02
              FRU:
                   scheme: sf-hc
                   part: 5015396
                   serial: 445883
                   authority:
                            product-id: SF15000
                            chassis-id: 301AFFFFF
                            domain-id: D
```

```
                component: IO15
          resource:
                    scheme: sf-hc
                    part: 5015396
                    serial: 445883
                    authority:
                              product-id: SF15000
                              chassis-id: 301AFFFFF
                              domain-id: D
                    component: io15/p0
          certainty: 100
```

EXAMPLE 7 Displaying the Last Event in the Event Log – Condensed Format

```
sc0:sms-user:> showlogs -E -p e
List Event Class: list.suspects
Diagnosis Engine Name: sms-de
Diagnosis EngineVersion: 1.5
Timestamp: Mon Jun  6 17:20:15 MDT 2004
Code: SF15000-8000-A1

Number of fault events:  1
Fault Event Class: fault.board.io.proc
Fault Event Timestamp: Mon Jun  6 17:20:15 MDT 2005
Domain ID affected by the failure: D
Diagnosis Engine Name: sms-de
Diagnosis Engine Version: 1.5

Indicted resource component: io15/p0
PnSn: 5015396445883

Certainty: 100
FRU: IO15
Part Number Serial Number : 5015396445883
```

EXAMPLE 8 Displaying the Last Three List Events in the Event Log – Condensed Format

```
sc0:sms-user:> showlogs -E -p e list 3
List Event Class: list.suspects
Diagnosis Engine Name: sms-de
Diagnosis EngineVersion: 1.5
Timestamp: Mon Jun  6 17:20:15 MDT 2005
Code: SF15000-8000-A1

Number of fault events:  1
Fault Event Class: fault.board.io.proc
Fault Event Timestamp: Mon Jun  6 17:20:15 MDT 2005
Domain ID affected by the failure: D
Diagnosis Engine Name: sms-de
Diagnosis Engine Version: 1.5

Indicted resource component: io15/p0
PnSn: 5015396445883

Certainty: 100
FRU: IO15
```

```
Part Number Serial Number : 5015396445883

List Event Class: list.suspects
Diagnosis Engine Name: sms-de
Diagnosis EngineVersion: 1.5
Timestamp: Mon Jun  6 13:21:20 MDT 2005
Code: SF15000-8000-A1

Number of fault events:  1
Fault Event Class: fault.board.io.proc
Fault Event Timestamp: Mon Jun  6 13:21:20 MDT 2005
Domain ID affected by the failure: D
Diagnosis Engine Name: sms-de
Diagnosis Engine Version: 1.5

Indicted resource component: io4/p1
PnSn: 5015397028518

Certainty: 100
FRU: IO4
Part Number Serial Number : 5015397028518

List Event Class: list.suspects
Diagnosis Engine Name: sms-de
Diagnosis EngineVersion: 1.5
Timestamp: Mon Jun  6 13:15:18 MDT 2005
Code: SF15000-8000-A1

Number of fault events:  1
Fault Event Class: fault.board.io.proc
Fault Event Timestamp: Mon Jun  6 13:15:18 MDT 2005
Domain ID affected by the failure: D
Diagnosis Engine Name: sms-de
Diagnosis Engine Version: 1.5

Indicted resource component: io17/p1
PnSn: 5015397028488

Certainty: 100
FRU: IO17
Part Number Serial Number : 5015397028488
```

EXAMPLE 9 Displaying the Event Log for a Specific Event Code – Condensed Format

```
sc0:sms-user:> showlogs -E -p e SF15000-8000-H7
List Event Class: list.suspects
Diagnosis Engine Name: sms-de
Diagnosis EngineVersion: 1.5
Timestamp: Mon Jun 6 12:28:12 MDT 2005
Code: SF15000-8000-H7

Number of fault events:  1
Fault Event Class: fault.board.sb.proc
Fault Event Timestamp: Mon Jun 6 12:28:12 MDT 2005
Domain ID affected by the failure: B
Diagnosis Engine Name: sms-de
Diagnosis Engine Version: 1.5

Indicted resource component: sb12/p0
```

```
PnSn: 5014362008423

Certainty: 100
FRU: SB12
Part Number Serial Number : 5014362008423
```

EXAMPLE 10 Displaying an Ereport (Error Report) – Condensed Format

```
sc0:sms-user:> showlogs -E -p e ereport
Error Event Class:
ereport.asic.proc.emushad.isap:_system_request_parity_error_on_incoming
addr.
Domain ID affected by the failure: D
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

FILES

The following files are used:

/var/opt/SUNWSMS/adm/platform/messages	Platform message file
/var/opt/SUNWSMS/adm/*domain_id*/messages	Domain message file
/var/opt/SUNWSMS/adm/*domain_id*/console	Domain console file
/var/opt/SUNWSMS/adm/*domain_id*/syslog	Domain syslog file
/var/opt/SUNWSMS/SMS/adm/platform/events/eventlog	Stores all the hardware-related error and fault events

ATTRIBUTES

See attributes(5) for a description of the following attribute.

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO

tail(1)

NAME

showobpparams - display OpenBoot PROM bring up parameters for a domain

SYNOPSIS

showobpparams `-d` *domain_indicator* [`-v`]

showobpparams `-h`

DESCRIPTION

`showobpparams`(1M) enables a domain administrator to display the virtual `NVRAM` and `REBOOT` parameters passed to OpenBoot PROM by `setkeyswitch`(1M). The `-d` option with *domain_id* or *domain_tag* is required.

OPTIONS

The following options are supported:

`-d` *domain_indicator* — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-v` — Verbose. Displays all available command information.

EXTENDED DESCRIPTION

Group Privileges Required

You must have domain administrator or domain configurator privileges for the specified domain to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying OpenBoot PROM Parameters for Domain A

```
sc0:sms-user:> showobpparams -d a
auto-boot?=false
diag-switch?=true
fcode-debug?=false
use-nvramrc?=false
security-mode=none
```

EXIT STATUS

The following exit values are returned:

`0` — Successful completion.

`>0` — An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`addtag(1M)`, `setkeyswitch(1M)`, `setobpparams(1M)`

NAME

showpcimode - Displays the mode settings for all PCI-X slots on the V2HPCIX board.

SYNOPSIS

showpcimode [location], [location], ...

DESCRIPTION

This command lists the mode settings for all the PCI-X slots on a V2HPCIX I/O board in your server. The settings are specified by the `setpcimode` command. A slot that returns a status of `normal` is running in PCI-X mode. A slot that returns a status of `pci_only` has been forced to run in PCI mode.

If you specify an I/O board that is not a V2HPCIX board, the command returns an error.

Note – You can only run this command from the main SC.

OPERANDS

The following operand is supported:

`[location]` Specifies the board for which you want to display the mode setting. If you use `showpcimode` without specifying a location, the command returns information for all boards which you have permission to view.

`location` takes the form `IO#`, where:

`IO#`	# is an integer from 0 to the maximum expander number on the platform.

EXTENDED DESCRIPTION

Group Privileges Required

You must have domain or platform administrator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

ERROR MESSAGES

`showpcimode` returns error messages when certain error conditions are encountered. The following table lists these error messages and the conditions that cause them to be displayed.

`ERROR: Invalid location: %s`

The location specified in the command line argument is invalid. This error can occur if the location string (`%s`) is not in an acceptable format.

`ERROR: not a V2HPCIX board`

This error is returned when the location specified in the command line argument does not belong to a V2HPCIX board.

`ERROR: unable to get domain board info for %s`

This error appears when `showpcimode` is unable to retrieve PCI slot status for the specified location, or when the command fails to find which domain contains the I/O board corresponding to the specified slot. This error should not occur if SMS is functioning normally.

`ERROR: Solaris Access Control Lists is not properly setup for user`

This error appears when you do not have the appropriate domain/platform privileges to issue the `showpcimode` command for the specified location.

`ERROR: an error occurred while processing %s`

This error appears if an internal error occurred for the command. Some examples of errors include running out of memory, or other types of unexpected errors. This error should not occur if the SC is functioning properly.

`ERROR: unable to retrieve PCI mode setting for %s`

The `showpcimode` command is not applicable to the location specified in the command line argument. This error is only returned when the user specifies slot `C5V0` as the location..

EXAMPLES

EXAMPLE 1 Displaying Settings for One V2HPCIX Board

```
sc0:sms-user:> showpcimode IO3
IO3/C3V0: normal
IO3/C3V1: normal
IO3/C3V2: normal
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See `attributes(5)` for a description of the following attribute:

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`esmd(1M)`, `pcd(1M)`, `setpcimode(1M)`

NAME

showplatform - display the platform type, board available component list, the domain state for each domain, and Capacity on Demand (COD) information.

SYNOPSIS

showplatform [`-d` *domain_indicator*] [`-p` *report*] [`-v`]

showplatform `-h`

DESCRIPTION

`showplatform`(1M) displays the platform type, chassis serial number, available component list, domain state, and Ethernet address for domains. COD information includes the headroom amount, number of installed COD right-to-use (RTU) licenses, and the number of COD RTU licenses reserved for domains. If a *domain_id* or *domain_tag* is specified, only the information for that domain is displayed. If *domain_indicator* and `-p` option are not specified, the available component list, domain states, Ethernet addresses, and COD information for all domains for which you have privileges are displayed.

For platform type, older Sun Fire 12K/15K systems will display Sun Fire 15000, unless they have been manually updated by Sun service. Newer systems will display one of the following:

Sun Fire 12K

Sun Fire 15K

Sun Fire E20K

Sun Fire E25K

OPTIONS

The following options are supported:

`-d` *domain_indicator* — Specifies the domain using one of the following:

domain_id – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-h` — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-p` *report* — Displays specific reports.

Valid reports are:

`domains`	List output is grouped by domain state.
`available`	List output is grouped by domain available component list.
`ethernet`	List output is grouped by domain Ethernet addresses.
`cod`	Capacity on Demand information is displayed.
`csn`	The chassis serial number of the Sun Fire high-end system is displayed.
`platform`	Displays the platform type of the Sun Fire high-end system.

`-v` — Verbose. Displays all available command information.

EXTENDED DESCRIPTION

The domain status is one of the following:

`Unknown`	The domain state could not be determined. For Ethernet addresses, the domain `idprom` image file does not exist. Contact your Sun service representative.
`Powered Off`	The domain is powered off.
`Keyswitch Standby`	The keyswitch for the domain is in STANDBY position.
`Running Domain POST`	The domain power-on self-test is running.
`Loading OBP`	The OpenBoot PROM for the domain is being loaded.

`Booting OBP`	The OpenBoot PROM for the domain is booting.
`Running OBP`	The OpenBoot PROM for the domain is running.
`In OBP Callback`	The domain has been halted and has returned to the OpenBoot PROM.
`Loading Solaris`	The OpenBoot PROM is loading the Solaris software.
`Booting Solaris`	The domain is booting the Solaris software.
`Domain Exited OBP`	The domain OpenBoot PROM has exited.
`OBP Failed`	The domain OpenBoot PROM has failed.
`OBP in sync Callback to OS`	The OpenBoot PROM is in sync callback to the Solaris software.
`Exited OBP`	The OpenBoot PROM has exited.
`In OBP Error Reset`	The domain is in OpenBoot PROM due to an error `reset` condition.
`Solaris Halted, in OBP`	Solaris software is halted and the domain is in OpenBoot PROM.
`OBP Debugging`	The OpenBoot PROM is being used as a debugger.
`Environmental Domain Halt`	The domain was shut down due to an environmental emergency.
`Booting Solaris Failed`	OpenBoot PROM is running, but boot attempt has failed.
`Loading Solaris Failed`	OpenBoot PROM is running, but loading attempt has failed.
`Running Solaris`	Solaris software is running on the domain.
`Solaris Quiesce In-progress`	A Solaris software quiesce is in progress.
`Solaris Quiesced`	Solaris software has quiesced.
`Solaris Resume In-progress`	A Solaris software resume is in progress.
`Solaris Panic`	Solaris software has panicked, and panic flow has started.
`Solaris Panic Debug`	Solaris software has panicked and is entering debugger mode.
`Solaris Panic Continue`	Solaris software has exited debugger mode, and panic flow continues.

`Booting OBP`	The OpenBoot PROM for the domain is booting.
`Running OBP`	The OpenBoot PROM for the domain is running.
`In OBP Callback`	The domain has been halted and has returned to the OpenBoot PROM.
`Loading Solaris`	The OpenBoot PROM is loading the Solaris software.
`Booting Solaris`	The domain is booting the Solaris software.
`Domain Exited OBP`	The domain OpenBoot PROM has exited.
`OBP Failed`	The domain OpenBoot PROM has failed.
`OBP in sync Callback to OS`	The OpenBoot PROM is in sync callback to the Solaris software.
`Exited OBP`	The OpenBoot PROM has exited.
`In OBP Error Reset`	The domain is in OpenBoot PROM due to an error `reset` condition.
`Solaris Halted, in OBP`	Solaris software is halted and the domain is in OpenBoot PROM.
`OBP Debugging`	The OpenBoot PROM is being used as a debugger.
`Environmental Domain Halt`	The domain was shut down due to an environmental emergency.
`Booting Solaris Failed`	OpenBoot PROM is running, but boot attempt has failed.
`Loading Solaris Failed`	OpenBoot PROM is running, but loading attempt has failed.
`Running Solaris`	Solaris software is running on the domain.
`Solaris Quiesce In-progress`	A Solaris software quiesce is in progress.
`Solaris Quiesced`	Solaris software has quiesced.
`Solaris Resume In-progress`	A Solaris software resume is in progress.
`Solaris Panic`	Solaris software has panicked, and panic flow has started.
`Solaris Panic Debug`	Solaris software has panicked and is entering debugger mode.
`Solaris Panic Continue`	Solaris software has exited debugger mode, and panic flow continues.

`Solaris Panic Dump`	Panic dump has started.
`Solaris Halt`	Solaris software is halted.
`Solaris Panic Exit`	Solaris software has exited as a result of a panic.
`Environmental Emergency`	An environmental emergency has been detected.
`Debugging Solaris`	Debugging Solaris software; this is not a hung condition.
`Solaris Exited`	Solaris software has exited.
`Domain Down`	The domain is down and `setkeyswitch` is in the ON, DIAG, or SECURE position.
`In Recovery`	The domain is in the midst of an automatic system recovery.

Group Privileges Required

If you have platform administrator, operator, or service privileges, `showplatform` displays the available component list and board state information on all domains. Otherwise, only information for domains for which you have domain administrator or configurator privileges is displayed.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

Note – An `Unknown` state for an Ethernet address as shown in the following example indicates a missing `idprom` image file for the domain. Contact your Sun service representative.

EXAMPLE 1 Displaying the Available Component List, Domain State, and COD Information for All Domains on a Sun Fire 15K/E25K System

The output shown is what you would see if you had platform privileges.

```
sc0:sms-user:>  showplatform

PLATFORM:
========
Platform Type: Sun Fire 15000

CSN:
====
Chassis Serial Number: 353A00053

COD:
====
Chassis HostID : 5014936C37048
PROC RTUs installed : 8
PROC Headroom Quantity : 0
PROC RTUs reserved for domain A : 4
```

```
PROC RTUs reserved for domain B : 0
PROC RTUs reserved for domain C : 0
PROC RTUs reserved for domain D : 0
PROC RTUs reserved for domain E : 0
PROC RTUs reserved for domain F : 0
PROC RTUs reserved for domain G : 0
PROC RTUs reserved for domain H : 0
PROC RTUs reserved for domain I : 0
PROC RTUs reserved for domain J : 0
PROC RTUs reserved for domain K : 0
PROC RTUs reserved for domain L : 0
PROC RTUs reserved for domain M : 0
PROC RTUs reserved for domain N : 0
PROC RTUs reserved for domain O : 0
PROC RTUs reserved for domain P : 0
PROC RTUs reserved for domain Q : 0
PROC RTUs reserved for domain R : 0


Available Component List for Domains:
=====================================
Available for domain newA:
          SB0 SB1 SB2 SB7
          IO1 IO3 IO6
Available for domain engB:
          No System boards
          No IO boards
Available  for domain domainC:
          No System boards
          IO0 IO1 IO2 IO3 IO4
Available  for domain eng1:
          No System boards
          No IO boards
Available  for domain E:
          No System boards
          No IO boards
Available  for domain domainF:
          No System boards
          No IO boards
Available  for domain dmnG:
          No System boards
          No IO boards
Available  for domain domain H:
          No System boards
          No IO boards
Available  for domain I:
          No System boards
          No IO boards
Available  for domain dmnJ:
          No System boards
          No IO boards
Available  for domain K:
          No System boards
          No IO boards
Available  for domain L:
          No System boards
          No IO boards
Available  for domain M:
          No System boards
          No IO boards
```

```
Available  for domain N:
          No System boards
          No IO boards
Available  for domain O:
          No System boards
          No IO boards
Available  for domain P:
          No System boards
          No IO boards
Available  for domain Q:
          No System boards
          No IO boards
Available  for domain dmnR:
          No System boards
          No IO boards


Domain Ethernet Addresses:
==========================
Domain ID   Domain Tag        Ethernet Address
A           newA              8:0:20:b8:79:e4
B           engB              8:0:20:b4:30:8c
C           domainC           8:0:20:b7:30:b0
D               -             8:0:20:b8:2d:b0
E           eng1              8:0:20:f1:b7:0
F           domainF           8:0:20:be:f8:a4
G           dmnG              8:0:20:b8:29:c8
H               -             8:0:20:f3:5f:14
I               -             8:0:20:be:f5:d0
J           dmnJ              UNKNOWN
K               -             8:0:20:f1:ae:88
L               -             8:0:20:b7:5d:30
M               -             8:0:20:f1:b8:8
N               -             8:0:20:f3:5f:74
O               -             8:0:20:f1:b8:8
P               -             8:0:20:b8:58:64
Q               -             8:0:20:f1:b7:ec
R           dmnR              8:0:20:f1:b7:10


Domain Configurations:
======================
DomainID    Domain Tag     Solaris Nodename     Domain Status
A           newA           -                    Powered Off
B           engB           sun15-b              Keyswitch Standby
C           domainC        sun15-c              Running OBP
D           -              sun15-d              Running Solaris
E           eng1           sun15-e              Running Solaris
F           domainF        sun15-f              Running Solaris
G           dmnG           sun15-g              Running Solaris
H           -              sun15-g              Solaris Quiesced
I           -              -                    Powered Off
J           dmnJ           -                    Powered Off
K           -              sun15-k              Booting Solaris
L           -              -                    Powered Off
M           -              -                    Powered Off
N           -              sun15-n              Keyswitch Standby
O           -              -                    Powered Off
```

```
P           -              sun15-p             Running Solaris
Q           -              sun15-q             Running Solaris
R           dnmR           sun15-r             Running Solaris
```

EXAMPLE 2 Showing the Available Component List and Domain State for Domain `engB`

```
sc0:sms-user:>  showplatform -d engB

PLATFORM TYPE:
====
Platform Type: Sun Fire 15000

COD:
====
PROC RTUs reserved for domain B : 0

Available Component List for Domains:
=====================================
Available  for domain engB:
          SB4 SB5 SB6
          IO4 IO5

Domain Ethernet Addresses:
==========================
Domain ID   Domain Tag        Ethernet Address
B           engB              8:0:20:b4:30:8c

Domain Configurations:
======================
DomainID  Domain Tag    Solaris Nodename    Domain Status
B          engB          sun15-b            Keyswitch Standby
```

EXAMPLE 3 Displaying the Platform for Domain Administrators

The following example shows the domain available component list and state information for all domains for which you have domain administrator or configurator privileges; in this case, domains `engB`, `C`, and `E`.

```
sc0:sms-user:> showplatform

PLATFORM TYPE
=============
Platform type: Sun Fire 15000

COD:
====
PROC RTUs reserved for domain B : 0
PROC RTUs reserved for domain C : 0
PROC RTUs reserved for domain E : 0

Available Component List for Domains:
=====================================
Available  for domain engB:
          SB1 SB2 SB3 SB4 SB5 SB6
          IO1 IO2 IO3 IO4 IO5 IO6 IO7
Available  for domain C:
          SB1 SB2 SB3 SB4 SB5 SB6
          IO1 IO2 IO3 IO4 IO5 IO6 IO7
Available  for domain E:
          SB1 SB2 SB3 SB4 SB5 SB6
          IO1 IO2 IO3 IO4 IO5 IO6 IO7

Domain Ethernet Addresses:
==========================
Domain ID   Domain Tag        Ethernet Address
B           engB              8:0:20:b4:30:8c
C           domainC           8:0:20:b7:30:b0
E           eng1              8:0:20:f1:b7:0

Domain Configurations:
======================
DomainID   Domain Tag    Solaris Nodename     Domain Status
B          engB           sun15-b             Keyswitch Standby
C          domainC        sun15-c             Running OBP
E          eng1           sun15-e             Running Solaris
```

EXAMPLE 4 Showing the Available Component List for Domain `engB`

```
sc0:sms-user:> showplatform -d engB -p available

Available Component List for Domains:
=====================================
Available  for domain engB:
    SB4 SB5 SB6
    IO4 IO5
```

EXAMPLE 5 Displaying Domain Status for Domain `engB`

```
sc0:sms-user:>  showplatform -d engB -p domains

Domain Configurations:
======================
DomainID   Domain Tag    Solaris Nodename     Domain Status
B          engB           sun15-b             Keyswitch Standby
```

EXAMPLE 6 Displaying COD Right-to-Use (RTU) License Reservation for Domain `engB`

```
sc0:sms-user:>  showplatform -d engB -p cod

COD:
====
PROC RTUs reserved for domain B : 0
```

EXAMPLE 7 Displaying All COD Information

The output shown is what you would see if you had platform privileges. If the Chassis HostID value is `Unknown`, power on the centerplane support boards and then run the command `showplatform -p cod` again to display the Chassis HostID. After you power on the centerplane support boards, allow up to one minute for the Chassis HostID information to display in the `showplatform` output.

```
sc0:sms-user:>  showplatform -p cod

COD:
====
Chassis HostID : 5014936C37048
PROC RTUs installed : 8
PROC Headroom Quantity : 0
PROC RTUs reserved for domain A : 4
PROC RTUs reserved for domain B : 0
PROC RTUs reserved for domain C : 0
PROC RTUs reserved for domain D : 0
PROC RTUs reserved for domain E : 0
PROC RTUs reserved for domain F : 0
PROC RTUs reserved for domain G : 0
PROC RTUs reserved for domain H : 0
PROC RTUs reserved for domain I : 0
PROC RTUs reserved for domain J : 0
PROC RTUs reserved for domain K : 0
PROC RTUs reserved for domain L : 0
PROC RTUs reserved for domain M : 0
PROC RTUs reserved for domain N : 0
PROC RTUs reserved for domain O : 0
PROC RTUs reserved for domain P : 0
PROC RTUs reserved for domain Q : 0
PROC RTUs reserved for domain R : 0
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
1	An invalid domain was specified.
2	An invalid command-line option was specified.
3	An incorrect number of domains was specified.
4	The user does not have valid privileges.
5	An error occurred while communicating with the platform configuration daemon (pcd(1M)).
6	An error occurred while communicating with the hardware access daemon (hwad(1M)).
7	An error occurred while communicating with the task management daemon (tmd(1M)).
8	An internal error occurred.
9	An error occurred while communicating with the Capacity on Demand daemon (codd(1M)).

ATTRIBUTES

See attributes(5) for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop
Interface Stability	Evolving
Command Output	Unstable

SEE ALSO

addcodlicense(1M), addtag(1M), hwad(1M), pcd(1M), setupplatform(1M), tmd(1M)

NAME

showxirstate - display CPU dump information only after a reset pulse has been sent to the processors

SYNOPSIS

showxirstate `-d` *domain_indicator* [`-v`]

showxirstate `-f` *filename* [`-v`]

showxirstate `-h`

DESCRIPTION

`showxirstate`(1M) displays CPU dump information, but only after a reset pulse (with the reset -x command) has been sent to the processors. This save-state dump can be used to analyze the cause of abnormal domain behavior. `showxirstate` creates a list of all active processors in that domain and retrieves the save-state information for each processor, including its signature.

If a *domain_indicator* or *filename* is not specified, `showxirstate` returns an error.

OPTIONS

The following options are supported:

`-d` *domain_indicator*	Specifies the domain using one of the following: *domain_id* – ID for a domain. Valid *domain_id*s are A–R and are not case sensitive. *domain_tag* – Name assigned to a domain using `addtag`(1M).
`-f` *filename*	Name of the file containing a previously generated `xir_dump`. The default is `/var/opt/SUNWSMS/adm/`*domain_id*`/dump` and cannot be changed.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-v`	Verbose. Displays all available command information.

EXTENDED DESCRIPTION

Group Privileges Required

You must have domain administrator privileges on the specified domain to run this command. No special privileges are required to read the `xir_dump` files.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying Dump Information for Domain A With 1 CPU

```
sc0:sms-user:> showxirstate -dA

Location:  SB4/P0
XIR Magic   XIR Version 00415645 Buglevel 00000000

XIR Save Total Size 0x58495253 bytes

ver     : 00000000.00000000
tba     : 00000000.00000000

pil     : 0x0
y       : 00000000.00000000
afsr    : 00000000.00000000  afar    : 00000000.00000000
pcontext: 00000000.00000000  scontext: 00000000.00000000
dcu     : 00000000.00000000
dcr     : 00000000.00000000
pcr     : 00000000.00000000
gsr     : 00000000.00000000
softint : 0x0000
pa_watch: 00000000.00000000
va_watch: 00000000.00000000
instbp  : 00000000.00000000
tick:  00000000.00000000  tick_cmpr:  00000000.00000000
stick: 00000000.00000000  stick_cmpr: 00000000.00000000
tl: 0
tt    tstate        tpc                 tnpc
0x00  0x0000000000  00000000.00000000  00000000.00000000
0x00  0x0000000000  00000000.00000000  00000000.00000000
0x00  0x0000000000  00000000.00000000  00000000.00000000
0x00  0x0000000000  00000000.00000000  00000000.00000000
0x00  0x0000000000  00000000.00000000  00000000.00000000

Globals:
R Normal             Alternate          Interrupt          MMU
0 00000000.00000000  00000000.00000000  00000000.00000000
00000000.00000000
1 00000000.00000000  00000000.00000000  00000000.00000000
00000000.00000000
2 00000000.00000000  00000000.00000000  00000000.00000000
00000000.00000000
3 00000000.00000000  00000000.00000000  00000000.00000000
00000000.00000000
4 00000000.00000000  00000000.00000000  00000000.00000000
00000000.00000000
5 00000000.00000000  00000000.00000000  00000000.00000000
00000000.00000000
6 00000000.00000000  00000000.00000000  00000000.00000000
00000000.00000000
7 00000000.00000000  00000000.00000000  00000000.00000000
00000000.00000000

wstate: 0x00
cansave:    0  cleanwin: 0
canrestore: 0  otherwin: 0

Register Windows:
Window 0
```

```
R Locals              Ins
0 00000000.00000000  00000000.00000000
1 00000000.00000000  00000000.00000000
2 00000000.00000000  00000000.00000000
3 00000000.00000000  00000000.00000000
4 00000000.00000000  00000000.00000000
5 00000000.00000000  00000000.00000000
6 00000000.00000000  00000000.00000000
7 00000000.00000000  00000000.00000000

Window 1

R Locals              Ins
0 00000000.00000000  00000000.00000000
1 00000000.00000000  00000000.00000000
2 00000000.00000000  00000000.00000000
3 00000000.00000000  00000000.00000000
4 00000000.00000000  00000000.00000000
5 00000000.00000000  00000000.00000000
6 00000000.00000000  00000000.00000000
7 00000000.00000000  00000000.00000000

Window 2
R Locals              Ins

0 00000000.00000000  00000000.00000000
1 00000000.00000000  00000000.00000000
2 00000000.00000000  00000000.00000000
3 00000000.00000000  00000000.00000000
4 00000000.00000000  00000000.00000000
5 00000000.00000000  00000000.00000000
6 00000000.00000000  00000000.00000000
7 00000000.00000000  00000000.00000000

Window 3
R Locals              Ins

0 00000000.00000000  00000000.00000000
1 00000000.00000000  00000000.00000000
2 00000000.00000000  00000000.00000000
3 00000000.00000000  00000000.00000000
4 00000000.00000000  00000000.00000000
5 00000000.00000000  00000000.00000000
6 00000000.00000000  00000000.00000000
7 00000000.00000000  00000000.00000000

Window 4
R Locals              Ins

0 00000000.00000000  00000000.00000000
1 00000000.00000000  00000000.00000000
2 00000000.00000000  00000000.00000000
3 00000000.00000000  00000000.00000000
4 00000000.00000000  00000000.00000000
5 00000000.00000000  00000000.00000000
6 00000000.00000000  00000000.00000000
7 00000000.00000000  00000000.00000000

Window 5
R Locals              Ins
```

```
0 00000000.00000000  00000000.00000000
1 00000000.00000000  00000000.00000000
2 00000000.00000000  00000000.00000000
3 00000000.00000000  00000000.00000000
4 00000000.00000000  00000000.00000000
5 00000000.00000000  00000000.00000000
6 00000000.00000000  00000000.00000000
7 00000000.00000000  00000000.00000000

Window 6
R Locals             Ins

0 00000000.00000000  00000000.00000000
1 00000000.00000000  00000000.00000000
2 00000000.00000000  00000000.00000000
3 00000000.00000000  00000000.00000000
4 00000000.00000000  00000000.00000000
5 00000000.00000000  00000000.00000000
6 00000000.00000000  00000000.00000000
7 00000000.00000000  00000000.00000000

Window 7
R Locals             Ins

0 00000000.00000000  00000000.00000000
1 00000000.00000000  00000000.00000000
2 00000000.00000000  00000000.00000000
3 00000000.00000000  00000000.00000000
4 00000000.00000000  00000000.00000000
5 00000000.00000000  00000000.00000000
6 00000000.00000000  00000000.00000000
7 00000000.00000000  00000000.00000000

nest_save_ptr: 00000000
XIR Nest Version 00000000 Buglevel 00000000
XIR Nest nest_count 0 save_block 88

tick:  00000000.00000000
stick: 00000000.00000000

tl: 73
tt    tstate        tpc                 tnpc

0x00  0x0000000000  00000000.00000000  00000000.00000000
0x00  0x0000000000  00000000.00000000  00000000.00000000
```

```
0x00  0x0000000000  00000000.00000000  00000000.00000000
0x00  0x0000000000  00000000.00000000  00000000.00000000
0x00  0x0000000000  00000000.00000000 00000000.00000000

Processor signatures:
SB0/P0: Solaris/Run/Null (4f530100)
SB0/P1: Solaris/Run/Null (4f530100)
SB0/P2: Solaris/Run/Null (4f530100)
SB0/P3: Solaris/Run/Null (4f530100)
IO0/P0: Solaris/Run/Null (4f530100)
IO0/P1: Solaris/Run/Null (4f530100)
SB1/P0: OBP/???/Null (4f421300)
SB1/P1: Solaris/Run/Null (4f530100)
SB1/P2: OBP/Exit/Error Reset Reboot (4f420209)
SB1/p3: Solaris/Run/Null (4f530100)
IO1/P0: Solaris/Run/Null (4f530100)
IO1/P1: Solaris/Run/Null (4f530100)
```

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`reset(1M)`

NAME

smsbackup - back up the SMS environment

SYNOPSIS

smsbackup *directory_name*

smsbackup `-h`

DESCRIPTION

`smsbackup`(1M) creates a `cpio`(1) archive of files that maintain the operational environment of SMS. To create a complete and accurate backup, turn off SMS before running `smsbackup`. For information about manually starting and stopping SMS refer to the *System Management Services (SMS) 1.6 Installation Guide.*

Whenever changes are made to the SMS environment—for example, by shutting down a domain— you must run `smsbackup` again to maintain a current backup file for the system controller.

The name of the backup file is `sms_backup.`*X.X*`.cpio`, where *X.X* represents the active version from which the backup was taken.

Restore SMS backup files using the `smsrestore`(1M) command.

If any errors occur, `smsbackup` writes error messages to `/var/sadm/system/logs/smsbackup` if `/var/sadm/system/logs` exists and `/var/tmp` if it does not exist.

OPTIONS

The following option is supported

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

OPERANDS

The following operands are supported:

directory_name Name of the directory in which the backup file is created. This file can reside in any directory on the system, connected network or tape device to which you have read/write privileges. If no *directory_name* is specified, a backup file is created in `/var/tmp`. The *directory_name* does not require the absolute path name for the file.

The *directory_name* specified must be mounted as a UFS file system. Specifying a TMPFS file system, such as `/tmp`, causes `smsbackup` to fail. If you are not certain that your *directory_name* is mounted as a UFS file system, type:

`/usr/bin/df -F ufs` *directory_name*

A UFS file system returns directory information. Any other type of file system returns a warning.

EXTENDED DESCRIPTION

Group Privileges Required

You must have superuser privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Backing Up SMS to `/var/opt/SUNWSMS/bkup`

```
sc0:sms-user:> smsbackup  /var/opt/SUNWSMS/bkup
```

EXAMPLE 2 Backing Up SMS to Tape Device 0

```
sc0:sms-user:> smsbackup /dev/rmt/0
```

EXAMPLE 3 Backing Up SMS to TMPFS System

```
sc0:sms-user:> smsbackup /tmp
ERROR: smsbackup fails to backup to /tmp, a TMPFS
file system.  Please specify a directory that is
mounted on a UFS file system.
ABORT:
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following file is used by this command:

`/var/sadm/system/logs/smsbackup`	`smsbackup` log file

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`smsrestore(1M)`

NAME

smsconfig - configures the SMS environment

SYNOPSIS

smsconfig -m

smsconfig -m I1 [*domain_id* | sc | netmask]

smsconfig -m I2 [sc0 | sc1 | netmask]

smsconfig -m L

smsconfig -g

smsconfig -a -u *username* -G *platform_role* platform

smsconfig -r -u *username* -G *platform_role* platform

smsconfig -a -u *username* -G *domain_role* *domain_id*

smsconfig -r -u *username* -G *domain_role* *domain_id*

smsconfig -l *domain_id*

smsconfig -l platform

smsconfig -v

smsconfig -h

DESCRIPTION

smsconfig(1M) configures the SMS environment in a three areas: network management, security, and user group privileges.

smsconfig configures and modifies host name and IP address settings used by the MAN daemon, mand(1M). For each network, smsconfig can singularly set one or more *interface* designations within that network. By default, smsconfig steps through the configuration of both internal networks and the external community network.

Note – Once you have configured or changed the configuration of the MAN network you *must* reboot the system controller (SC) in order for the changes to take effect.

To configure an individual network, append the *net_id* to the command line. Management network *net_id*s are designated I1, I2, and L. Configure a single *interface* within an enterprise network by specifying both the desired *interface* and its *net_id*. Any changes made to the network configuration on one SC using smsconfig -m must be run on the other SC. Network configurations files are not automatically propagated.

For security purposes, SMS disables forwarding, broadcast, and multicast by setting the appropriate ndd utility variables upon startup.

`smsconfig` configures the UNIX groups used by SMS to describe user privileges. SMS uses a default set of UNIX groups installed locally on each SC. `smsconfig` enables you to customize those groups using the `-g` option. For more information refer to the *System Management Services (SMS) 1.6 Installation Guide*.

`smsconfig` also adds users to SMS groups and configures domain and platform administrative privileges. `smsconfig` sets access control list (ACL) attributes on SMS directories.

Note – Do *not* manually edit the `/etc/group` SMS file entries to add or remove users. Otherwise user access will be compromised.

OPTIONS

The following options are supported:

`-a`	Adds a user to an SMS group and provides read, write, and execute access for a domain or for the platform directories. You must specify a valid *username*, SMS group, and if applicable, a *domain_id*
`-G`	Indicates an SMS group. No group name is case sensitive.
`-g`	Configures the UNIX groups used by SMS to describe user privileges.
`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-l`	Lists all users with access to the specified SMS domain or platform.
`-m`	Configures all interfaces for all enterprise networks and the external community.
`-m I1`	Configures all interfaces for enterprise network `I1`. Network designation is not case sensitive. You can exclude a domain from the I1 network configuration by using the word `NONE` as the MAN *hostname*. This applies to the I1 network only.
`-m I2`	Configures all interfaces for enterprise network `I2`. Network designation is not case sensitive.
`-m L`	Configures all interfaces for the external community network. Network designation is not case sensitive.
`-r`	Removes a user from an SMS group and denies read, write, and execute access for a domain or for the platform directories. You must specify a valid *username*, SMS group and if applicable, a *domain_id*.

`-s`	This interface is deprecated. `rsh/ssh` is no longer required.
`-u` *username*	Indicates user login name.
`-v`	Displays the network configuration.

OPERANDS

The following operands are supported:

domain_id	ID for a domain. Valid *domain_id*s are A–R and are not case sensitive.

domain_role	Valid *domain_role*s are:
	`admn`
	`rcfg`

`platform`	Specifies the Sun Fire high-end platform and platform-specific directories. The platform name must begin with a letter, can contain numbers, letters, and the "-" symbol, and must not exceed 15 characters. (It must comply with RFC-921.) The platform name is used as the default prefix for hostnames of internal network interfaces. In Example 1, below, the platform name is sun15.

platform_role	Valid *platform_role*s are:
	`admn`
	`oper`
	`svc`

`SC0, SC1`	Interface designation for the Sun Fire high end systems SC. Interface designations are not case sensitive.
`netmask`	A 32-bit number that masks or screens out the network part of an IP address in a subnet so that only the host computer part of the address remains visible. Commonly displayed in decimal numbers; for example, 255.255.255.0 is a common netmask in a Class C subnet. `netmask` is not case sensitive.

EXTENDED DESCRIPTION

Group Privileges Required

You must have superuser privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

Setting Up the MAN Network

You must configure all interfaces in the `MAN` network. This example steps through all the prompts needed to completely set up all three enterprise networks using IPv4.

Caution — The IP addresses shown in the following examples are examples only. Refer to your *Sun Fire System Site Planning Guide* for valid IP addresses for your network. Using invalid network IP addresses could, under certain circumstances, make your system unbootable!

There are no prompts for `netmasks`, and `/etc/ipnodes` are modified in addition to `/etc/hosts`.

On the CP1500 board, the default NICs for community C1 are hme0 and eri1. On the CP2140 board, they are eri0 and eri3. The CP2140 board has no hme devices. IP addresses on the external network for `failover`, hme0, eri0, eri1, and eri3 on each SC must be unique. The floating IP address is the same on both SCs.

By default, the `I1` network settings are derived from the base network address entered for that network. You can exclude a domain from the I1 network configuration by using the word `NONE` as the MAN *hostname*. See EXAMPLE 4. For more information, refer to the *System Management Services (SMS) 1.6 Installation Guide.*

Once you have configured the MAN network, you *must* reboot the SC.

EXAMPLE 1 Configuring the MAN Network

```
sc0:# smsconfig -m
The platform name identifies the entire host machine to the SMS software.
The platform name occupies a different name space than domain names
(hostnames of bootable systems).

What is the name of the platform this SMS will service? sun15

Configuring the External Network for Community C1

Do you want to define this Community? [y,n] y
Two network interfaces controllers (NICs) are required for IPMP network
failover.
Enter NICs associated with community C1 [hme0 eri1]: [Return]

Enter hostname for hme0 [sun15-sc0-hme0]:[Return]
Enter IP address for sun15-sc0-hme0: 10.1.1.52

Enter hostname for eri1 [sun15-sc0-eri1]:[Return]
Enter IP address for sun15-sc0-eri1: 10.1.1.53

The Logical/Floating IP hostname and address will "float" over to
whichever system controller (SC0 or SC1) is acting as the main SC.

Enter Logical/Floating IP hostname for community C1 [sun15-sc-C1]:
 [Return]
Enter IP address for sun15-sc-C1: 10.1.1.50
```

```
Enter Netmask for community C1: 255.255.255.0


Enter hostname for community C1 failover address [sun15-sc0-C1-
failover]:[Return]
Enter IP address for sun15-sc0-C1-failover: 10.1.1.51

Hostname                IP Address (platform=sun15)
--------                ----------
sun15-sc-C1             10.1.1.50
sun15-sc0-C1-failover   10.1.1.51
sun15-sc0-eri0          10.1.1.52
sun15-sc0-eri3          10.1.1.53

Do you want to:
 1) Accept these network settings.
 2) Edit these network settings.
 3) Delete these network settings and go onto the next community? [y,n] y

Configuring the External Network for Community C2

Do you want to define this Community? [y,n] n

Configuring I1 Management Network - 'I1' is the Domain to SC MAN.
MAN I1 Network Identification
Enter the IP network number (base address) for the I1 network: 10.2.1.0
Enter the netmask for the I1 MAN network [ 255.255.255.224 ]: [Return}

Hostname        IP Address   platform=sun15)
--------        ----------
netmask-i1      255.255.255.224
sun15-sc-i1     10.2.1.1
sun15-a         10.2.1.2
sun15-b         10.2.1.3
sun15-c         10.2.1.4
sun15-d         10.2.1.5
sun15-e         10.2.1.6
sun15-f         10.2.1.7
sun15-g         10.2.1.8
sun15-h         10.2.1.9
sun15-i         10.2.1.10
sun15-j         10.2.1.11
sun15-k         10.2.1.12
sun15-l         10.2.1.13
sun15-m         10.2.1.14
sun15-n         10.2.1.15
sun15-o         10.2.1.16
sun15-p         10.2.1.17
sun15-q         10.2.1.18
sun15-r         10.2.1.19

Do you want to accept these network settings? [y,n] y

Configuring I2 Management Network - 'I2' is for SC to SC MAN.
MAN I2 Network Identification
Enter the IP network number (base address) for the I2 network: 10.3.1.0
Enter the netmask for the I2 MAN network [ 255.255.255.252 ]:[Return]

Hostname           IP Address(platform=sun15)
--------           ----------
```

```
netmask-i2          255.255.255.252
sun15-sc0-i2        10.3.1.1
sun15-sc1-i2        10.3.1.2

Do you want to accept these settings? [y,n] y
Creating /.rhosts to facilitate file propagation ... done.

MAN Network configuration modified!
Changes will take effect on next reboot.

The following changes are about to be applied to the "/etc/hosts" hosts
file.
----------------------
ADD: 10.2.1.2   sun15-a #smsconfig-entry#
ADD: 10.2.1.3   sun15-b #smsconfig-entry#
ADD: 10.2.1.4   sun15-c #smsconfig-entry#
ADD: 10.2.1.5   sun15-d #smsconfig-entry#
ADD: 10.2.1.6   sun15-e #smsconfig-entry#
ADD: 10.2.1.7   sun15-f #smsconfig-entry#
ADD: 10.2.1.8   sun15-g #smsconfig-entry#
ADD: 10.2.1.9   sun15-h #smsconfig-entry#
ADD: 10.2.1.10  sun15-i #smsconfig-entry#
ADD: 10.2.1.11  sun15-j #smsconfig-entry#
ADD: 10.2.1.12  sun15-k #smsconfig-entry#
ADD: 10.2.1.13  sun15-l #smsconfig-entry#
ADD: 10.2.1.14  sun15-m #smsconfig-entry#
ADD: 10.2.1.15  sun15-n #smsconfig-entry#
ADD: 10.2.1.16  sun15-o #smsconfig-entry#
ADD: 10.2.1.17  sun15-p #smsconfig-entry#
ADD: 10.2.1.18  sun15-q #smsconfig-entry#
ADD: 10.2.1.19  sun15-r #smsconfig-entry#
ADD: 10.2.1.1   sun15-sc-i1 #smsconfig-entry#
ADD: 10.1.1.50  sun15-sc-C1 #smsconfig-entry#
ADD: 10.1.1.51  sun15-sc0-C1-failover #smsconfig-entry#
ADD: 10.1.1.52  sun15-sc0-hme0 #smsconfig-entry#
ADD: 10.1.1.53  sun15-sc0-eri1 #smsconfig-entry#
ADD: 10.3.1.1   sun15-sc0-i2 #smsconfig-entry#
ADD: 10.3.1.2   sun15-sc1-i2 #smsconfig-entry#
----------------------
Update the hosts file, "/etc/hosts", with these changes? [y,n] y
Hosts file "/etc/hosts" has been updated.

The following information is about to be applied to the "/etc/netmasks"
file.
----------------------
ADD network: 10.1.1.50, mask:  255.255.255.0
ADD network: 10.2.1.0, mask: 255.255.255.224
ADD network: 10.3.1.0, mask: 255.255.255.252
----------------------
Update the netmasks file, "/etc/netmasks", with these changes? [y,n] y
Netmasks file "/etc/netmasks" has been updated.
smsconfig complete. Log file is /var/sadm/system/logs/smsconfig
sc#
```

EXAMPLE 2 Configuring the I2 Network

```
sc0: # smsconfig -m I2
Configuring I2 Management Network - 'I2' is for SC to SC MAN
Which System Controller are you configuring [choose 0 or 1]: 0.
Hostname          IP Address (platform=sun15)
--------          ----------
netmask-i2        255.255.255.252
sun15-sc0-i2      10.3.1.1
sun15-sc1-i2      10.3.1.2
Do you want to accept these network settings? [y,n] n
MAN I2 Network Identification
Enter the IP network number (base address) for the I2 network: 172.16.0.0
Enter the netmask for the I2 MAN network [ 255.255.255.252 ]: [Return]
Hostname          IP Address (platform=sun15)
--------          ----------
netmask-i2        255.255.255.252
sun15-sc0-i2      172.16.0.1
sun15-sc1-i2      172.16.0.2
Do you want to accept these network settings? [y,n] y
Creating /.rhosts to facilitate file propagation ... done.

MAN Network configuration modified!
Changes will take effect on the next reboot.
The following changes are about to be applied to the "/etc/hosts" hosts
file.
----------------------
ADD: 172.16.0.1   sun15-sc0-i2 #smsconfig-entry#
ADD: 172.16.0.2   sun15-sc1-i2 #smsconfig-entry#
----------------------
Update the hosts file, "/etc/hosts". with these changes [y,n] y
Hosts file "/etc/hosts" has been updated.

The following information is about to be applied to the "/etc/netmasks"
file.
---------------------
ADD network: 172.16.0.0, mask: 255.255.255.252
---------------------
Update the netmasks file, "/etc/netmasks", with these changes? [y,n] y
Netmasks file "/etc/netmasks" has been updated.

sc#
```

EXAMPLE 3 Configuring Internal Host Name and IP Address, SC to Domain B, on the I1 Network

```
sc0: # smsconfig -m I1 B

Enter the MAN hostname for DB-I1 [ sun15-b ]: domainB-i1
I could not automatically determine the IP address of domainB-i1.

Please enter the IP address of domainB-i1: 10.2.1.20

You should make sure that this host/IP address is set up properly in the
 /etc/inet/hosts file or in your local name service system.

Network: I1 (DB-I1)  Hostname: domainB-i1  IP Address: 10.2.1.20

Do you want to accept these settings? [y,n] y

Creating /.rhosts to facilitate file propagation ... done.

MAN Network configuration modified!
Changes will take effect on the next reboot.

The following changes are about to be applied to the "/etc/hosts" hosts
file.
----------------------
ADD: 10.2.1.20   domainB-i1 #smsconfig-entry#
----------------------
Update the hosts file, "/etc/hosts", with these changes? [y,n] y
Hosts file "/etc/hosts" has been updated.

sc#
```

EXAMPLE 4 Excluding Domain D From the I1 Network

By excluding a domain, you will no longer be able to perform DR operations (`rcfgadm`) from the SC on that domain. You can still perform DR operations (`cfgadm`) on the domain itself. Refer to the *Sun Fire High-End Systems Dynamic Reconfiguration (DR) User Guide* for more information.

```
sc0: # smsconfig -m I1 D
Enter the MAN hostname for DB-I1 [ sun15-b ]: NONE
Network: I1 (DB-I1)
Hostname: NONE  IP Address: None

Do you want to accept these settings? [y,n] y

Creating /.rhosts to facilitate file propagation ... done.

sc#
```

EXAMPLE 5 Configuring Non-Default Groups

In this example, all domain administrator and domain reconfiguration groups are left as the default groups.

```
sc0: # smsconfig -g
1) Edit current configuration
2) Restore default groups
3) Quit

Select one of the above options: 1

NOTE: In order to configure a new group the group must already exist.

The Platform Administrator group has configuration control, a means to
get environmental status, the ability to assign boards to domains, power
control and other generic service processor functions.
Enter the name of the Platform Administrator group [platadmn]? zeus

The Platform Operator group has a subset of the platform privileges,
limited generally to platform power control and platform status.

Enter the name of the Platform Operator group [platoper]? poseidon

The Platform Service group posses platform service command privileges in
addition to limited platform control and platform configuration status
privileges

Enter the name of the Platform Service group [platsvc]? kronos

The Domain Administrator group posses domain control and status, and
console access privileges (for the respective domain), but does not
posses platform wide control or platform resource allocation privileges.

Enter the name of the Domain A Administrator group [dmnaadmn]? [Return]

Enter the name of the Domain B Administrator group [dmnbadmn]? [Return]
Enter the name of the Domain C Administrator group [dmncadmn]? [Return]
Enter the name of the Domain D Administrator group [dmndadmn]? [Return]
Enter the name of the Domain E Administrator group [dmneadmn]? [Return]
Enter the name of the Domain F Administrator group [dmnfadmn]? [Return]
Enter the name of the Domain G Administrator group [dmngadmn]? [Return]
Enter the name of the Domain H Administrator group [dmnhadmn]? [Return]
Enter the name of the Domain I Administrator group [dmniadmn]? [Return]
Enter the name of the Domain J Administrator group [dmnjadmn]? [Return]
Enter the name of the Domain K Administrator group [dmnkadmn]? [Return]
Enter the name of the Domain L Administrator group [dmnladmn]? [Return]
Enter the name of the Domain M Administrator group [dmnmadmn]? [Return]
Enter the name of the Domain N Administrator group [dmnnadmn]? [Return]
Enter the name of the Domain O Administrator group [dmnoadmn]? [Return]
Enter the name of the Domain P Administrator group [dmnpadmn]? [Return]
Enter the name of the Domain Q Administrator group [dmnqadmn]? [Return]
Enter the name of the Domain R Administrator group [dmnradmn]? [Return]

The Domain Reconfiguration group posses a subset of the Domain
Administration group privileges. This group has no domain control other
than board power and reconfiguration (for the respective domain).
```

```
Enter the name of the Domain A Reconfiguration group [dmnarcfg]? [Return]
Enter the name of the Domain B Reconfiguration group [dmnbrcfg]? [Return]
Enter the name of the Domain C Reconfiguration group [dmncrcfg]? [Return]
Enter the name of the Domain D Reconfiguration group [dmndrcfg]? [Return]
Enter the name of the Domain E Reconfiguration group [dmnercfg]? [Return]
Enter the name of the Domain F Reconfiguration group [dmnfrcfg]? [Return]
Enter the name of the Domain G Reconfiguration group [dmngrcfg]? [Return]
Enter the name of the Domain H Reconfiguration group [dmnhrcfg]? [Return]
Enter the name of the Domain I Reconfiguration group [dmnircfg]? [Return]
Enter the name of the Domain J Reconfiguration group [dmnjrcfg]? [Return]
Enter the name of the Domain K Reconfiguration group [dmnkrcfg]? [Return]
Enter the name of the Domain L Reconfiguration group [dmnlrcfg]? [Return]
Enter the name of the Domain M Reconfiguration group [dmnmrcfg]? [Return]
Enter the name of the Domain N Reconfiguration group [dmnnrcfg]? [Return]
Enter the name of the Domain O Reconfiguration group [dmnorcfg]? [Return]
Enter the name of the Domain P Reconfiguration group [dmnprcfg]? [Return]
Enter the name of the Domain Q Reconfiguration group [dmnqrcfg]? [Return]
Enter the name of the Domain R Reconfiguration group [dmnrrcfg]? [Return]

Configuration complete.

Select one of the above options:
1) Edit current configuration
2) Restore default groups
3) Quit
Select one of the above options: 3

sc#
```

EXAMPLE 6 Displaying the Network Configuration

```
sc0: # smsconfig -v
NETWORK CONFIGURATION

Platform name: xc

Internal I1 Management Network - Used for Domain to SC
communication

    MAN I1 SC Hostname: sun15-sc-i1
    MAN I1 SC IP Address: 192.168.103.1
    MAN I1 Network Mask: 255.255.255.224

    Domain Hostname     IP Address
    -----------------------------
            sun15-a  192.168.103.2
            sun15-b  192.168.103.3
            sun15-c  192.168.103.4
            sun15-d  192.168.103.5
            sun15-e  192.168.103.6
            sun15-f  192.168.103.7
            sun15-g  192.168.103.8
            sun15-h  192.168.103.9
            sun15-i 192.168.103.10
            sun15-j 192.168.103.11
            sun15-k 192.168.103.12
            sun15-l 192.168.103.13
            sun15-m 192.168.103.14
            sun15-n 192.168.103.15
```

```
            sun15-o 192.168.103.16
            sun15-p 192.168.103.17
            sun15-q 192.168.103.18
            sun15-r 192.168.103.19

Internal I2 Management Network - Used for SC to SC
communication

    MAN I2 Network Mask: 255.255.255.252

        SC Hostname     IP Address
    ------------------------------
       sun15-sc0-i2 192.168.103.33
       sun15-sc1-i2 192.168.103.34

No External Network Configured.

Fri Jun  3 11:00:47 PDT 2005
smsconfig complete.
```

EXAMPLE 7 Adding a User to the Domain Administrator Group and Configuring Access to the Domain B Directories

You must specify a valid user name and valid SMS group and domain.

```
sc0: # smsconfig -a -u fdjones -G admn B
fdjones has been added to the dmnBadmn group.
All privileges to domain B have been applied.
```

EXAMPLE 8 Adding a User to the Domain Configurator Group and Configuring Access to the Domain C Directories

You must specify a valid user name and valid SMS group and domain.

```
sc0: # smsconfig -a -u fdjones -G rcfg C
fdjones has been added to the dmnCrcfg group.
All privileges to domain C have been applied.
```

EXAMPLE 9 Configuring Access to the Platform Directories

You must specify a valid user name and valid SMS group and the platform.

```
sc0: # smsconfig -a -u jtd -G svc platform
jtd has been added to the platsvc group.
All privileges to the platform have been applied.
```

EXAMPLE 10 Displaying Users With Access to the Domain C Directories

```
sc0: # smsconfig -l C
fdjones
shea
```

EXAMPLE 11 Displaying Users With Access to the Platform Directories

```
sc0: # smsconfig -l platform
fdjones
jtd
```

EXAMPLE 12 Removing User Access to the Domain C Directories

You must specify a valid username and valid SMS group. Any user who belongs to more than one group with access to a domain must be removed from all groups before directory access is denied.

```
sc0: # smsconfig -r -u fdjones -G rcfg C
fdjones has been removed from the dmnCrcfg group.
fdjones belongs to the dmnCadmn group
Access to domain C remains unchanged.


sc0: # smsconfig -r -u fdjones -G admn C
fdjones has been removed from the dmnCadmn group.
All access to domain C is now denied.
```

EXAMPLE 13 Configuring Using an Invalid Group name

You must specify a valid SMS group.

```
sc0: # smsconfig -a -u fdjones -G staff D
ERROR:  group staff does not exist
ABORTING.
```

EXAMPLE 14 Mixing Groups and Designations

You must specify group names with the correct area designations. The `admn` group works with either designation.

```
sc0: # smsconfig -a -u fdjones -G rcfg platform
ERROR:  group rcfg cannot access the platform
ABORTING.
```

```
sc0: # smsconfig -a -u fdjones -G oper D
ERROR:  group oper cannot access a domain
ABORTING.
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following configuration files are required:

`/etc/hostname.scman0`	MAN Ethernet interface file
`/etc/hostname.scman1`	MAN Ethernet interface file
`/etc/opt/SUNWSMS/config/MAN.cf`	MAN daemon configuration file

Note – `MAN.cf` is an internal SMS system file and should *not* be modified except by authorized Sun Microsystems personnel.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop
Interface Stability	Evolving

SEE ALSO

`mand(1M)`, `ndd(1M)`, `scp(1)`

NAME

smsconnectsc - accesses a remote SC console

SYNOPSIS

smsconnectsc [-y|n]

smsconnectsc -h

DESCRIPTION

smsconnectsc creates a remote tip console session from a local SC in order to reach a hung remote SC console.

smsconnectsc enables the bit that connects the local SC's port B to the remote SC's RS-232 port A when you are logged in to the local SC. The remote SC is the SC that is hanging. Once the tty connection is enabled, smsconnectsc invokes a tip console session to the remote SC. Using the tip console session, you can do whatever needs to be done to the remote SC.

smsconnectsc works in the absence of an external connection to the remote SC. If the remote SC has an active external connection to port A, then smsconnectsc fails and the session usually hangs. To exit, type:

~.

After you finish, there are several ways to end the session, depending on whether you logged in to the local SC using telnet or rlogin. See the EXTENDED DESCRIPTION section.

Using smsconnectsc as your permanent console connection to your system controller is not recommended; it should only be used when there is a problem with your external console connection.

OPTIONS

The following options are supported.

-h	Help. Displays usage descriptions.
	Note – Use alone. Any option specified in addition to -h is ignored.
-n	Automatically answers “no” to all prompts.
-y	Automatically answers “yes” to all prompts.

EXTENDED DESCRIPTION

Usage

In the `tip` console window established by `smsconnectsc`, a tilde (~) that appears as the first character of a line is interpreted as an escape signal that directs the `tip` console to perform the following action:

`~.` Disconnect the `tip` session.

If you have established a `telnet` session to the local SC this disconnects the `tip` session and you remain logged in to the local SC.

If you have established an `rlogin` session in to the local SC, this disconnects the `tip` session and *also* disconnects your `rlogin` session.

Note – The tilde does not appear on the display until after you have pressed the period.

`~~.` Disconnect `tip` session.

~~. Works only with rlogin. If you have established a telnet session in to the local SC, you receive the following error message:

```
~.: Command not found
```

If you have established an `rlogin` session in to the local SC, this disconnects the `tip` session and you remain logged in to the local SC.

Note – The first tilde does not appear on the display screen. The second tilde does not appear until after you have pressed the period.

`rlogin` also processes `tilde-escape` sequences whenever a tilde is displayed at the beginning of a new line. If you need to send tilde sequence at the beginning of a line and you are using `rlogin`, use two tildes (the first escapes the second for rlogin). Alternatively, do not enter a tilde at the beginning of a line when running inside of `rlogin`. If you use a `kill -9` command to terminate a console session, the window or terminal in which the `smsconnectsc` command was executed goes into raw mode, and appears hung. To escape this condition, type:

```
sc#:sms-user:> CTRL-j
sc#:sms-user:> stty sane
sc#:sms-user:> CTRL-j
```

Group Privileges Required

You must have platform administrator privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Creating a Remote Connection From the Local SC to the Hung Remote SC

In the following example the local SC is shown as `sc1` and the remote SC is shown as `sc0`. Log in to the local SC as a platform administrator.

```
sc1:sms-user:> smsconnectsc
TTY connection is OFF. About to connect to other SC.
Do you want to continue (yes/no)? y
connected
sc0:sms-user:>
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`rlogin(1M)`, `tip(1M)`

NAME

smsinstall - install the SMS environment

SYNOPSIS

smsinstall *directory_name*

smsinstall `-p`

smsinstall `-h`

DESCRIPTION

`smsinstall`(1M) installs SMS packages on the main or spare SMS system controllers (SCs). You use this command when there are no previous versions of SMS on your system. If you use this command and there are previous versions of SMS installed, the command exits and returns an error message.

The smsinstall command automatically hardens the SCs after installation. Refer to the *System Management Services (SMS) 1.6 Installation Guide*for instructions on how to install SMS using this command.

If you are not running `smsinstall` from the Product directory (*/download_directory*`/System_Management_Services_1_6/Tools`), you must use the absolute path.

If any errors occur, `smsinstall` writes error messages to `/var/sadm/system/logs/smsinstall`.

OPTIONS

The following options are supported

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

`-p` Performs post-package add actions only.

OPERANDS

The following operand is supported:

directory_name Name of the directory which contains the SMS packages.

The default path to the Product directory, *directory_name*, is: */download_directory*`/System_Management_Services_1_6/Product` where *download_directory* is the location where you downloaded the files from the Web.

EXTENDED DESCRIPTION

Group Privileges Required

You must have superuser privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Installing SMS From the Web

```
sc#:sms-user:> download_directory/System_Management_Services_1_6/
Tools/smsinstall download_directory/System_Management_Services_1_6/
Product
Checking if Solaris Security Toolkit is already installed.
Installing Solaris Security Toolkit package SUNWjass
Copyright 2005 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.

Installation of <SUNWjass> was successful.
Installing Solaris Security Toolkit package SUNBEfixm

Installation of <SUNBEfixm> was successful.
Installing Solaris Security Toolkit package SUNBEmd5

Installation of <SUNBEmd5> was successful.
Solaris Security Toolkit packages installed successfully.
Installing SMS packages.  Please wait. . .
pkgadd -n -d "../Product" -a /tmp/smsinstall.admin.24308 SUNWscdvr.u
SUNWSMSr SUNWSMSop SUNWSMSdf SUNWSMSjh SUNWSMSlp SUNWSMSmn SUNWSMSob
SUNWSMSod SUNWSMSpd SUNWSMSpo SUNWSMSpp SUNWSMSsu SUNWufrx.u SUNWufu
SUNWwccmn
Copyright 2005 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.

Installation of <SUNWscdvr> was successful.
[...]
Verifying that all SMS packages are installed.OK
Setting up /etc/init.d/sms run control script for SMS 1.6
Setting up /etc/init.d/zoedsms run control script for SMS 1.6
/etc/opt/SUNWSMS/SMS1.5/startup/zoedsms.

Attempting to restart daemon picld
/etc/init.d/picld stop
/etc/init.d/picld start
Running Solaris Security Toolkit 4.2.0 hardening on System Controller.
[NOTE] The following prompt can be disabled by setting JASS_NOVICE_USER
to 0.
[WARN] Depending on how the Solaris Security Toolkit is configured, it is
both possible and likely that by default all remote shell and file transfer
access to this system will be disabled upon reboot effectively locking out
any user without console access to the system.

Are you sure that you want to continue? (YES/NO) [YES]
[NOTE] Executing driver, sunfire_15k_sc-secure.driver
Solaris Security Toolkit hardening step executed successfully on the
System Controller but it will not take effect until the next reboot.
Before rebooting, please make sure SSH or the serial line is setup for use
after the reboot.
smsinstallcomplete.  Log file is /var/sadm/system/logs/smsinstall.
```

EXIT STATUS

The following exit values are returned:

0	Successful completion.
>0	An error occurred.

FILES

The following file is used by this command:

`/var/sadm/system/logs/smsinstall` `smsinstall` log file

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop
Interface Stability	Evolving

SEE ALSO

`smsconfig(1M)`, `smsupgrade(1M)`

System Management Services (SMS) 1.6 Installation Guide

NAME

smsrestore - restore the SMS environment

SYNOPSIS

smsrestore *filename*

smsrestore `-h`

DESCRIPTION

`smsrestore`(1M) restores the operational environment of the SMS from a backup file created by `smsbackup`(1M). Use `smsrestore` to restore the SMS environment after the SMS software has been installed on a new disk.

Turn off failover and stop SMS before running `smsrestore`; start SMS and turn on failover, if you want, afterward. For information about manually starting and stopping SMS refer to the *System Management Services (SMS) 1.6 Installation Guide.*

If any errors occur, `smsrestore` writes error messages to `/var/sadm/system/logs/smsrestore`.

Note – If the main SMS environment has changed since the backup file was created— for example, by shutting down a domain— you must run `smsbackup`(1M) again in order to maintain a current backup file for the system controller.

OPTIONS

The following option is supported.

`-h` Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to `-h` is ignored.

OPERANDS

The following operands are supported:

filename Name of the backup file that was created by `smsbackup`(1M). If the specified file is not in the current directory, the *filename* must contain the full path name for the file. This file can reside anywhere on the system, connected network, or tape device. If no *filename* is specified, you receive an error message.

EXTENDED DESCRIPTION

Group Privileges Required

You must have superuser privileges to run this command.

Refer to Chapter 2, "SMS Security Options and Administrative Privileges" in the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Restoring SMS

```
sc# /opt/SUNWSMS/bin/smsrestore sms_backup.1.0.cpio
```

EXAMPLE 2 Restoring SMS From Tape Device 0

```
sc# /opt/SUNWSMS/bin/smsrestore /dev/rmt/0/sms_backup.1.0.cpio
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following file is used by this command:

`/var/sadm/system/logs/smsrestore`	`smsrestore` log file

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`smsbackup(1M)`

NAME

smsupgrade - upgrades the SMS software to the current version

SYNOPSIS

smsupgrade [-b] [-r] *directory_name*

smsupgrade -p

smsupgrade -h

DESCRIPTION

smsupgrade(1M) upgrades SMS software on the main or spare SMS system controllers. Refer to the *System Management Services (SMS) 1.6 Installation Guide* for instructions about how to upgrade SMS using this command.

The Solaris 10 OS on the SCs is supported only in SMS 1.6. If you are running the Solaris 10 OS on the SCs, you must upgrade your SMS version from 1.4.1 or 1.5 to SMS 1.6. You may not downgrade from SMS 1.6 to a different SMS version unless you first reinstall the Solaris 9 OS on the SCs. This restriction does not apply to Solaris 10 OS on domains.

If you are not running smsupgrade from the Product directory you must use the absolute path. The absolute path is *download_directory*/System_Management_Services_1_6/Product, where *download_directory* is the location where you downloaded the files from the web.

If any errors occur, smsupgrade writes error messages to /var/sadm/system/logs/smsupgrade.

OPTIONS

The following options are supported

-b — Neither smsbackup(1M) nor smsrestore(1M) are performed.

The default is to run smsbackup to directory /var/tmp before upgrading SMS.

-h — Help. Displays usage descriptions.

Note – Use alone. Any option specified in addition to -h is ignored.

-p — Performs post-package add actions only; it does not perform smsbackup or smsrestore.

-r — smsrestore(1M) is *not* performed after the upgrade

The default is to run smsrestore from directory /var/tmp after upgrading SMS.

OPERANDS

The following operands are supported:

directory_name Name of the directory that contains the SMS packages.

If you are installing from the Web, the path to the Product directory, *directory_name*, is */download_directory/*`System_Management_Services_1_6/Product`; where *download_directory* is the location where you downloaded the files from the Web.

EXTENDED DESCRIPTION

Group Privileges Required

You must have superuser privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Upgrading SMS

```
sc0:sms-user:> cd download_directory/System_Management_Services_1_6/Tools/

sc0:sms-user:> ./smsupgrade directory_name

Attempting to stop daemon picld
/etc/init.d/picld stop

Verifying that all SMS packages are installed
..............OK

Backing up SMS to /var/tmp/sms_backup.1.6.cpio before upgrade.
Please wait. . .
smsbackup /var/tmp
smsbackup: Backup configuration file created: /var/tmp
sms_backup.1.6.cpio

SMS backup complete.

Checking if Solaris Security Toolkit is already installed.
Installing Solaris Security Toolkit package SUNWjass
Copyright 2006 Sun Microsystems, Inc.  All rights
reserved.
Use is subject to license terms.
Installation of <SUNWjass> was successful.

Installing Solaris Security Toolkit package SUNBEfixm

Installation of <SUNBEfixm> was successful.
Installing Solaris Security Toolkit package SUNBEmd5

Installation of <SUNBEmd5> was successful.
Solaris Security Toolkit packages installed successfully.

Installing SMS packages.  Please wait. . .
pkgadd -n -d "../Product" -a /tmp/smsinstall.admin.24308 SUNWscdvr.u
```

```
SUNWSMSr SUNWSMSop SUNWSMSdf SUNWSMSjh SUNWSMSlp
SUNWSMSmn SUNWSMSob
SUNWSMSod SUNWSMSpd SUNWSMSpo SUNWSMSpp SUNWSMSsu
SUNWufrx.u SUNWufu
SUNWwccmn
Copyright 2006 Sun Microsystems, Inc.  All rights
reserved.
Use is subject to license terms.
Installation of <SUNWscdvr> was successful.
Installation of <SUNWSMSr> was successful.
Verifying that all SMS packages are installed. OK
Setting up /etc/init.d/sms run control script for SMS 1.6
Setting up /etc/init.d/zoedsms run control script for SMS 1.6
New SMS version 1.6 is active
Restoring SMS from /var/tmp/sms_backup.1.6.cpio after upgrade.
Please wait...
smsrestore /var/tmp/sms_backup.1.6.cpio
smsrestore complete. Log file is /var/sadm/system/logs/smsrestore.
Attempting to start daemon picld
/etc/init.d/picld start
Attempting to start zoed...
zoed started.
It is recommended to harden the System Controller after an
SMS upgrade.  Execute the following to do this:
  1) /opt/SUNWjass/bin/jass-execute -q \
       sunfire_15k_sc-secure.driver
  2) Reboot the System Controller
smsupgrade complete.  Log file is /var/sadm/system/logs
smsupgrade.
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following files are used by this command:

`/var/sadm/system/logs/smsupgrade`	`smsupgrade` log file
`/var/tmp/sms_backup.1.5.cpio`	SMS backup file

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop
Interface Stability	Evolving

SEE ALSO

`smsbackup(1M)`, `smsconfig(1M)`, `smsinstall(1M)`, `smsrestore(1M)`

System Management Services (SMS) 1.6 Installation Guide

NAME

smsversion - change the active version of SMS to another co-resident version of the SMS software

SYNOPSIS

smsversion *new_version*

smsversion `-t`

smsversion `-h`

DESCRIPTION

`smsversion`(1M) can be used to switch between two co-installed (and consecutively released) versions of SMS.

The Solaris 10 OS on the SCs is supported only in SMS 1.6. If you are running the Solaris 10 OS on the SCs, you must upgrade your SMS version to SMS 1.6. If you upgraded both SMS and the operating system, you cannot switch to the previous version of SMS unless you first reinstall the previous version of the operating system.

On the Solaris 9 OS, if you upgrade SMS from SMS 1.5 to SMS 1.6, you can switch back to SMS 1.5. If you upgrade from SMS 1.4 to SMS 1.4.1, SMS 1.5, and then to SMS 1.6, you cannot switch back to SMS 1.4 or SMS 1.4.1, only to SMS 1.5. Refer to the *System Management Services (SMS) 1.6 Installation Guide* for more information.

`smsversion` permits two-way SMS version-switching between sequential co-resident installations on the same operating system, but with the following conditions:

Condition	Explanation
New features	The features supported in the newer version of SMS may not be supported in the older version. Switching to an older version of SMS can result in the loss of those features. Also, the settings for the new features might be erased.
Flash PROM differences	Switching versions of SMS requires reflashing the CPU flash PROMs with the correct files. These files can be found in the `/opt/SUNWSMS/<SMS_version>/firmware` directory. Use `flashupdate`(1M) to reflash the PROMs after you have switched versions. Refer to the `flashupdate` man page, the *System Management Services (SMS) 1.6 Administrator Guide*, and the *System Management Services (SMS) 1.6 Installation Guide* for more information on updating flash PROMs.

`smsversion`, when invoked with no command-line argument, displays the list of all properly installed versions of SMS on the current system controller. You can pick from that list, and `smsversion` stores a copy of the current configuration environment and then switches all necessary software links necessary for activating the new version of the software. `smsversion` can run with an optional command-line argument specifying the target version for switching.

Once `smsversion` completes the switch, the target version becomes the active version. To restore the configuration automatically saved by `smsversion` you must use `smsrestore`(1M). Your previous configuration is not automatically restored as part of the version switch.

To restore your previous configuration, do the following:

- Turn off failover and stop SMS. Then run `smsrestore`.

Note – If you changed your network configuration using `smsconfig -m` *after* you created the backup you just restored, you must run `smsconfig -m` and reboot before proceeding to the next step.

- Start SMS and turn on failover. For information about manually starting and stopping SMS, refer to the *System Management Services (SMS) 1.6 Installation Guide.*

If any errors occur, `smsversion` writes error messages to `/var/sadm/system/logs/smsversion`.

OPTIONS

The following options are supported:

`-h`	Help. Displays usage descriptions. **Note –** Use alone. Any option specified in addition to `-h` is ignored.
`-t`	Displays the current active version of SMS and exits.

OPERANDS

The following operand is supported:

version_number	Release number of the target SMS version

EXTENDED DESCRIPTION

Group Privileges Required

You must have superuser privileges to run this command. This command must be run as root, or SMS will return an error.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Displaying the Version with One Version of SMS Installed

Displays the active version and exits when only one version of SMS is installed.

```
sc# /opt/SUNWSMS/bin/smsversion -t
1.6
```

EXAMPLE 2 Changing the Active Version of SMS

Displays versions of SMS installed on this system controller. Choose the inactive version and perform a version switch.

You must stop SMS before switching versions; otherwise, SMS returns an error.

To switch versions, you perform these steps in order, as described in the *System Management Services (SMS) 1.6 Installation Guide*. Examples of the output for upgrade and downgrade follow.

Disable failover.

Stop SMS on the Main SC.

Run smsversion on the Main SC.

Run smsrestore on the Main SC, using the backup cpio file created by smsversion.

Restart SMS on the Main SC.

Stop SMS on the Spare SC.

Run smsversion on the Spare SC.

Run smsrestore on the Spare SC.

Restart SMS on the Spare SC.

Enable failover.

EXAMPLE 3 Switching To a Higher SMS Version

```
sc# /opt/SUNWSMS/bin/smsversion 1.6
smsversion: Active SMS version < 1.5 >
You have requested SMS Version 1.6

Is this correct? [y,n] y

smsversion: Upgrading SMS from <1.5> to <1.6>.
smsversion: SMS version 1.6 installed
To move to a different version of SMS an archive of
critical files will be created. What is the name of
the directory or tape device where the archive will be
stored? [/var/tmp][return]

smsversion: Backup configuration file created: /var/tmp/
sms_backup.1.6.cpio
smsversion: Switching to target version <1.6>.
smsversion: New Version <1.6> Active
smsversion: Active SMS version < 1.6 >
To use the previous SMS configuration settings type:
smsrestore /var/tmp/sms_backup.1.6.cpio
smsversion complete. Log file is /var/sadm/system/logs/smsversion.
```

EXAMPLE 4 Downgrading SMS Versions

```
sc# /opt/SUNWSMS/bin/smsversion 1.5
smsversion: Active SMS version < 1.6 >
You have requested SMS Version 1.5

Is this correct? [y,n] y

smsversion: Downgrading SMS from <1.6> to <1.5>.
To move to a different version of SMS an archive of
critical files will be created. What is the name of
the directory or tape device where the archive will be
stored?
Enter archive location [/var/tmp]: [return]
smsversion: Backup configuration file created: /var/tmp/
sms_backup.1.6.cpio
smsversion: Switching to target version <1.5>.
smsversion: New Version <1.5> Active
You are downgrading to a version of SMS that requires you
to perform the following operations:
 1) Undo hardening
 2) Redo hardening
 3) Reboot the System Controller

Perform these activities with the following command sequence:
 1) /opt/SUNWjass/bin/jass-execute -u
 2) /opt/SUNWjass/bin/jass-execute -q sunfire_15k_sc-secure.driver
 3) reboot

smsversion: Active SMS version < 1.5 >
To restore the previous SMS configuration setting type: smsrestore
/var/tmp/sms_backup.1.6.cpio
smsversion complete. Log file is /var/sadm/system/logs/smsversion.
```

EXIT STATUS

The following exit values are returned:

`0`	Successful completion.
`>0`	An error occurred.

FILES

The following files are used by this command:

/opt/SUNWSMS/bin/smsversion	`smsversion` command
`/var/sadm/system/logs/smsversion`	`smsversion` log file

ATTRIBUTES

See `attributes(5)` for a description of the following attribute.

Attribute Type	Attribute Value
Availability	SUNWSMSop

SEE ALSO

`smsbackup(1M)`, `smsrestore(1M)`

NAME

ssd - SMS startup daemon

SYNOPSIS

ssd [`-f` *startup_file*]

ssd [`-i` *message*]

DESCRIPTION

`ssd`(1M) starts, stops, and monitors all the key daemons and servers of SMS. When executed with no options `ssd` reads from the `ssd_start` file, which lists the daemons and servers that `ssd` starts and monitors.

Do *not* execute this program manually. `ssd`(1M) is automatically invoked by a Solaris software run-control script and is periodically monitored for restart.

OPTIONS

The following options are supported:

`-f` *startup_file* Uses this file instead of the default `ssd_start` file.

`-i` *message* Places a notice message in the platform log file. Specified and used exclusively by the `sms` startup script.

FILES

The following files are supported:

`/etc/opt/SUNWSMS/startup/ssd_start` Default startup file for `ssd`

`/etc/opt/SUNWSMS/startup/sms` Default startup file for SMS

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

NAME

testemail - Test SMS event-reporting features

SYNOPSIS

testemail `-c` *fault_class[,fault_class...]* `-d` *domain_indicator* `-i` *indicted_resource[,indicted_resource...]*

testemail `-h`

DESCRIPTION

`testemail`(1M) generates test events so you can verify that they were logged in the SMS platform message log file and reported by email as specified in the email control file (`event_email.cf`). For more information about the email control file, see the *System Management Services (SMS) 1.6 Administrator Guide*.

When you invoke `testemail`, you can specify:

- The event (or events) that `testemail` will generate
- The domain in which the event (or events) will be generated
- The components that will be reported faulty by the test event.

When invoking `testemail` using an Ecache resource, make sure that the system board containing the Ecache is powered on. Otherwise, the `testemail` invocation will fail and no email will be generated.

OPTIONS

Use these options:

`-h`

Help. Displays descriptions of `testemail` arguments and options.

Note – Use alone. Other options included with `-h` are ignored.

`-c` *fault_class* [,fault_class ...]

The fault class or comma-separated list of fault classes that testemail uses to generate an event. For example:

-c fault_class,fault_class,fault_class

Examples of valid fault classes are in the file `/etc/opt/SUNWSMS/SMS/config/SF15000.dict` .

`-d` domain-indicator

The domain in which testemail will generate the events. Can be one of the following:

domain_id – Valid *domain_id*s are the characters A–R, and are not case sensitive.

domain_tag – Name assigned to a domain using `addtag`(1M).

`-i` *indicted_resource* [,indicted_resource ...]

Optional. The component, or a comma-separated list of components, that will be reported faulty by the event. For example:

-i indicted_resource,indicted_resource,indicted_resource

A component does not have to be faulty to be included in the test event.

Accepted components are:

- *board* (a *system_board*, *io_board*, *expander_board*, or *centerplane_support_board*)
- *system_board/port*
- *io_board/port*
- *system_board/port/physical bank/dimm*
- *system_board/port/physical bank/dimm/logical_dimm*
- *system_board/port/ecache*
- *io_board/port/iobus*
- *centerplane*
- *centerplane support*
- *bus* (address bus, data bus, or response bus)
- *expander_board*/cdcdimm0
- *expander_board/bus*
- *system_controller*
- *system_controller_peripheral*
- *fan _tray*
- *power_supply*

`-i` *indicted_resource* [,indicted_resource ...]

Optional. The component, or a comma-separated list of components, that will be reported faulty by the event. For example:

-i indicted_resource,indicted_resource,indicted_resource

A component does not have to be faulty to be included in the test event.

Accepted components are:

- *board* (a *system_board*, *io_board*, *expander_board*, or *centerplane_support_board*)
- *system_board/port*
- *io_board/port*
- *system_board/port/physical bank/dimm*
- *system_board/port/physical bank/dimm/logical_dimm*
- *system_board/port/ecache*
- *io_board/port/iobus*
- *centerplane*
- *centerplane support*
- *bus* (address bus, data bus, or response bus)
- *expander_board*/cdcdimm0
- *expander_board/bus*
- *system_controller*
- *system_controller_peripheral*
- *fan _tray*
- *power_supply*

where

Components are:

system_board	SB(0...17) for Sun Fire 15K/E25K systems`SB(0...8)` for Sun Fire 12K/E20K systems
io_board	IO(0...17) for Sun Fire 15K/E25K systems`IO(0...8)` for Sun Fire 12K/E20K systems
expander_board	EX(0...17) for Sun Fire 15K/E25K systems`EX(0...8)` for Sun Fire 12K/20K systems
port or processors for system_boards	P(0...3)
physical bank	B(0 | 1)
dimm	D(0...3)
logical dimm	L(0 | 1)
io_bus	I(0 | 1)
ecache	E(0 | 1)
centerplane	CP(0 | 1)
centerplane support	CS(0 | 1)
bus	ABUS | DBUS | RBUS(0 | 1)
system_controller	SC(0 | 1)
system_controller_peripheral	SCPER(0 | 1)
fan tray	FT(0...7)
power supply	PS(0...5)

EXTENDED DESCRIPTION

Test Results

The results of the email test consist of:

- Event messages in the platform messages log that record the test fault and event code, and a text string to ignore the message.
- Emails that notify the recipients specified in the email control file

Group Privileges Required

You must have platform administrator or platform service privileges to run this command.

Refer to the *System Management Services (SMS) 1.6 Administrator Guide* for more information.

EXAMPLES

EXAMPLE 1 Generating Test Fault Events for Expander and I/O Boards

```
sc0:sms-user:> /opt/SUNWSMS/SMS/lib/smsadmin/testemail -c
fault.board.ex.l1l2,fault.board.io.l1l2 -dD -i EX7,IO8
```

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

FILES

The following files are used by this command:

`/etc/opt/SUNWSMS/SMS/config/event_email.cf`

Specifies the recipients of email notifications for different fault classes

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes:

Attribute Types	Attribute Values
Availability	SUNWSMSop
Interface Stability	Evolving
Command Output	Unstable

SEE ALSO

`erd(1M)`

NAME

tmd - task management daemon

SYNOPSIS

tmd [`-t` *number*]

DESCRIPTION

`tmd`(1M) provides task management services, such as scheduling for SMS. The purpose of this service is to reduce the number of conflicts that can arise during concurrent invocations of the hardware tests and configuration software.

This daemon is started automatically by `ssd`(1M). Do *not* start it manually from the command line.

OPTIONS

The following option is supported:

`-t` *number* This option allows the number of concurrent invocations to be throttled. The value must be a positive number, greater than or equal to one.

Caution– Changing the default value can adversely affect system functionality. Do *not* adjust this parameter unless instructed by a Sun service representative to do so.

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`ssd(1M)`

NAME

wcapp - wPCI application daemon

SYNOPSIS

wcapp

DESCRIPTION

wcapp(1M) is responsible for implementing Sun Fire Link clustering functionality—specifically, handling requests from the domain-side drivers and responding to requests for information from the external Sun Fire Link fabric manager server. wcapp runs in a Java Virtual Machine (JVM) included with Solaris 8 02/02 operating environment or later.

`wcapp` is responsible for managing Sun Fire Link clustering for all the domains in the Sun Fire high-end system. The Java side of `wcapp` exports a set of Java Remote Method Invocation (RMI) interfaces that can be used by the Sun Fire Link fabric manager to set up and monitor a cluster.

This daemon is automatically started by `ssd`(1M). Do *not* start it manually from the command line.

EXIT STATUS

The following exit values are returned:

`0` Successful completion.

`>0` An error occurred.

ATTRIBUTES

See `attributes(5)` for descriptions of the following attributes.

Attribute Types	Attribute Values
Availability	SUNWSMSop

SEE ALSO

`ssd`**(`1M`)**