## Contents

<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intro(1M)</td>
<td>7</td>
</tr>
<tr>
<td>addboard(1M)</td>
<td>11</td>
</tr>
<tr>
<td>addtag(1M)</td>
<td>18</td>
</tr>
<tr>
<td>initcmdsync(1M)</td>
<td>22</td>
</tr>
<tr>
<td>console(1M)</td>
<td>26</td>
</tr>
<tr>
<td>dca(1M)</td>
<td>30</td>
</tr>
<tr>
<td>deleteboard(1M)</td>
<td>32</td>
</tr>
<tr>
<td>deletetag(1M)</td>
<td>37</td>
</tr>
<tr>
<td>disablecomponent(1M)</td>
<td>39</td>
</tr>
<tr>
<td>dsmd(1M)</td>
<td>45</td>
</tr>
<tr>
<td>dxs(1M)</td>
<td>47</td>
</tr>
<tr>
<td>enablecomponent(1M)</td>
<td>49</td>
</tr>
<tr>
<td>esmd(1M)</td>
<td>56</td>
</tr>
<tr>
<td>flashupdate(1M)</td>
<td>59</td>
</tr>
<tr>
<td>fomd(1M)</td>
<td>65</td>
</tr>
<tr>
<td>frad(1M)</td>
<td>66</td>
</tr>
<tr>
<td>help(1M)</td>
<td>67</td>
</tr>
<tr>
<td>hpost(1M)</td>
<td>69</td>
</tr>
<tr>
<td>hwad(1M)</td>
<td>70</td>
</tr>
</tbody>
</table>
showdevices(1M) 180
showenvironment(1M) 185
showfailover(1M) 194
showkeyswitch(1M) 198
showlogs(1M) 200
showobpparams(1M) 204
showplatform(1M) 206
showxirstate(1M) 213
smsbackup(1M) 217
smsconfig(1M) 220
smsconnectsc(1M) 232
smsrestore(1M) 235
smsversion(1M) 237
ssd(1M) 241
tmd(1M) 242
Maintenance Commands
NAME
Intro – SMS administration

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

LIST OF COMMANDS
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>addboard(1M)</td>
<td>Assign, connect, and configure a board to a domain</td>
</tr>
<tr>
<td>addtag(1M)</td>
<td>Assign a domain name (tag) to a domain</td>
</tr>
<tr>
<td>audithotspares(1M)</td>
<td>Check the current number of unassigned system boards in the system</td>
</tr>
<tr>
<td>cancelcmdsync(1M)</td>
<td>Remove a command synchronization descriptor from the command synchronization list</td>
</tr>
<tr>
<td>console(1M)</td>
<td>Access the domain console</td>
</tr>
<tr>
<td>dca(1M)</td>
<td>Domain configuration agent</td>
</tr>
<tr>
<td>deleteboard(1M)</td>
<td>Unconfigure, disconnect and unassign a board from a domain</td>
</tr>
<tr>
<td>deletetag(1M)</td>
<td>Remove the domain name (tag) associated with the domain</td>
</tr>
<tr>
<td>disablecomponent(1M)</td>
<td>Add the specified component to the platform or domain blacklist</td>
</tr>
<tr>
<td>dsmd(1M)</td>
<td>Domain status monitoring daemon</td>
</tr>
<tr>
<td>dxs(1M)</td>
<td>Domain X server</td>
</tr>
<tr>
<td>enablecomponent(1M)</td>
<td>Remove the specified component from the platform, domain or ASR blacklist</td>
</tr>
<tr>
<td>esmd(1M)</td>
<td>Environmental status monitoring daemon</td>
</tr>
<tr>
<td>flashupdate(1M)</td>
<td>Update system board FROMs</td>
</tr>
<tr>
<td>fomd(1M)</td>
<td>Failover management daemon</td>
</tr>
<tr>
<td>frad(1M)</td>
<td>FRU access daemon</td>
</tr>
<tr>
<td>help(1M)</td>
<td>Display help information for SMS commands</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>hpost(1M)</td>
<td>Sun Fire 15K power-on self-test (POST) control application</td>
</tr>
<tr>
<td>hwad(1M)</td>
<td>Hardware access daemon</td>
</tr>
<tr>
<td>initcmdsync(1M)</td>
<td>Create a command synchronization descriptor that identifies the script to be recovered</td>
</tr>
<tr>
<td>kmd(1M)</td>
<td>Key management daemon</td>
</tr>
<tr>
<td>mand(1M)</td>
<td>Management network daemon</td>
</tr>
<tr>
<td>mld(1M)</td>
<td>Message logging daemon</td>
</tr>
<tr>
<td>moveboard(1M)</td>
<td>Move a board from one domain to another</td>
</tr>
<tr>
<td>osd(1M)</td>
<td>OpenBoot PROM server daemon</td>
</tr>
<tr>
<td>pcd(1M)</td>
<td>Platform configuration database daemon</td>
</tr>
<tr>
<td>poweroff(1M)</td>
<td>Control power off</td>
</tr>
<tr>
<td>poweron(1M)</td>
<td>Control power on</td>
</tr>
<tr>
<td>rcfadm(1M)</td>
<td>Remote configuration administration</td>
</tr>
<tr>
<td>reset(1M)</td>
<td>Send reset to all CPU ports of a specified domain</td>
</tr>
<tr>
<td>resetsc(1M)</td>
<td>Reset the spare system controller (SC)</td>
</tr>
<tr>
<td>runcmdsync(1M)</td>
<td>Prepare a specified script for recovery after a failover</td>
</tr>
<tr>
<td>savecmdsync(1M)</td>
<td>Add a marker that identifies a location in the script from which processing can be resumed after a failover</td>
</tr>
<tr>
<td>setbus(1M)</td>
<td>Perform dynamic bus reconfiguration on active expanders in a domain</td>
</tr>
<tr>
<td>setdatasync(1M)</td>
<td>Modify the data propagation list used in data synchronization</td>
</tr>
<tr>
<td>setdate(1M)</td>
<td>Set the date and time for the system controller or a domain</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>setdefaults(1M)</td>
<td>Remove all instances of a previously active domain</td>
</tr>
<tr>
<td>setfailover(1M)</td>
<td>Modify the state of the system controller (SC) failover mechanism</td>
</tr>
<tr>
<td>setkeyswitch(1M)</td>
<td>Change the position of the virtual keyswitch</td>
</tr>
<tr>
<td>setobpparams(1M)</td>
<td>Set up OpenBoot PROM variables for a domain</td>
</tr>
<tr>
<td>setupplatform(1M)</td>
<td>Set up the available component list for domains</td>
</tr>
<tr>
<td>showboards(1M)</td>
<td>Show the assignment information and status of the boards.</td>
</tr>
<tr>
<td>showbus(1M)</td>
<td>Display the bus configuration of expanders in active domains</td>
</tr>
<tr>
<td>showcmdsync(1M)</td>
<td>Display the current command synchronization list</td>
</tr>
<tr>
<td>showcomponent(1M)</td>
<td>Display blacklist status for a component</td>
</tr>
<tr>
<td>showdatasync(1M)</td>
<td>Display the status of system controller (SC) data synchronization for failover</td>
</tr>
<tr>
<td>showdate(1M)</td>
<td>Display the date and time for the system controller (SC) or a domain</td>
</tr>
<tr>
<td>showdevices(1M)</td>
<td>Display system board devices and resource usage information</td>
</tr>
<tr>
<td>showenvironment(1M)</td>
<td>Display the environmental data</td>
</tr>
<tr>
<td>showfailover(1M)</td>
<td>Manage or display system controller (SC) failover status</td>
</tr>
<tr>
<td>showkeyswitch(1M)</td>
<td>Display the position of the virtual keyswitch</td>
</tr>
<tr>
<td>showlogs(1M)</td>
<td>Display message log files</td>
</tr>
<tr>
<td>showobpparams(1M)</td>
<td>Display OpenBoot PROM bringup parameters for a domain</td>
</tr>
<tr>
<td>showplatform(1M)</td>
<td>Display the board available component list and domain state for each of the domains</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>showxirstate(1M)</td>
<td>Display CPU dump information after sending a reset pulse to the processors</td>
</tr>
<tr>
<td>smsbackup(1M)</td>
<td>Back up the SMS environment</td>
</tr>
<tr>
<td>smsconfig(1M)</td>
<td>System controller (SC) configuration utility for the Sun Fire 15K system</td>
</tr>
<tr>
<td>smsconnectsc(1M)</td>
<td>Access a remote SC console</td>
</tr>
<tr>
<td>smsrestore(1M)</td>
<td>Restore the SMS environment</td>
</tr>
<tr>
<td>smsversion(1M)</td>
<td>Change the active version of SMS to another co-resident version of the SMS software</td>
</tr>
<tr>
<td>ssd(1M)</td>
<td>SMS startup daemon</td>
</tr>
<tr>
<td>tmd(1M)</td>
<td>Task management daemon</td>
</tr>
</tbody>
</table>
addboard – assign, connect and configure a board to a domain

**SYNOPSIS**

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

**DESCRIPTION**

`addboard(1M)` assigns, connects and configures a location to the domain `domain_id | 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, or 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 will power on the board and test it.

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, then goes on to the next board or after the last board, exits.

**OPTIONS**

The following options are supported:

- `-c function`

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

**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 will need to reboot the domain in order to use that board.

The possible transition states and their meaning 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 either by using setkeyswitch on or using the connect or configure options.

- **connect**:  
Assigns the board to the logical domain (if it is not already).  
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 environment. 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).  
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 environment. The hardware resources on the board can be used by Solaris software.

---

**-d domain_id**

ID for a domain. Valid domain_ids are ‘A’...'R' and are case insensitive.

**-d 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 −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 −t timeout

These command arguments allow the user to specify retries in case of failures encountered during state transitions. The −r retry_count option indicates the number of times the configuration state change request should be retried by the domain. The −t timeout option specifies the number of seconds that the domain should wait before the next retry is made. This option must be specified with retry_count. The default is zero, meaning the request is retried immediately.

−y

Automatically answers “yes” to all prompts. Prompts are displayed unless used with −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:

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 only perform 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) 1.2 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
SB at SB0 assigned to domain: C
IO at IO1 assigned to domain: C
SB at SB1 assigned to domain: C
SB at 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
SB at SB0 assigned to domain: C
IO at IO2 assigned to domain: C
SB at SB1 assigned to domain: C
Warning: IO at IO2 is blacklisted. You will not be able to connect or configure it.
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
SB at 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 IO at 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 APID doesn’t 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
56 DR command syntax error
68 DR operation failed

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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. To remove a component from the ASR blacklist file, use enablecomponent(1M).

SEE ALSO

addtag(1M), enablecomponent(1M), esmd(1M), showcomponent(1M)
NAME
ddtag – assign a domain name (tag) to a domain

SYNOPSIS
addtag –d domain_id | domain_tag [-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
- -d domain_id ID of a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.
- -d domain_tag Name assigned to a domain using addtag(1M).
new_tag New tag name assigned to a domain. See Extended Description for a description of invalid domain names.
- -h Help. Displays usage descriptions.
- -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

<table>
<thead>
<tr>
<th>Domain Name (Tag) Restrictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>The following restrictions are required on a domain name tag:</td>
</tr>
<tr>
<td>- No single character names</td>
</tr>
<tr>
<td>- All domain name tags must be unique across all domains within a single chassis.</td>
</tr>
<tr>
<td>- Tags must adhere to the same restrictions as defined for Solaris software nodenames. Currently, the size restriction is set to 2 to 64 characters.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Group Privileges Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>You must have platform administrator privileges to run this command.</td>
</tr>
<tr>
<td>Refer to Chapter 2 in the <em>System Management Services (SMS) 1.2 Administrator Guide</em> for more information.</td>
</tr>
</tbody>
</table>

## 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 will be 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 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

```
s0:sms-user:> addtag -d A -n eng2
```

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

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

```
s0:sms-user:> addtag -d A -qy eng2
```

You are not prompted.

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

```
s0: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

```
s0: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 will 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:
<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

delegetag(1M)
NAME
initcmdsync, cmdsync, cancelcmdsync, savecmdsync – command synchronization commands

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 will be 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 is used only to mark specific points in a script from which processing can be resumed. If specific restart points are not needed, 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.
Maintenance Commands

Note - Use alone. Any option specified in addition to \texttt{-h} is ignored.

\texttt{-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.

\texttt{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.

\texttt{script\_name} \\
Identifies the name of the user-defined script to be synchronized.

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:

```bash
# 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 \texttt{-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 initcmds can command.

initcmds can *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) 1.2
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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>All</td>
</tr>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
<tr>
<td>Stability</td>
<td>Evolving</td>
</tr>
<tr>
<td>MT-Level</td>
<td>Safe</td>
</tr>
</tbody>
</table>

SEE ALSO
runcmdsync(1M), showcmdsync(1M)

NOTES
An example of a user-defined script (with synchronization commands) is
provided in the /opt/SUNWSMS/examples/cmdsync directory.
console(1M) Maintenance Commands

NAME

console – access the domain console

SYNOPSIS

console -d domain_id | domain_tag [[-f][[-l][[-g][[-r]]][-e escapeChar]]

console -h

DESCRIPTION

console(1M) creates a remote connection to the domain’s 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 only be taken away 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_id ID for a domain. Valid domain_ids are ‘A’...'R' and are case

  insensitive.

- d 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:
  # @ ^ & ? * = . |

  See the note on rlogin in the Usage section below.

- 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
- \(\sim\) Toggle the communication path between the network and IOSRAM interfaces. You can use \(\sim\) only in Private mode (see \(~*\)).
- `~&` Acquire Locked Write permission; see `−l`. You may 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 seen 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 console command was executed goes into raw mode, and appears hung. To escape this condition, type `^j`, then `stty sane`, then `^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) 1.2 Administrator Guide* for more information.

**EXAMPLES**

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

```
sc0sms-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:

```
sc0sms-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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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_id|domain_tag [-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_id   ID for a domain. Valid domain_ids are ‘A’...’R’ and are case
                insensitive.

- -d 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.

- -H hostname   The Solaris software hostname 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

SMS 1.2         Last modified 01 October 2001
ATTRIBUTES

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

addboard(1M), deleteboard(1M), moveboard(1M), rcfadm(1M)
NAME
deleateboard – unconfigure, disconnect and unassign a system board from a
domain

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

Configuration states are: unconfigure, disconnect, or 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 is
not allowed to unassign a board from a domain unless the board is in the
domain available component list. See setupplatform(1M). This means the
deleateboard location field must appear in the domain available component
list list.

OPTIONS
- 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 then disconnects the board before unassigning it.

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
will need to reboot the domain in order to use that board.

The possible transition states and their meaning are as follows:

- unconfigure:

Unconfigures the board from the Solaris operating environment 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
(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 to the Solaris operating environment. Operations allowed on the board are limited to configuration administration operations.

- **disconnect:**
  
  Unconfigures the board from the Solaris operating environment running on the domain. See `unconfigure` above.
  
  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. This is an intermediate state and does not have any standalone implementation at this time.

- **unassign:**
  
  Unconfigures the board from the Solaris operating environment running on the domain. See `unconfigure` above.
  
  Disconnects the board. See `disconnect` above.
  
  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 `−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.

- retry_count -t timeout

These command arguments allow the user to specify retries in case of failures encountered during state transitions. The -r retry_count option indicates the number of times the configuration state change request should be retried by the domain. The -t timeout option specifies the number of seconds that the domain should wait before the next retry is made. This option must be specified with retry_count. The default is zero, meaning the request is retried immediately.

-y

Automatically answers “yes” to all prompts. Prompts are displayed unless used with -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:

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

Note - Use showboards(1M) to display board type.
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) 1.2 Administrator Guide for more information.

**EXAMPLES**

**EXAMPLE 1**  Unassigning a Board from a Domain

To unassign four boards from their domain, the user must have platform administrator privileges or domain administrator/configurator privileges and the boards must be in the domain available component list.

All boards are in the assigned state in the example domain.

```
sc0:sm-user> deleteboard -c unassign SB0 IO1 SB1 SB2
```

**EXAMPLE 2**  Unconfiguring a Board from a Domain

To unconfigure three boards from their domain, setting retries to five and timeout to three seconds.

```
sc0:sm-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      APID doesn’t 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
56      DR command syntax error
68      DR operation failed

**ATTRIBUTES**

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**SEE ALSO**

addboard(1M), moveboard(1M)
NAME
deletag – remove the domain tag name associated with the domain

SYNOPSIS
deletag -d domain_id | domain_tag [-q] [-y] [-n]
deletag -h

DESCRIPTION
deletag(1M) removes the domain tag associated with the domain.

OPTIONS
- d domain_id ID of a domain. Valid domain_ids are ‘A’...'R' and are case insensitive.
- d domain_tag Name assigned to a domain using addtag(1M).
- h Help. Displays usage descriptions.

<table>
<thead>
<tr>
<th>Note</th>
<th>Use alone. Any option specified in addition to -h is ignored.</th>
</tr>
</thead>
</table>

- 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 in the System Management Services (SMS) 1.2 Administrator Guide for more information.
EXAMPLES

EXAMPLE 1  Deleting Tag “eng2” From the Domain to Which it was Assigned

```
sc0sms-user: > deletetag -d eng2 -qy
```

You will not be 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

addtag(1M)
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 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 and one for platform boards; and the internal automatic system recovery (ASR) blacklist.

disablecomponent used without any option edits the platform blacklist file.

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

For more information on the use and editing of platform and domain blacklists refer to Chapter 7 in the System Management Services (SMS) 1.2 Administrator Guide.

**OPTIONS**

- **−d domain_id**  ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive. Specifies the component to add to the domain blacklist.

- **−d domain_tag**  Name assigned to a domain using addtag(1M). Specifies the component to add to the domain blacklist.

- **−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 comprised of:

board_loc/proc/bank/logical_bank
board_loc/proc/bank/all_dimms_on_that_bank
board_loc/proc/bank/all_banks_on_that_proc
board_loc/proc/bank/all_banks_on_that_board
board_loc/proc

board_loc/procs

board_loc/cassette

board_loc/bus

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:

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 will cause disablecomponent to exit and display an error message.

The following proc forms are accepted:
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 hsPCI assemblies contain hot-swappable cassettes.
The following hsPCI forms are accepted:

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

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, domain configurator privileges to run this command. If you have platform privileges you may run this command for the platform components only. If you have domain privileges you may only run this command on the domain for which you have privileges.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

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

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

EXAMPLE 2 Add 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**  Add 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**  Add All Banks on Processor 0 on System Board 1 to the Domain B Blacklist

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

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

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

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

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

**EXAMPLE 7**  Add 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**  Add the Data Bus CS0 on EX9 to the Domain A Blacklist

```
sc0:sms-user> disablecomponent -dA EX9/DBUS0
```
EXAMPLE 9  Add 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 10  Add Processor Pair 1 on System Board 3 to the Platform Blacklist Because It Needs Service

  sc0:sms-user:> disablecomponent -i "Because it needs service" SB3/PP1

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

FILES  The following file is 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.

SEE ALSO  addboard(1M), enablecomponent(1M), esmd(1M), showcomponent(1M)
NAME
dsmd – domain status monitoring daemon

SYNOPSIS
dsmd

described(1M) monitors domain status and operating system (OS) heartbeat for up
to 18 domains.

dsmd automatically recovers the domain and handles domain-related hardware
errors. In the event of a domain hang, dsmd will reset(1M) the domain, collect
CPU registers and hardware configuration dumps, and save them to two files.

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 sscd(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:

- dxs(1M) — domain X server daemon
- efe — Sun Management Center 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
- setkeyswhitch(1M) — virtual keyswitch control command

For more information see the System Management Services (SMS) 1.2
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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO
dxs(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_id | domain_tag

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 allows one or more users running the console program to access the domain’s virtual console.

When the domain is up and 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 shutdown.

OPTIONS
- d domain_id ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.
- d domain_tag Name assigned to a domain using addtag(1M).
- S Disable console output logging. By default, logging is enabled and 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO
addtag(1M), console(1M), ssd(1M)
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 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 and 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 used without any option will edit 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 Chapter 7 in the System Management Services (SMS) 1.2 Administrator Guide.

OPTIONS

−a
Specifies the component to remove from the ASR blacklist.

−d domain_id
ID for a domain. Valid domain_ids are ’A’...’R’ and case insensitive. Specifies the component to remove from the domain blacklist.

−d domain_tag
Name assigned to a domain using addtag(1M). Specifies the component to remove from the domain blacklist.

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/bank/all_banks_on_that_proc`
- `board_loc/proc/bank/all_banks_on_that_board`
- `board_loc/proc`
- `board_loc/procs`
- `board_loc/cassette`
- `board_loc/bus`

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 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:

- `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 will cause 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 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 hsPCI assemblies contain hot-swappable cassettes. The following hsPCI forms are accepted:

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

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, domain configurator privileges to run this command. If you have platform privileges you may run this command for the platform components only. If you have domain privileges you may only run this command on the domain for which you have privileges.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  Remove CSB0 from the ASR Blacklist

sc0:smi-user> enablecomponent -a CS0

EXAMPLE 2  Remove the Logical Bank 0 of Bank 0 on Processor 0 on System Board 0 from the Domain A Blacklist

sc0:smi-user> enablecomponent -dA SB0/P0/B0/L0
EXAMPLE 3   Remove All DIMMs on Bank 0 of Processor 1 on System Board 3 from the Domain A Blacklist

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

EXAMPLE 4   Remove All Banks on Processor 0 on System Board 1 from the Domain B Blacklist

sc0:sm-user> enablecomponent -dB SB1/P0/B

EXAMPLE 5   Remove All Banks on System Board 0 from the Domain D Blacklist

sc0:sm-user> enablecomponent -dD SB0/B

EXAMPLE 6   Remove Processor Pair 0 on I/O Board 7 from the Platform Blacklist

sc0:sm-user> enablecomponent I07/PP0
EXAMPLE 7  Remove Processor 1 on System Board 3 from the Domain A Blacklist

```bash
sc0:smi-user> enablecomponent -dA SB3/P1
```

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

```bash
sc0:smi-user> enablecomponent -dA IO6/C3V0
```

EXAMPLE 9  Remove the Address Bus CS0 on EX7 from the Domain A Blacklist

```bash
sc0:smi-user> enablecomponent -dA EX7/ABUS0
```

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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.

/etc/opt/SUNWSMS/config/platform/blacklist
List of platform components excluded.

/etc/opt/SUNWSMS/config/domain_id/blacklist

List of domain components excluded.

SEE ALSO
addboard(1M), disablecomponent(1M), esmd(1M), showcomponent(1M)
esmd(1M) Maintenance Commands

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
- 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 only sends notification 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 only sends notification if that client (hwad, pcd, dsmd 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.

56 SMS 1.2 Last modified 01 October 2001
- 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 will remove the board from the domain and power 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 see the System Management Services (SMS) 1.2 Administrator Guide.

EXIT STATUS
The following exit values are returned:
0 Successful completion.
>0 An error occurred.

FILES
The following files are supported:

Last modified 01 October 2001
SMS 1.2 57
ATTRIBUTES

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO
dsm(1M), hwad(1M), pcd(1M), ssd(1M)
NAME  
flashupdate – update the Flash PROMs located on the CPU boards, MaxCPU boards and system controllers (SC)

SYNOPSIS  
flashupdate [-d domain_id | domain_tag] [-f path] [-q] [-y | -n]
flashupdate [-f path] [-y | -n] location [location...]
flashupdate -h

DESCRIPTION  
flashupdate(1M) updates the Flash PROM in the system controller (SC), and the Flash PROMs in a domain’s CPU and MaxCPU boards, given the board location.

In order to update the Flash PROMs in the system controller, log in to the SC you wish to update and specify the Flash PROM to be updated. Each Flash PROM has a specific image file associated with it. Once you have finished updating the SC Flash PROMs you must shutdown and reset the SC. See example 7 below. You do not need to reset the SC after updating CPU Flash PROMs.

To update the CPU FPROMs, SMS must be running and the specified board must be powered on. This is not required to update the SC FPROMs. If any of the domain’s CPU or MaxCPU boards have the virtual keyswitch setting in the secure position, the Flash PROM(s) will not be updated.

flashupdate displays both the current Flash PROM and the flash image file information prior to any updates.

OPTIONS  
−d domain_id  ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.
−d domain_tag  Name assigned to a domain using add.tag(1M).
−f path  Name of the flash image file.
The path argument specifies the name of the image file used to update the Flash PROM 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.

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

OPERANDS

The following operands are supported:

location Flash PROM location.

The Flash PROM location consists of the board_loc/FlashPROM_id separated by a forward slash.

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

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

The following board_loc forms are accepted:

SB(0...17)
IO(0...17)
SC(0-1)

The following FlashPROM_id forms are accepted:

FP(0|1)

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 Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  Updating Flash PROM 0 in the System Controller 0
You must reset the SC after running this command.

```
sc0:sm-user> flashupdate -f /opt/SUNWSMS/firmware/SCOBPimg.d1 SC0/FP0
```

EXAMPLE 2  Updating Flash PROM 1 in the System Controller 0
You must reset the SC after running this command.

```
sc0:sm-user> flashupdate -f /opt/SUNWSMS/firmware/SSCPOST.d1 SC0/FP1
```

EXAMPLE 3  Updating Flash PROM 0 in the System Controller 1
You must reset the SC after running this command.

```
sc1:sm-user> flashupdate -f /opt/SUNWSMS/firmware/SCOBPimg.d1 SC1/FP0
```

EXAMPLE 4  Updating Flash PROM 0 in the slot0 Board
SMS must be running and the SB0/FP0 board must be powered on.
EXAMPLE 5  Updating Both Flash PROMs on CPU Board 17
SMS must be running and the SB17 board must be powered on.

```
sc0:sms-user: > flashupdate -f /opt/SUNWSMS/hostobjs/sgcpu.flash SB0/FP0
```

EXAMPLE 6  Updating Flash PROMs 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 7  Resetting the SC After Updating the SC PROMs.
Switch to superuser and shutdown 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 15K cabinet and depress the Abort and then Reset buttons 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 flashupdate worked, type:

(continued)
ok show-dropins

Dropins for Flash device: /pci@1f,0/pci@1,1/ebus@1/flashprom@10,400000

<table>
<thead>
<tr>
<th>Dropin name</th>
<th>Size</th>
<th>Checksum</th>
<th>Date created</th>
<th>Date flashed</th>
<th>Version</th>
<th>Vendor</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSCOBP-dropins</td>
<td>90</td>
<td>c84e</td>
<td>11/13/2001</td>
<td>11/13/2001</td>
<td>1.2</td>
<td>SUNW,sscobp</td>
</tr>
</tbody>
</table>

Note the version number of the Dropins (1.2)

You can then boot your new installation of the Solaris software.

ok boot new disk

Login in as a platform administrator and type:

sc1:~> 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

sc1:~>

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.

**ATTRIBUTES**

See attributes(5) for descriptions of the following attributes:
<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**FILES**

The following files are used by this command:

- `/opt/SUNWSMS/firmware/SCOBPimg.di` Used to update the Flash PROM 0 on the SC.
- `/opt/SUNWSMS/firmware/SSCPOST.di` Used to update the Flash PROM 1 on the SC.
- `/opt/SUNWSMS/hostobjs/sgcpu.flash` Used to update the Flash PROMs on the CPU and MaxCPU boards.

**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 master and the standby SCs.

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

FILES
The following configuration file is required:
/etc/opt/SUNWSMS/config/fomd.cf  Failover daemon configuration file

Note - This is an internal SMS system file and should not be modified except by authorized Sun Microsystems personnel.

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO
setfailover(1M), showfailover(1M)

Last modified 01 October 2001  SMS 1.2  65
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 15K system. frad also provides the platform-dependent interface to the Sun Fire 15K 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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

−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:

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 Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1 Using Help
Displays all commands.

c0:~> help
addtag −d domain_id|domain_tag −anew_tag [−q ] [−y | −n]
addboard −d domain_id|domain_tag [−c function] [−t retry_count [−t timeout]]
[−q ] [−y | −n]location [location] ...

... tmd [−t number]
EXAMPLE 2 Using Help for a Command

Displays man(1M) page.

sc0:sm-user:> help addtag
 Maintenance Commands addtag(1M)

NAME

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

SYNOPSIS

  addtag -d domain_id|domain_tag -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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

man(1M)
**NAME**
hpost – Sun Fire 15K 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 15K domain, preparing it for use by the OpenBoot PROM and the Solaris operating environment. 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 non fatal hardware error states, and perform related Sun Fire 15K 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSlp</td>
</tr>
</tbody>
</table>

**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 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 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO dsmd(1M), ssd(1M)
NAME
initcmdsync, cmdsync, cancelsync, savecmdsync – command synchronization commands

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 will be 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 is used only to mark specific points in a script from which processing can be resumed. If specific restart points are not needed, consider using `runcmdsync` 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.

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:

```bash
# 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
}
#
# Place this script and all its parameters in the command synchronization list, which indicates the commands to be restarted after an SC failover.
#
for arg in $*; do
    case $arg in
    "-M")
        goto_label=$arg
        ;;
    .
    .
    esac
done
```

# 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 initcmsync command.

initcmsync  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) 1.2*
*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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>All</td>
</tr>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
<tr>
<td>Stability</td>
<td>Evolving</td>
</tr>
<tr>
<td>MT-Level</td>
<td>Safe</td>
</tr>
</tbody>
</table>

**SEE ALSO**

runcmdsync(1M), showcmdsync(1M)

**NOTES**

An example of a user-defined script (with synchronization commands) is
provided in the `/opt/SUNWSMS/examples/cmdsync` directory.
NAME
kmd – SMS key management daemon

SYNOPSIS
kmd

DESCRIPTION
kmd(1M) manages the IPSec security associations (SAs) needed to secure 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.

OPTIONS

EXTENDED
DESCRIPTION
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.

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSr,</td>
</tr>
<tr>
<td></td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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. The fields are identified below.

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

The fields are defined as:

- `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, space A space for the domain ID defines a policy which applies to all domains. A policy for a specific domain overrides a policy which applied to all domains.
- `login`— Login name. Values: Any valid login name. The default policies in the kmd_policy.cf file are shown below.

```
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 }
```
SEE ALSO

ssd(1M), sckmd(1M), ipseconf(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 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:

<table>
<thead>
<tr>
<th>Return Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful completion.</td>
</tr>
<tr>
<td>&gt;0</td>
<td>An error occurred.</td>
</tr>
</tbody>
</table>
SIGNALS

SIGHUP
Rereads the MAN.cf file and reconfigures the external community as well as the network fields in the pcd.

FILES

The following configuration file is required:
/etc/opt/SUNWMS/config/MAN.cf
This file includes the domain-to-SC, 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

fomd(1M), pcd(1M), smsconfig(1M), ssd(1M)
mld(1M) provides logging services to all SMS daemons and processes. mld is the first SMS daemon started by ssc(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 only appears in verbose mode.
  For example:

- 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:
  [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.
0 Successful completion.
>0 An error occurred.

ATTRIBUTES

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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 output 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 case insensitive.

SEE ALSO

ssd(1M)
NAME
moveboard – move a board from one domain to another

SYNOPSIS
moveboard 
−d domain_id | domain_tag [−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, then proceeds to assign, connect, and configure location to the
domain domain_id | 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, or 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
will power on the board and test it.

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

OPTIONS

−c function
Valid function values are assign, connect, and configure. This value is used to control the configuration state transition.

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

The possible transition states and their meaning are as follows:

- **assign:**
  - Unconfigures the board from the Solaris operating environment 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 either by using setkeys on or using the connect or configure options.

- **connect:**
  - Unconfigures the board from the Solaris operating environment 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 above; 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 environment. 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 environment 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 above; 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 environment. Allowed operations on the board are limited to configuration administration operations. (See connect above; 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 environment. The hardware resources on the board can be used by Solaris software. (See also addboard -c configure).

-d domain_id

ID for a domain. Valid domain_ids are ‘A’...'R' and are case insensitive. This is the domain to which the board is being moved.

-d 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 -t timeout`

These command arguments allow the user to specify retries in case of failures encountered during state transitions. The `-r retry_count` option indicates the number of times the configuration state change request should be retried by the domain. The `-t timeout` option specifies the number of seconds that the domain should wait before the next retry is made. This option must be specified with `retry_count`. The default is zero, meaning 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:

- 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 Chapter 2 in the *System Management Services (SMS) 1.2 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
SB at 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
SB at SB4 assigned to domain: A
Warning: CPU at 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 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 IO17 to Domain R

```
sc0:SMS-USER> moveboard -d R -c connect IO17
```

**EXAMPLE 5**  Connecting a Blacklisted Board to Domain C

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

```
sc0:SMS-USER> moveboard -d C -c connect SB0
SB at SB0 is blacklisted. Exiting.
```
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 APID doesn’t exist
14 Invalid attribute
15 Invalid board ID type
16 Invalid permissions
17 Assigned to another domain
18 Unable to get permissions
19 Unable to get domain board info
20 Unable to get active board list
21 Unable to get assigned board list
Maintenance Commands

ATTRIBUTES

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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. To remove a component from the ASR blacklist file, use enablecomponent(1M).

SEE ALSO

addtag(1M), addboard(1M), deleteboard(1M), enablecomponent(1M), esmd(1M), showcomponent(1M)
osd(1M) Maintenance Commands

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, NVRAM 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. There is one instance of osd on the system controller (SC) 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO
setkeyswitch(1M)

90 SMS 1.2 Last modified 01 October 2001
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 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 includes:

Platform type
Platform name,
Rack ID
Cacheable Memory Address Slice Map
System clock frequency
System clock type
SC IP address
SC0 to SC1 IP address
SC1 to SC0 IP address
SC to SC IP netmask

Domain data includes:

Domain ID/Tag
OS version (not used)
OS type (not used)
Available component list
Assigned board list
Active board list
Golden IOSRAM I/O board
Virtual keyswitch setting
Active Ethernet I/O board
Domain creation time
Domain dump state
Domain bring up priority
IP host address
Host name
Host netmask
Host broadcast address

Last modified 01 October 2001
System board data includes:

- Expander Position
- Slot position
- Board type
- Board state
- Domain assignment of the board
- Available component list state
- Board test status
- Board test level
- Memory cleared state

**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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**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 there are active domains utilizing the component that is going to be powered off, a listing of those domains and a “Are you sure?” prompt will be given by default.

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:
- Help. Displays usage descriptions.
  -h

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

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

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

        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.

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

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:

- 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 Chapter 2 in the *System Management Services (SMS) 1.2 Administrator Guide* for more information.

### EXAMPLES

**EXAMPLE 1  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:sm5-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:
<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO
poweron(1M)
**NAME**

poweron – control power up

**SYNOPSIS**

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

**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).

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, the `poweron` operation will fail. 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 will fail.

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 will be prompted to continue. If not, `poweron` exits.

**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:

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

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 Chapter 2 in the System Management Services (SMS) 1.2 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:smx-user:~> poweron PS0

EXAMPLE 2  Powering On a CPU in the ASR Blacklist File

You must have platform administrator privileges. Otherwise, poweron exits with an error.
```
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.

**ATTRIBUTES**

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**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.

**SEE ALSO**
esmd(1M), poweroff(1M)
NAME

rcfgadm – remote configuration administration

SYNOPSIS

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

rcfgadm -d domain_id | domain_tag [-f] [-y] [-n] [-v] [-o hardware_options] -x hardware_function ap_id...

rcfgadm -d domain_id | domain_tag [-v] [-a] [-s listing_options] [-o hardware_options] -l [-l [ap_id|ap_type ]...]

rcfgadm -d domain_id | domain_tag [-v] [-o hardware_options] -t ap_id...

rcfgadm -d domain_id | domain_tag [-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 makes a distinction 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 are performed by calling hardware-specific libraries.

Configuration administration operates on an attachment point. Hardware resources located at attachment points can or cannot 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, while 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 environment. 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 environment. 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 non volatile record keeping.

An attachment point with an occupant in the configured state is in one of four conditions: unknown, ok, failing, 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, unknown, or 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 re-evaluated.

Attachment points are referred to using hardware-specific identifiers (ap_ids) 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 is SB6.

Attachment points can also be created dynamically. A dynamic attachment point is named relative to a base attachment point which 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 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` 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 output on the standard error output for confirmation on the standard input. Confirmation can be overridden using the `−y` or `−n` options 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 15K Dynamic Reconfiguration User Guide for more information.

**OPTIONS**

The following options are supported:

`−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.

**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 it is unrecoverable, you will need to reboot the domain in order to use that board.

The possible transition states and their meaning 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 since another domain might have assigned the board to itself.

- **connect**

  Performs hardware-specific operations to put the receptacle in 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 usable 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), 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 environment.

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. An attempt to override a condition and force a state change that would otherwise fail can be made by...
specifying the force option (-f). Hardware-specific safety and integrity checks can prevent the force option from having any effect.

- **-d domain_id**

  ID for a domain. Valid domain_ids are ‘A’...'R’ and are case insensitive.

- **-d 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 [ap_id | ap_type]**

  Prints out 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. Attachment points can be filtered by using 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.

  - **-o parsable**

    Return the information as a set of "name=value" pairs separated by the space character. All strings will be enclosed within double quotes. Any double quote and "\" characters in a string will be escaped with a "\". The parsable option is intended to be used in conjunction with the -s option of cfgadm.

- **-n**

  104

  SMS 1.2

  Last modified 24 October 2001
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 -l option is used. The parsable suboption specifies info is returned as a set of "name=value" pairs.

- unassign
  Applies only when the -c disconnect option is used. The unassign suboption specifies the domain is to give up ownership of the board.

- nopoweroff
  Applies only when the -c disconnect option is used. The nopoweroff suboption specifies 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. \textit{listing\_options} conforms to the \texttt{getsubopt(3C)} syntax convention. The suboptions are used to specify the attachment point selection criteria (\texttt{select=select\_string}), the type of matching desired (\texttt{match=match\_type}), order of listing (\texttt{sort=field\_spec}), the data that is displayed (\texttt{cols=field\_spec} and \texttt{cols2=field\_spec}), the column delimiter (\texttt{delim=string}) and whether to suppress column headings (\texttt{noheadings}).

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

\begin{verbatim}
rcfgadm -s select=attr1(value1):attr2(value2)...
\end{verbatim}

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

\begin{verbatim}
rcfgadm -s match=match\_type,select=attr1(value1)...
\end{verbatim}

where \texttt{match\_type} can be either \texttt{exact} or \texttt{partial}. The default value is \texttt{exact}.

Suboptions can contain special characters which can be interpreted in ways other than part of \texttt{rcfgadm} suboptions. For example, a command may 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.

\begin{quote}
\texttt{field\_spec} is one or more data-fields concatenated using a colon (:), as in data-field:data-field:data-field. A data-field is one of \texttt{ap\_id, physid, r\_state, o\_state, condition, type, busy, status\_time, status\_time\_p} and \texttt{info}. The \texttt{ap\_id} field output is the logical name for the attachment point, while the \texttt{physid} field contains the physical name. The \texttt{r\_state} field can be \texttt{empty}, \texttt{disconnected}, or \texttt{connected}. The \texttt{o\_state} field can be \texttt{configured} or \texttt{unconfigured}. The \texttt{busy} field can be either \texttt{y} if the attachment point is \texttt{busy}, or \texttt{n} if it is not. The \texttt{type} and \texttt{info} fields are hardware-specific. The \texttt{status\_time\_p} field is a parsable version of the \texttt{status\_time} field. If an attachment point has an associated class, the \texttt{class} field lists the class name.
\end{quote}

The order of the fields in \texttt{field\_spec} is significant. For the sort suboption, the first field given is the primary sort key. For the \texttt{cols} and \texttt{cols2} suboptions, the fields are printed in the order requested. The order of sorting on a data-field can be reversed by placing a minus (-) before the data-field name within the \texttt{field\_spec} for the sort suboption. The default value for sort is \texttt{ap\_id}. The default values for \texttt{cols} and \texttt{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. See NOTES.

−T timeout

Specifies the time interval, in seconds, between retries. This option must be specified with the −r retry_count option. The default value is zero, meaning the DR request is retried immediately.

−t

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

The results of the test are used to update the condition of the specified occupant to either 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 only be set to ok upon normal completion of testing with no errors.

−v

Executes in verbose mode. For the −c, −t, and −x options, outputs a message giving the results of each attempted operation. Outputs detailed help information for the −h option. Outputs verbose information for each attachment point for the −l option.

−x hardware_function

Performs hardware-specific functions.

The following are valid hardware_function:

■ assign ap_id

Assign a board to a domain.

■ unassign ap_id
Unassign a board to a domain.

- `poweron ap_id`
  Power on a board.

- `poweroff ap_id`
  Power off a board.

List hardware-specific private functions using `rcfgadm -h ap_id`.

- `-y` automatically answers “yes” to all prompts. Prompts are displayed.

**OPERANDS**

The following operands are supported:

*ap_id*

Attachment points are referred to using hardware-specific identifiers (*ap_ids*) 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.

**EXAMPLE 1** 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
```

**EXAMPLE 2** 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_ids* begin with *pci*.

**EXAMPLE 3 ap_types**

**Static ap_types**

- HPCI
- CPU
- MCPU
- pci-pci/hp

**Dynamic ap_types**

- cpu
- mem
- io

---

**EXTENDED DESCRIPTION**

**Group Privileges Required**

The privileges required to use this command depend on the desired operation. `rcfgadm` can assign or unassign boards which are not connected to a domain. To assign or unassign a board, you must have either platform administrator privileges or domain administrator/configurator privileges 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_ids* for assign and unassign must be logical *ap_ids* specifying a board such as: SB0 or IO2.

Domain administrator or domain configurator privileges are required for test, state change, or hardware-specific operations.
You must have domain administrator or configurator privileges on the domain specified. Otherwise, you must have platform administrator privileges.

No privileges are required for listing operations.

Refer to Chapter 2 in the *System Management Services (SMS) 1.2 Administrator Guide* for more information.

### EXAMPLES

#### EXAMPLE 4  Listing Attachment Points in the Device Tree for Domain A

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

```bash
sc0:~> rcfgadm -d a
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>IO4</td>
<td>PCI</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO6</td>
<td>MCPU</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>IO14</td>
<td>PCI</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB4</td>
<td>CPU</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>SB6</td>
<td>CPU</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB16</td>
<td>CPU</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
</tbody>
</table>

#### EXAMPLE 5  Listing All Configurable Hardware Information for Domain A

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

```bash
sc0:~> rcfgadm -d a -al
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>IO4</td>
<td>PCI</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO4::pci0</td>
<td>io</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO4::pci1</td>
<td>io</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO4::pci2</td>
<td>io</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO4::pci3</td>
<td>io</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO6</td>
<td>MCPU</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>IO14</td>
<td>PCI</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO14::pci0</td>
<td>io</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO14::pci1</td>
<td>io</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO14::pci2</td>
<td>io</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>IO14::pci3</td>
<td>io</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB4</td>
<td>CPU</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>SB6</td>
<td>CPU</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB6::cpu0</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB6::cpu1</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB6::cpu2</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB6::cpu3</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB6::memory</td>
<td>memory</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB16</td>
<td>CPU</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB16::cpu0</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB16::cpu1</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB16::cpu2</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
</tbody>
</table>
SB16::cpu3 cpu connected configured ok
SB16::memory memory connected configured ok

EXAMPLE 6  Selective Listing Based on Attachment Point Attributes for Domain A

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

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

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB6::cpu0</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB6::cpu1</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB6::cpu2</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>SB6::cpu3</td>
<td>cpu</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
</tbody>
</table>

EXAMPLE 7  Listing Current Configurable Hardware Information in Verbose Mode for Domain A

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

```
sc0:sms-unix:
rcfgadm -d a -v -l SB16
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB16</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
<td>powered-on, assigned</td>
</tr>
</tbody>
</table>

When  Type  Busy  Phys_Id
Mar 6 13:30 CPU n /devices/pseudo/dr@0:SB16

EXAMPLE 8  Force Option on Domain A

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

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

EXAMPLE 9  Unconfiguring an Occupant From the System on Domain A

The following example unconfigures an occupant from the system:

```
sc0:sms-unix:
rcfgadm -d a -c unconfigure IO14
```
EXAMPLE 10  Configuring an Occupant at an Attachment Point

The following example configures an occupant:

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

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

112  SMS 1.2  Last modified 24 October 2001
Maintenance Commands

10 Insufficient condition
11 Invalid
12 Error
13 APID doesn’t 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
56 DR command syntax error
68 DR operation failed

ATTRIBUTES

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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

Last modified 24 October 2001
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 cancelled
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_id | domain_tag] [-d domain_id | domain_tag...] [-q] [-y | -n] [-x]
```

```
reset -h
```

## DESCRIPTION
reset(1M) allows 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 will receive an error if the virtual keys switch is in the `secure` position. An optional confirmation prompt is given by default. Refer to Chapter 6 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

## OPTIONS
- `d domain_id`: ID for a domain. Valid `domain_ids` are ‘A’...'R' and are case insensitive.
- `d 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.

- `-x`: Send 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

Last modified 01 October 2001

SMS 1.2

115
You must have domain administrator privileges to run this command.

Refer to Chapter 2 in the *System Management Services (SMS) 1.2 Administrator Guide* for more information.

**EXAMPLES**

**EXAMPLE 1**  Reset Domain C

```
sc0:smi-user> reset -d C
Do you want to send RESET to domain C? [y|n] : y
RESET to port SB4/P0 initiated.
RESET to port SB4/P1 initiated.
RESET initiated to all ports for domain: C
```

**EXAMPLE 2**  XIR Reset of Domain C

```
sc0:smi-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 descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**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 runs either from the main SC and resets the spare or 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 exits with 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 Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

Last modified 01 October 2001

SMS 1.2
117
EXAMPLES

EXAMPLE 1  Resetting the Other SC Using Prompts

```shell
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

```shell
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

```shell
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

```shell
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.
Unable to reset other SC.
Flag registration failed.
Invalid command line argument.

ATTRIBUTES

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>
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.

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 Chapter 2 in the System Management Services (SMS) 1.2 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>All</td>
</tr>
<tr>
<td>Availability</td>
<td>SUNWSMSoft</td>
</tr>
<tr>
<td>Stability</td>
<td>Evolving</td>
</tr>
<tr>
<td>MT-Level</td>
<td>Safe</td>
</tr>
</tbody>
</table>

SEE ALSO

cancelcmdsyc(1M), initcmdsyc(1M), savecmdsyc(1M), and showcmdsyc(1M)
NAME
initcmdsync, cmdsync, cancelcmdsync, savecmdsync – command synchronization commands

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 will be 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 is used only to mark specific points in a script from which processing can be resumed. If specific restart points are not needed, 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.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>−M identifier</td>
<td>Marks a location in the script from which the script can be resumed after a failover. The identifier must be a positive integer.</td>
</tr>
<tr>
<td>parameters</td>
<td>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.</td>
</tr>
<tr>
<td>script_name</td>
<td>Identifies the name of the user-defined script to be synchronized.</td>
</tr>
</tbody>
</table>

### 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:

```bash
# 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
go_to_label=1
#
# Process the arguments, capturing the −M marker point if provided
#
for arg in $*; do
   case $arg in
      -M )
         goto_label=$arg;;

      .
      .
      .
e 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
```

Last modified 01 October 2001
# 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 initcmdsnc command.

initcmdsnc 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 savecmdsnc 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.
  savecmdsnc -M $(( $goto_label + 1 )) $desc
goto_label=$(( $goto_label + 1 ))
  ;;

  # Step 2: Do more things
  # 2
  do_more_things . .
  .
  savecmdsnc -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
  ;;

  # Step 1: Do something
  # 1
  do_something . . .
  # Execute the savecmdsnc 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.
  savecmdsnc -M $(( $goto_label + 1 )) $desc
goto_label=$(( $goto_label + 1 ))
  ;;

  # Step 2: Do more things
  # 2
  do_more_things . .
  .
  savecmdsnc -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) 1.2
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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>All</td>
</tr>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
<tr>
<td>Stability</td>
<td>Evolving</td>
</tr>
<tr>
<td>MT-Level</td>
<td>Safe</td>
</tr>
</tbody>
</table>

SEE ALSO

runcmdsync(1M), showcmdsync(1M)

NOTES

An example of a user-defined script (with synchronization commands) is
provided in the /opt/SUNWSMS/examples/cmdsync directory.

Last modified 01 October 2001
NAME
setbus – perform dynamic bus reconfiguration on active expanders in a domain

SYNOPSIS
setbus [-q] [-y | -n] -c CS0|CS1|CS0,CS1 [-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 can allow you to swap out a CSB without having to power off the
system.

OPTIONS

Note - The -y and -n are optional arguments to the setbus(1M) command. If
one of these optional arguments is not provided, setbus prompts the user
with a confirmation message.

If changing the configuration on the chosen expander will require changing the
configuration on additional expanders, setbus displays the following
message:

The expander board in position location communicates with expanders not
already listed, and will be added to the list of boards to reconfigure.

-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 CS0|CS1|CS0,CS1  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 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*  Specifies which expander slots to configure. The default is to configure all. Multiple *locations* are separated by spaces.

Valid *locations* are:

EX0–EX17

**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 Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

**EXAMPLES**

**EXAMPLE 1**  Set All Buses on All Active Domains to Use CS0

```
sc0:sms-user> setbus -c CS0
```

**EXAMPLE 2**  Set All Buses on All Active Domains to Use Both CSBs

```
sc0:sms-user> setbus -c CS0,CS1
```

**EXAMPLE 3**  Set Address Bus on All Active Domains to Use CS0

```
sc0:sms-user> setbus -c CS0 -b a
```

**EXAMPLE 4**  Set 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 descriptions of the following attributes:
<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**SEE ALSO**

showbus(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.

**Note** - Data synchronization uses the available disk space under the `/var/opt/SUNWSMS` directory to copy files from the main SC to the spare. If you have files to be copied that are larger than the `/var/opt/SUNWSMS` directory, those files cannot be propagated. For example, if the data synchronization backup file (`ds_backup.cpio`) gets larger than the available space in `/var/opt/SUNWSMS`, you must reduce the size of this backup file before data propagation can occur. The size of the `sms_backup.cpio` file gives you an indication of the size of the data synchronization backup file. To create more disk space you can remove the following files:

- `/var/opt/SUNWSMS/adm/platform/messages.x`
- `/var/opt/SUNWSMS/adm/domain_id/messages.x`
- `/var/opt/SUNWSMS/adm/domain_id/post/files`

where `x` is the archive number of the file. Because these files are propagated from the new main SC to the spare after a failover, remove these files on both the main and spare SC.

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 is to 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 will not be 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`
  Backs up the main SC using `smsbackup(1M)`, moves the backup data from the main to the spare SC, and restores the backup data on the spare SC. For more information, see `smsbackup(1M)`.

- `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.

- `-i interval`
  Indicates how often the specified file is to 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 in the *System Management Services (SMS) 1.2 Administrator Guide* for more information.

**EXAMPLES**

**EXAMPLE 1** Propagate 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** Remove 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>All</td>
</tr>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
<tr>
<td>Stability</td>
<td>Evolving</td>
</tr>
<tr>
<td>MT-Level</td>
<td>Safe</td>
</tr>
</tbody>
</table>
SEE ALSO  showdatasync(1M), smsbackup(1M)

System Management Services (SMS) 1.2 Administrator Guide
NAME  
setdate – set the date and time for the system controller (SC) or a domain

SYNOPSIS  
setdate [−d domain_id | domain_tag] [−u] [−q]  
  [mmdd]HHMM | mmddHHMM[cc][yy][.SS]

setdate  −h

DESCRIPTION  
setdate(1M) allows the SC platform administrator to set the SC or optionally  
a domain date and time values. Allows domain administrators to set the date  
and time values for their domains. After setting the date and time,  
setdate(1M) displays the current date and time.

OPTIONS  
−d domain_id  ID for a domain. Valid domain_ids are ‘A’...'R’ and are case  
insensitive.

Sets the domain’s 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.

−d domain_tag  Name assigned to a domain using addtag(1M).

Sets the domain’s time of day (TOD) when the domain’s  
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 setting 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).
**mmdHHMM[cc]yy[.SS]**  
Date and time format. \textit{m}m is the month (1–12), \textit{dd} is the day of the month (1–31), \textit{HH} is the hour (0–23), \textit{MM} is the minute (0–59), \textit{cc} is century minus one, and \textit{yy} is the two digit year, \textit{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 domin administrator privileges you may only run this command for your domain.

Refer to Chapter 2 in the *System Management Services (SMS) 1.2 Administrator Guide* for more information.

**EXAMPLES**

**EXAMPLE 1** Setting the Local Date in Pacific Standard Time

```
sc0:sms-user> setdate 020210302000.00
System Controller: Wed Feb 2 10:30:00 PST 2000
```

**EXAMPLE 2** Setting the Date Using GMT

```
sc0:sms-user> setdate -u 020218302000.00
System Controller: Wed Feb 2 18:30:00 GMT 2000
```

**EXAMPLE 3** Setting the Local Time in Pacific Standard Time for Domain A

```
sc0:sms-user> setdate -d a 020210302000.00
Domain a: Wed Feb 2 10:30:00 PST 2000
```

**EXAMPLE 4** Setting the Date for Domain A Using GMT

```
sc0:sms-user> setdate -d a -u 020218302000.00
Domain a: Wed Feb 2 18:30:00 GMT 2000
```

Last modified 01 October 2001  
SMS 1.2  
135
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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO  addtag(1M), setkeyswitch(1M), showdate(1M)
NAME

setdefaults – remove all instances of a previously active domain

SYNOPSIS

setdefaults −d domain_id | domain_tag [−p ] [−y ]

setdefaults −h

DESCRIPTION

setdefaults(1M) removes all SMS instances of a previously active domain. A domain instance includes all pcid entries except network information; all message, console, and syslog log files; and, optionally, all NVRAM and boot parameters. pcid entries and NVRAM and boot parameters are returned to system default settings. IDPROM data is not affected.

Only one domain can be done at a time. The domain cannot be active and the virtual keyswitch must be set to off, otherwise, setdefaults exits with an error.

OPTIONS

The following options are supported:
−d domain_id ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.
−d 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 Preserve NVRAM and boot parameter data. By default, you are asked whether to remove the NVRAM and boot parameter data or not. If the −p option is used, you are not prompted and the data is preserved.
−y Automatically answers “yes” to all prompts.

EXTENDED DESCRIPTION

Group Privileges

Required You must have platform administrator or domain administrator privileges for the specified domain to run this command.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

Last modified 01 October 2001
EXAMPLES

EXAMPLE 1  Setting Defaults on Domain A With Domain, NVRAM and Boot Parameter Prompts

```
sc0:~> 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:~> setdefaults -d a -p -y
```

EXAMPLE 3  Setting Defaults on Domain A Without Prompts and Without Saving NVRAM and Boot Parameter Data

```
sc0:~> setdefaults -d a -y
```

EXIT STATUS  The following exit values are returned:

<table>
<thead>
<tr>
<th>Exit Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful completion.</td>
</tr>
<tr>
<td>1</td>
<td>An invalid domain was specified.</td>
</tr>
<tr>
<td>2</td>
<td>An invalid option was entered.</td>
</tr>
<tr>
<td>3</td>
<td>No domain, or more than one domain, was specified.</td>
</tr>
<tr>
<td>4</td>
<td>The user has invalid permission.</td>
</tr>
<tr>
<td>5</td>
<td>The keyswitch is in an invalid position.</td>
</tr>
<tr>
<td>6</td>
<td>The domain is currently active.</td>
</tr>
<tr>
<td>7</td>
<td>An error occurred talking to the pcd.</td>
</tr>
</tbody>
</table>
8 An error occurred talking to the mid.
9 An error occurred talking to the osd.
10 An internal error occurred.
11 The user cancelled the operation.

**ATTRIBUTES**
See attributes(5) for descriptions of the following attributes:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**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 messages 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 messages 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 messages 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.

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 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:

force

Forces a failover to the spare SC. The spare SC must be available.

−h

Help. Displays usage descriptions.

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

off

Disables the failover mechanism. This will prevent a failover until the mechanism is re-enabled.

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 re-enable failover only. If failover cannot be re-enabled, 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 Chapter 2 in the System Management Services (SMS) 1.2
Administrator Guide for more information.

EXAMPLES

These commands produce no output when successful. An error message appears if the action could not be performed.

EXAMPLE 1  Turn Failover On

sc0:sms-user> setfailover on

Last modified 01 October 2001
EXAMPLE 2  Turn Failover Off

```
#sc0:sms-user:> setfailover off
```

EXAMPLE 3  Force a Failover

```
sc0:sms-user:> setfailover force
```

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO  showfailover(1M)
NAME

setkeyswitch – change the position of the virtual keyswitch

SYNOPSIS

setkeyswitch -d domain_id|domain_tag [-q] [-y|n] [on|standby|off|diag|secure]

setkeyswitch -h

DESCRIPTION

setkeyswitch(1M) changes the position of the virtual keyswitch to the specified value. setkeyswitch is responsible for powering on or powering down boards and bringing up a domain. See the OPERANDS section for more information.

If the domain specified contains a board in the automatic system recovery (ASR) blacklist file, an error message is displayed 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

Note - The -y and -n are optional arguments to the setkeyswitch(1M) command. If one of these optional arguments is not provided, setkeyswitch prompts the user for confirmation when changing from the on, diag, or secure position to the off or standby position.

-d domain_id ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.

-d 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.

**OPERANDS**

The following operands are supported:

<table>
<thead>
<tr>
<th>Operands</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>on</code></td>
<td>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, but upon the next reboot of the domain, post is not invoked with verbosity and the diag level is set to its default value. From the secure position, on restores write permission to the domain.</td>
</tr>
<tr>
<td><code>standby</code></td>
<td>From the off position, standby powers on all boards assigned to the domain (if not already powered on). From the on, diag, or secure position, standby optionally causes a confirmation prompt and the domain is gracefully shut down. The boards remain fully powered.</td>
</tr>
<tr>
<td><code>off</code></td>
<td>From the on, diag, or secure position, off optionally causes a confirmation prompt and all boards are put into low-power mode. From the standby position, off puts all boards into low-power mode.</td>
</tr>
<tr>
<td><code>diag</code></td>
<td>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...</td>
</tr>
</tbody>
</table>
invoked with the verbosity and diag levels set to at least their defaults.

From the on position, diag results in nothing more than a position change, but upon the next reboot of the domain, post is invoked with the verbosity and diag levels set to at least their defaults.

From the secure position, diag restores write permission to the domain and upon the next reboot, post is invoked with the verbosity and diag levels set to at least their defaults.

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, flashupdates, and resets will not work.

From the on position, secure removes write permission to the domain (as described above).

From the diag position, secure removes write permission to the domain (as described above) and on the next reboot of the domain, post is invoked with the verbosity and diag levels set to at least their defaults.

You must have domain administrator privileges for the specified domain to run this command.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

**EXAMPLE 1** Set Keyswitch on Domain A On

```
s0:sm0-user:~> setkeyswitch -d A on
```

Last modified 01 October 2001
EXAMPLE 2  Using Keyswitch on a Domain Containing a Board in the ASR Blacklist File

sc0:sms-user:>setkeys -d A on
SB0 is in the ASR Blacklist.

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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.

SEE ALSO
addtag(1M), esmd(1M), flashupdate(1M), pcd(1M), reset(1M),
showkeys(1M)
NAME

setobpparams – set up OpenBoot PROM variables for a domain

SYNOPSIS

setobpparams
−d domain_id | domain_tag param=value...

setobpparams −h

DESCRIPTION

setobpparams(1M) allows 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 routine system administration. For more information refer to Chapter 4 in the System Management Services (SMS) 1.2 Administrator Guide.

OPTIONS

The following options are supported:

−d domain_id
ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.

−d 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?
auto-boot?
fcode-debug?
use-nvramrc?
security-mode

Valid variable values for all but security mode are:

ture
false

Valid variable values for security mode are:

none
command
full

where:
**none** - No password required (default)
**command** - All commands except for `boot(1M)` and `go` require the password
**full** - 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 shells require using single quotes around the variable values to prevent the '?' from being treated as a special character. See the examples below.

---

**EXTENDED DESCRIPTION**

**Group Privileges Required**

Domain administrator or configurator privileges for the specified domain are required.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

**EXAMPLES**

**EXAMPLE 1** Setting OpenBoot PROM Variable `diag-switch` On for Domain A

```
sc0:~> setobpparams -d a 'diag-switch?=true'
```
EXAMPLE 2  Setting OpenBoot PROM Variable security-mode to Full for Domain A

    sc0:sm-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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO  addtag(1M), setkeys(1M), showobpparams(1M)
NAME  setupplatform – set up the available component list for domains

SYNOPSIS  setupplatform [−d domain_id | domain_tag [−a | −r] location [location]...]
setupplatform [−d domain_id | domain_tag location [location]...]
setupplatform [−d domain_id | domain_tag −]
setupplatform −h

DESCRIPTION  setupplatform(1M) sets up the available component list for domains. If a
domain_id | domain_tag is specified, a list of boards must be specified. An empty
board list can be specified as ‘−’. In the case where no domain_id | domain_tag is
specified, current values are displayed in the “[]” at the command prompt. If
no value is specified for a parameter, it will retain its current value.

OPTIONS
−a  Add the slot(s) to the available component list for the
domain..
−d domain_id  ID for a domain. Valid domain_ids are ‘A’...’R’ and are case
insensitive.
−d 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.
−r  Remove the slots from the available component list for the
domain.
−  Clears the entire available component list.

OPERANDS  The following operands are supported:
location  Board location separated by a space.
The following location forms are accepted:
SB(0...17)
IO(0...17)
EXTENDED DESCRIPTION

Group Privileges Required

You must have platform administrator privileges to run this command.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

Use showplatform(1M) to display the available component list once you have run setupplatform.

EXAMPLE 1  Set 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 []?
```

```
sc0:sms-user:> showplatform -p available
Available for domain DomainA:
    SB3 SB2
    IO5 IO4 IO3

Available for domain DomainB:
        None
        None
```
Available for domain DomainC:
SB17 SB16 SB7 SB5
   IO8 IO7

Available for domain D:
SB9 SB8 SB4 SB2
   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  Set Up Available Component List for Domain engB to Boards at SB0, IO1, and IO2

```
sc0:sms-user:~> setupplatform -d engB SB0 IO1 IO2
```

EXAMPLE 3  Clear All Boards in engB Available Component List

```
%sc0:sms-user:~> setupplatform -d engB -
```

EXAMPLE 4  Add Boards at SB0 and IO2 to engB Available Component List

```
sc0:sms-user:~> setupplatform -d engB -a SB0 IO2
```

EXAMPLE 5  Remove Boards at SB3 and IO3 From engB Available Component List

```
sc0:sms-user:~> setupplatform -d engB -r SB3 IO3
```

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSoP</td>
</tr>
</tbody>
</table>

Last modified 01 October 2001
SEE ALSO

addtag(1M), showplatform(1M)
NAME  showboards – show the assignment information and status of the boards

SYNOPSIS  showboards [−d domain_id|domain_tag] [−v ]
showboards [−h ]

DESCRIPTION  showboards(1M) displays board assignments. If domain_id|domain_tag is specified, this command displays which boards are assigned or available to the given domain. If the −v option is used, showboards displays all components, including domain configuration units (DCUs) such as cpus, dpus, iobds, csbs and exbs, as well as the system controller (SC) are not DCUs.

OPTIONS  The following options are supported:
−d domain_id  ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.
−d 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 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 Chapter 2 in the System Management Services (SMS) 1.2 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|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 | domain_tag` and the `−a` 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 | 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 which 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.

### States in the Pwr Field

The `Pwr` field contains one of three measurements:

- **On** = Full voltage detected.
- **Off** = No voltage detected
- **Min** = Some voltage detected.
- **Unk** = Unknown. Unable to determine board power state.
- **—** = The slot is empty so power state is not applicable.

Min does not imply that the board may be used at this point 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 contains one of six values:

Passed. = The board passed POST.
Degraded = The board is in a degraded mode.
Failed = The board failed POST.
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.

EXAMPLES

EXAMPLE 1  Showboards for Platform Administrators

```
sc0:sms-user$ showboards
```

<table>
<thead>
<tr>
<th>Location</th>
<th>Pwr</th>
<th>Type of Board</th>
<th>Board Status</th>
<th>Test Status</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB0</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>domainC</td>
</tr>
<tr>
<td>SB1</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>A</td>
</tr>
<tr>
<td>SB2</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB3</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB4</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB5</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB6</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB7</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>domainC</td>
</tr>
<tr>
<td>SB8</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB9</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>dmnJ</td>
</tr>
<tr>
<td>SB10</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB11</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB12</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB13</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB14</td>
<td>Off</td>
<td>CPU</td>
<td>Assigned</td>
<td>Failed</td>
<td>domainC</td>
</tr>
</tbody>
</table>
**EXAMPLE 2**  Showboards for Platform Administrators for Domain B

The following example illustrates showboards output if you have platform administrator privileges and specify a domain. The output does not include boards which are assigned to other domains.

```
sc0:/sms-user: > showboards -d b
```

<table>
<thead>
<tr>
<th>Location</th>
<th>Pwr</th>
<th>Type of Board</th>
<th>Board Status</th>
<th>Test Status</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB3</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB4</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB5</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB6</td>
<td>Off</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB8</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB10</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB11</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB12</td>
<td>Off</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB13</td>
<td>Off</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO0</td>
<td>Off</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO2</td>
<td>Off</td>
<td>HPCI</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>IO5</td>
<td>Off</td>
<td>HPCI</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO6</td>
<td>Off</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO10</td>
<td>Off</td>
<td>HPCI</td>
<td>Assigned</td>
<td>Unknown</td>
<td>engB</td>
</tr>
<tr>
<td>IO11</td>
<td>Off</td>
<td>HPCI</td>
<td>Assigned</td>
<td>Failed</td>
<td>engB</td>
</tr>
<tr>
<td>IO12</td>
<td>Off</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO13</td>
<td>Off</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO14</td>
<td>Off</td>
<td>HPCI</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
</tbody>
</table>
```
### EXAMPLE 3  Showboards for Platform Administrators Using the −v Option

The following example illustrates showboards output if you have platform administrator privileges and use the −v option. The command shows all components.

```
s0:sm-user> showboards -v
```

<table>
<thead>
<tr>
<th>Location</th>
<th>Pwr</th>
<th>Type of Board</th>
<th>Board Status</th>
<th>Test Status</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC0</td>
<td>On</td>
<td>SC</td>
<td>Slave</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC1</td>
<td>On</td>
<td>SC Master</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PS0</td>
<td>On</td>
<td>PS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PS1</td>
<td>On</td>
<td>PS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PS2</td>
<td>On</td>
<td>PS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td>Off</td>
<td>Empty Slot</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td>Off</td>
<td>PS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PS5</td>
<td>On</td>
<td>PS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FT0</td>
<td>On</td>
<td>FT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FT1</td>
<td>On</td>
<td>FT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FT2</td>
<td>On</td>
<td>FT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FT3</td>
<td>On</td>
<td>FT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FT4</td>
<td>On</td>
<td>FT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FT5</td>
<td>Off</td>
<td>Empty Slot</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FT6</td>
<td>Off</td>
<td>FT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FT7</td>
<td>On</td>
<td>FT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS0</td>
<td>On</td>
<td>CSB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS1</td>
<td>On</td>
<td>CSB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX0</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX1</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX2</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX3</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX4</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX5</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX6</td>
<td>Off</td>
<td>Empty Slot</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX7</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX8</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX9</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX10</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX11</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX12</td>
<td>Off</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX13</td>
<td>Off</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX14</td>
<td>Off</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX15</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX16</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX17</td>
<td>On</td>
<td>EXB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IO1/C3V0</td>
<td>On</td>
<td>C3V</td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>IO1/C5V0</td>
<td>On</td>
<td>C5V</td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>IO1/C3V1</td>
<td>On</td>
<td>C3V</td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>IO1/C5V1</td>
<td>On</td>
<td>C5V</td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>IO2/C3V0</td>
<td>Off</td>
<td>C3V</td>
<td></td>
<td></td>
<td>engB</td>
</tr>
<tr>
<td>IO2/C5V0</td>
<td>On</td>
<td>C5V</td>
<td></td>
<td></td>
<td>engB</td>
</tr>
<tr>
<td>IO2/C3V1</td>
<td>On</td>
<td>C3V</td>
<td></td>
<td></td>
<td>engB</td>
</tr>
<tr>
<td>IO2/C5V1</td>
<td>On</td>
<td>C5V</td>
<td></td>
<td></td>
<td>domainC</td>
</tr>
<tr>
<td>IO3/C3V0</td>
<td>On</td>
<td>C3V</td>
<td></td>
<td></td>
<td>domainC</td>
</tr>
<tr>
<td>IO3/C5V0</td>
<td>On</td>
<td>C5V</td>
<td></td>
<td></td>
<td>domainC</td>
</tr>
<tr>
<td>IO3/C3V1</td>
<td>Off</td>
<td>Empty Slot</td>
<td></td>
<td></td>
<td>domainC</td>
</tr>
</tbody>
</table>

Last modified 09 October 2001  SMS 1.2  159
| SB0 | On CPU | Active | Passed | domainC |
| SB1 | On CPU | Active | Passed | A |
| SB2 | On CPU | Active | Passed | engB |
| SB3 | On CPU | Active | Passed | engB |
| SB4 | On CPU | Active | Passed | engB |
| SB5 | On CPU | Active | Passed | engB |
| SB6 | Empty Slot | Available | Unknown | Isolated |
| SB7 | On CPU | Active | Passed | domainC |
| SB8 | Off CPU | Available | Unknown | Isolated |
| SB9 | On CPU | Active | Passed | dmJ |
| SB10 | Off CPU | Available | Unknown | Isolated |
| SB11 | Off CPU | Available | Unknown | Isolated |
| SB12 | Empty Slot | Available | Unknown | Isolated |
| SB13 | Empty Slot | Available | Unknown | Isolated |
| SB14 | Off CPU | Assigned | Failed | domainC |
| SB15 | On CPU | Active | Passed | P |
### EXAMPLE 4  Showboards for Domain Administrators With Privileges on Domains B, J, and R

The following example illustrates showboards output if you have domain privileges for domains B, J, and R. showboards displays information for those boards which are assigned or available to domains B, J, and R. Boards which are assigned to other domains or do not appear in the available component list for domains B, J, or R are not displayed.

```bash
sc0:sms-user:~ $ showboards
```

<table>
<thead>
<tr>
<th>Location</th>
<th>Pwr</th>
<th>Type of Board</th>
<th>Board Status</th>
<th>Test Status</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB3</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB4</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB5</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB6</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB8</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB9</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>dmnJ</td>
</tr>
<tr>
<td>SB10</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB11</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB12</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td></td>
<td>Isolated</td>
</tr>
<tr>
<td>SB13</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td></td>
<td>Isolated</td>
</tr>
<tr>
<td>SB17</td>
<td>-</td>
<td>Empty Slot</td>
<td>Assigned</td>
<td></td>
<td>dmnR</td>
</tr>
<tr>
<td>I00</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td></td>
<td>Isolated</td>
</tr>
<tr>
<td>I02</td>
<td>On</td>
<td>HPCI</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>I05</td>
<td>Off</td>
<td>HPCI</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>I06</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td></td>
<td>Isolated</td>
</tr>
<tr>
<td>I07</td>
<td>On</td>
<td>HPCI</td>
<td>Active</td>
<td>Passed</td>
<td>dmnJ</td>
</tr>
<tr>
<td>I09</td>
<td>On</td>
<td>HPCI</td>
<td>Assigned</td>
<td>iPOST</td>
<td>dmnJ</td>
</tr>
<tr>
<td>I10</td>
<td>Off</td>
<td>HPCI</td>
<td>Assigned</td>
<td>Unknown</td>
<td>engB</td>
</tr>
<tr>
<td>I11</td>
<td>Off</td>
<td>HPCI</td>
<td>Assigned</td>
<td>Failed</td>
<td>engB</td>
</tr>
<tr>
<td>I17</td>
<td>-</td>
<td>Empty Slot</td>
<td>Assigned</td>
<td></td>
<td>dmnR</td>
</tr>
</tbody>
</table>

Last modified 09 October 2001  SMS 1.2  161
EXAMPLE 5  Showboards for Domain Administrators for Domain B

In the following example, showboards displays output if you have domain privileges on domains B, J and R. The command shows board information for those DCUs which are assigned or available to the specified domain. DCUs which are assigned to other domains or do not appear in the specified domain’s available component list are not displayed.

```
sc0:sms-user:~> showboards -d b
```

<table>
<thead>
<tr>
<th>Location</th>
<th>Pwr</th>
<th>Type of Board</th>
<th>Board Status</th>
<th>Test Status</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB3</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB4</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB5</td>
<td>On</td>
<td>CPU</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>SB6</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB8</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB10</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB11</td>
<td>Off</td>
<td>CPU</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB12</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>SB13</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO0</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO2</td>
<td>On</td>
<td>HPCI</td>
<td>Active</td>
<td>Passed</td>
<td>engB</td>
</tr>
<tr>
<td>IO5</td>
<td>Off</td>
<td>HPCI</td>
<td>Available</td>
<td>Unknown</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO6</td>
<td>-</td>
<td>Empty Slot</td>
<td>Available</td>
<td>-</td>
<td>Isolated</td>
</tr>
<tr>
<td>IO10</td>
<td>Off</td>
<td>HPCI</td>
<td>Assigned</td>
<td>Unknown</td>
<td>engB</td>
</tr>
<tr>
<td>IO11</td>
<td>Off</td>
<td>HPCI</td>
<td>Assigned</td>
<td>Failed</td>
<td>engB</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSOp</td>
</tr>
</tbody>
</table>
SEE ALSO  addtag(1M), setupplatform(1M), showplatform(1M)
NAME
showbus – display the bus configuration of expanders in active domains

SYNOPSIS

showbus

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

−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

You must have platform administrator, operator or service privileges to display all set 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 Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  Showbus Display 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 Data Address Response SOCX
-------------------------------------------------
EX0 CS0 CS1 CS0 0x0001
EX1 UNCONF UNCONF UNCONF UNCONF
EX2 UNCONF UNCONF UNCONF UNCONF

164  SMS 1.2  Last modified 01 October 2001
EXAMPLE 2  Display Showbus Information for All Domains Using \texttt{−v}

\texttt{sms-user:}\textgreater; \texttt{showbus \textbar{−v}}

\texttt{--------------}
\texttt{SOCX: 0x14010}
\texttt{--------------}
\texttt{    Data: CS0,CS1}
\texttt{    Address: CS0,CS1}
\texttt{    Response: CS0,CS1}
\texttt{--------------}
\texttt{Domain:A keyswitch: ON}
\texttt{Location:EX4 SB4:active IO4:active}
\texttt{Location:EX14 IO14:active}
\texttt{Location:EX16 IO16:active}
\texttt{--------------}
\texttt{SOCX: 0x00001}
\texttt{--------------}
\texttt{    Data: CS0}
\texttt{    Address: CS1}
\texttt{    Response: CS0}
\texttt{--------------}
\texttt{Domain:B keyswitch: ON}
\texttt{Location:EX0 SB0:active IO0:active}
\texttt{--------------}
\texttt{UNCONFIGURED}
\texttt{--------------}
\texttt{Domain: A keyswitch: ON}
\texttt{Location:EX6 SB6:unknown}

EXIT STATUS  The following exit values are returned:

\begin{itemize}
    \item \texttt{0}  Successful completion.
\end{itemize}

Last modified 01 October 2001  SMS 1.2  165
>0          An error occurred.

**ATTRIBUTES**

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**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 Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1 Example Command Synchronization List

% sc0:sms-user:~ > showcmdsync

<table>
<thead>
<tr>
<th>DESCRIPTOR</th>
<th>IDENTIFIER</th>
<th>CMD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-1</td>
<td>c1 a1 a2</td>
</tr>
</tbody>
</table>

Last modified 01 October 2001
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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO  cancelcmdsnc(1M), initcmdsnc(1M), runcmdsnc(1M),
           savecmdsnc(1M)
NAME
showcomponent – display the blacklist status for a component

SYNOPSIS
showcomponent [-a | -d domain_tag | domain_id] [-v] [location...]
showcomponent -h

DESCRIPTION
showcomponent(1M) displays 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 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 and 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 Chapter 7 in the System Management Services (SMS) 1.2 Administrator
Guide.

OPTIONS
-a Specifies the ASR blacklist.
-d domain_id ID for a domain. Valid domain_ids are 'A'...'R' and are case
insensitive. This option specifies the domain blacklist.
-d domain_tag Name assigned to a domain using addtag(1M). This option
specifies the domain blacklist.
-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/bank/all_banks_on_that_proc`
- `board_loc/proc/bank/all_banks_on_that_board`
- `board_loc/proc`
- `board_loc/procs`
- `board_loc/cassette`
- `board_loc/bus`

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:

- 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 will cause `disablecomponent` to exit and display an error message.

The following *proc* forms are accepted:
P(0..3)
P(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 hsPCI assemblies contain hot-swappable cassettes.
The following hsPCI forms are accepted:

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

There are three bus locations: address, data and response.
The following bus forms are accepted:

ABUS|DBUS|RBU (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 may only run this command on the domain for which you have privileges.

Refer to Chapter 2 in the *System Management Services (SMS) 1.2 Administrator Guide* for more information.

**EXAMPLES**

**EXAMPLE 1**  Display Whether SB0 is ASR Blacklisted

```
sc0sms-user> showcomponent -a SB0
Component SB0 is disabled: #High Voltage
```

**EXAMPLE 2**  Display Whether 4 Boards/Components in Domain B Are Blacklisted

```
sc0sms-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 3**  Display Whether the Logical Bank on IO7 in Domain B Is Blacklisted

```
sc0sms-user> showcomponent -dB IO7/P0/B1/L0
Component IO7/P0/B1/L0 is disabled: <no reason given>
```
EXAMPLE 4  Display All Platform Blacklisted Components
sc0:~$ showcomponent
Component SB0 is disabled: #High temp
Component SB3 is disabled:
Component IO2 is disabled. <no reason given>

EXAMPLE 5  Display All DomainB Blacklisted Components
sc0:~$ showcomponent -dB
Component IO4/PP0 is disabled: #High temp
Component SB5 is disabled: <no reason given>

EXAMPLE 6  Display All ASR Blacklisted Components
sc0:~$ showcomponent -a
Component SB0 is disabled: #High temp

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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.

/etc/opt/SUNWSMS/config/platform/blacklist

List of platform components excluded.

/etc/opt/SUNWSMS/config/domain_id/blacklist

List of domain components excluded.

**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. See the EXTENDED DESCRIPTION section for details on the information displayed.

- **-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 the data synchronization process is enabled and functioning normally.

Last modified 01 October 2001
Disabled indicates the data synchronization process has been disabled because SC failover was disabled.

Failed indicates the data synchronization process cannot currently propagate files to the spare SC even though an SC failure was detected.

### 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`, where:

- **Time Propagated**: 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.
- **File**: 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 Chapter 2 in the *System Management Services (SMS) 1.2 Administrator Guide* for more information.

### EXAMPLES

**EXAMPLE 1  Data Synchronization Status**

```bash
sc0:~> showdatasync
File Propagation State: ACTIVE
Active File: -
Queued files: 0
```

176 SMS 1.2 Last modified 01 October 2001
EXAMPLE 2  Data Synchronization List

    sc0:sms-user:> showdatasync -l
    TIME PROPAGATED  INTERVAL  FILE
    Mar 23 16:00:00    60      /tmp/t1

EXAMPLE 3  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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

setdatasync(1M)

System Management Services (SMS) 1.2 Administrator Guide
NAME  
showdate – display the date and time for the system controller (SC) or a domain

SYNOPSIS  
showdate [-d domain_id | domain_tag] [-u] [-v]

showdate -h

DESCRIPTION  
showdate (1M) displays the SC’s current date and time. Optionally, showdate displays domain time of day.

OPTIONS

-  
-d domain_id  ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.

-  
-d 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, 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 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  Showing the Current Local Date in Pacific Standard Time

sc0sms-user> showdate
EXAMPLE 2  Showing the Current Date Using GMT

sc0:~user> showdate -u

EXAMPLE 3  Showing the Current Local Date on Domain A in Pacific Standard Time

sc0:~user> showdate -d a

EXAMPLE 4  Showing the Current Date on Domain A Using GMT

sc0:~user> showdate -d a -u

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO  addtag(1M), setdate(1M)
showdevices(1M)  Maintenance Commands

NAME
showdevices – display system board devices and resource usage information

SYNOPSIS
showdevices [-v] [-p bydevice | byboard | query | force] location [location...]
showdevices [-v] [-p bydevice | byboard | query | force] -d
domain_id | domain_tag
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. Display the predicted impact of a system board DR operation by performing an offline query of managed resources. Unmanaged devices are not displayed by default, you must use the -v option.

showdevices gathers device information from one or more Sun Fire 15K domains. The command uses the dca(1M) as a proxy to gather the information from the domains.

OPTIONS
The following options are supported:

- domain_id
  ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive. Displays device and resource information for all configured boards in the domain.

- domain_tag
  Name assigned to a domain using addtag(1M). Displays device and resource information for all configured boards in the domain.

- 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 — Query predicted result 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 \texttt{cfgadm}(1M)).

\texttt{\(-v\)} 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:

\texttt{location} List of board locations separated by a space. Multiple \texttt{location} arguments are permitted.

The following \texttt{location} forms are accepted:

\texttt{SB(0...17)}
\texttt{IO(0...17)}

**EXTENDED DESCRIPTION**

The showdevice fields are:

\texttt{domain} Tag or identifier
\texttt{board} Board identifier

CPU:

\texttt{id} Processor id
\texttt{state} Processor state
\texttt{speed} CPU frequency in MHz
\texttt{ecache} CPU ecache size in MB

Memory:

\texttt{board mem} Board memory size in MB

Last modified 01 October 2001    SMS 1.2    181
showdevices(1M) Maintenance Commands

perm mem
Amount of non-relocatable 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
You must have domain administrator/configurator privileges on all boards specified to run this command.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES
EXAMPLE 1  Showdevices for System Board IO1

    sc0:SMS-user:> showdevices IO1
    IO Devices
    ----------
    domain location 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"

182  SMS 1.2  Last modified 01 October 2001
EXAMPLE 2  Showdevices for Domain A

```
sc0:sms-user:> showdevices -v -d A
CPU
-----
domain location id state speed ecache usage
A  C1  40 online 400 4
A  C1  41 online 400 4
A  C1  42 online 400 4
A  C1  43 online 400 4
A  C2  55 online 400 4
A  C2  56 online 400 4
A  C2  57 online 400 4
A  C2  58 online 400 4

Memory

<table>
<thead>
<tr>
<th>domain</th>
<th>location</th>
<th>mem MB</th>
<th>perm</th>
<th>base</th>
<th>target</th>
<th>deleted</th>
<th>remaining</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>C1</td>
<td>2048</td>
<td>723</td>
<td>0x600000</td>
<td>4096</td>
<td>250</td>
<td>1500</td>
</tr>
<tr>
<td>A</td>
<td>C2</td>
<td>2048</td>
<td>0</td>
<td>0x200000</td>
<td>4096</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

IO Devices

<table>
<thead>
<tr>
<th>domain</th>
<th>location</th>
<th>device</th>
<th>resource</th>
<th>usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>IO1</td>
<td>sd0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>IO1</td>
<td>sd1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>IO1</td>
<td>sd2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| 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"
| A      | IO1      | sd4    |          |       |
| A      | IO1      | sd5    |          |       |
| A      | IO1      | sd6    |          |       |
```

EXAMPLE 3  Display Offline Query Result for System Board IO1

```
sc0:sms-user:> showdevices -p query IO1
Location IO1 - Domain A

IO Devices

<table>
<thead>
<tr>
<th>device</th>
<th>resource</th>
<th>query</th>
<th>usage/reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>sd3</td>
<td>/dev/dsk/c0t3d0s0</td>
<td>fail</td>
<td>mounted filesystem &quot;/&quot;</td>
</tr>
<tr>
<td>sd3</td>
<td>/dev/dsk/c0t3s0s1</td>
<td>fail</td>
<td>dump device (swap)</td>
</tr>
<tr>
<td>sd3</td>
<td>/dev/dsk/c0t3s0s1</td>
<td>fail</td>
<td>swap area</td>
</tr>
<tr>
<td>sd3</td>
<td>/dev/dsk/c0t3d0s3</td>
<td>fail</td>
<td>mounted filesystem &quot;/var&quot;</td>
</tr>
<tr>
<td>sd3</td>
<td>/var/run</td>
<td></td>
<td>mounted filesystem &quot;/var/run&quot;</td>
</tr>
</tbody>
</table>
```
The query field shows the predicted result of removing the resource. The failure of the mounted filesystem /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 communicating with pcd.
- 5  An error occurred communicating with a domain.
- 6  An error occurred handling device information.
- 7  An internal error such as failed memory allocation occurred.

**ATTRIBUTES**

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**SEE ALSO**

addtag(1M), dca(1M), pcd(1M)
showenvironment (1M)

NAME
showenvironment – display the environmental data

SYNOPSIS
showenvironment [-d domain_id | domain_tag ...] [-p temps | volts | currents | fans | powers | faults...] [-v ]

DESCRIPTION
showenvironment (1M) displays the environmental data (temperatures, voltages, and so on). If a domain domain_id | domain_tag is specified, environmental data relating to the domain will be displayed providing that the user has domain privileges for that domain. If a domain is not specified, all domain data permissible to the user will be displayed.

Note - Only domain configuration units (DCUs) (for example, CPU, 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, faults and fan tray status with the -p option. If the -p option is not present, all reports are shown.

OPTIONS
The following options are supported:
-h Help. Displays usage descriptions.

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

-d domain_id ID for a domain. Valid domain_ids are 'A'...'R' and are case insensitive.

d domain_tag Domain name assigned to the domain using addtag (1M).

-p Display specific reports. Multiple report arguments are separated by commas.

Valid arguments for -p 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.
faults — List output is of all component readings not within the optimum thresholds.

**Note** - The `faults` argument may not be used in conjunction with any other report argument.

−v  
Verbose. Displays all available command information.

### States in the Display Fields

The Unit field contains one of three measurements:

- C = Celsius
- V = Volts
- A = Ampere

The Status field can contain one of 16 states.

**Temperature Readings:**

- OVERLIMIT = Overlimit
- 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 | Both companion component readings are within 10% of each other. |
| BAD | Both companion component readings are not within 10% of each other. |
| 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 will be displayed. Otherwise, you must have platform administrator, operator or service privileges.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

**EXAMPLES**

**EXAMPLE 1**  Example showenvironment Display for All Domains

```
sc0:sms-user:> showenvironment

<table>
<thead>
<tr>
<th>LOCATION</th>
<th>DEVICE</th>
<th>SENSOR</th>
<th>VALUE</th>
<th>UNIT</th>
<th>AGE</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC at SC0</td>
<td>max1617</td>
<td>RIO Temp</td>
<td>31.00</td>
<td>C</td>
<td>23.4 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SC at SC0</td>
<td>max1617</td>
<td>PCIB Temp</td>
<td>26.00</td>
<td>C</td>
<td>23.4 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SC at SC0</td>
<td>pcf8591</td>
<td>P51 Temp</td>
<td>31.97</td>
<td>C</td>
<td>23.4 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SC at SC0</td>
<td>sbbc</td>
<td>SBBC Temp</td>
<td>40.50</td>
<td>C</td>
<td>23.4 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SC at SC0</td>
<td>cbh</td>
<td>CBH Temp</td>
<td>45.16</td>
<td>C</td>
<td>23.4 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SCPER at SCPER0</td>
<td>max1617</td>
<td>AMB 0 Temp</td>
<td>22.00</td>
<td>C</td>
<td>24.1 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SCPER at SCPER0</td>
<td>max1617</td>
<td>AMB 1 Temp</td>
<td>22.00</td>
<td>C</td>
<td>24.1 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SCPER at SCPER0</td>
<td>max1617</td>
<td>AMB 2 Temp</td>
<td>22.00</td>
<td>C</td>
<td>24.1 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SC at SC0</td>
<td>pcf8591</td>
<td>1.5 VDC</td>
<td>1.46</td>
<td>V</td>
<td>24.7 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SC at SC0</td>
<td>pcf8591</td>
<td>3.3 VDC</td>
<td>3.26</td>
<td>V</td>
<td>24.7 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SC at SC0</td>
<td>pcf8591</td>
<td>3.3 VDC HK</td>
<td>3.28</td>
<td>V</td>
<td>24.7 sec</td>
<td>OK</td>
</tr>
<tr>
<td>SC at SC0</td>
<td>pcf8591</td>
<td>5.0 VDC</td>
<td>5.01</td>
<td>V</td>
<td>24.7 sec</td>
<td>OK</td>
</tr>
</tbody>
</table>
```

Last modified 20 November 2001
## showenvironment(1M) Maintenance Commands

| SC at SC0 | pcf8591 | +12.0 VDC | 11.95 V | 24.7 sec | OK |
| SC at SC0 | pcf8591 | -12.0 VDC | -12.01 V | 24.7 sec | OK |
| SC at SC0 | pcf8591 | 1.5 CVT0 VDC | 1.59 V | 24.7 sec | OK |
| SC at SC0 | pcf8591 | 1.5 CVT1 VDC | 1.60 V | 24.7 sec | OK |
| SC at SC0 | pcf8591 | 3.3 VDC HK | 3.26 V | 25.0 sec | OK |
| SC at SC0 | pcf8591 | 5.0 VDC | 5.04 V | 25.0 sec | OK |
| SC at SC0 | pcf8591 | +12.0 VDC | 12.55 V | 25.0 sec | OK |
| SC at SC1 | max1617 RIO Temp | 36.00 C | 21.8 sec | OK |
| SC at SC1 | max1617 PCSB Temp | 29.00 C | 21.9 sec | OK |
| SC at SC1 | pcf8591 | 1.5 VDC | 1.48 V | 21.9 sec | OK |
| SC at SC1 | pcf8591 | 3.3 VDC | 3.28 V | 21.9 sec | OK |
| SC at SC1 | pcf8591 | 3.3 VDC HK | 3.26 V | 21.9 sec | OK |
| SC at SC1 | pcf8591 | 5.0 VDC | 5.01 V | 21.9 sec | OK |
| SC at SC1 | pcf8591 | +12.0 VDC | 12.41 V | 21.9 sec | OK |
| SC at SC1 | pcf8591 | -12.0 VDC | -12.01 V | 21.9 sec | OK |
| SC at SC1 | pcf8591 | 1.5 CVT0 VDC | 1.60 V | 21.9 sec | OK |
| SC at SC1 | pcf8591 | 1.5 CVT1 VDC | 1.60 V | 21.9 sec | OK |
| SC at SC1 | pcf8591 | 3.3 V_PS0 | 7.76 A | 57.8 sec | BAD |
| SC at SC1 | pcf8591 | 3.3 V_PS1 | 6.59 A | 57.8 sec | BAD |
| SC at SC1 | pcf8591 | 5.0 V_PS0 | 5.12 A | 57.8 sec | BAD |
| SC at SC1 | pcf8591 | 5.0 V_PS1 | 3.90 A | 57.8 sec | BAD |
| CSB at CS0 | max1617 AMB Top Temp | 23.00 C | 21.4 sec | OK |
| CSB at CS0 | max1617 AMB Bot Temp | 21.00 C | 21.4 sec | OK |
| CSB at CS0 | max1617 AMB Temp | 31.83 C | 21.4 sec | OK |
| CSB at CS0 | pcf8591 | 1.5 VDC | 1.51 V | 21.4 sec | OK |
| CSB at CS0 | pcf8591 | 3.3 VDC | 3.28 V | 21.4 sec | OK |
| CSB at CS0 | pcf8591 | 2.5 VDC | 2.52 V | 21.4 sec | OK |
| CSB at CS0 | pcf8591 | 3.3 VDC HK | 3.26 V | 21.4 sec | OK |
| CSB at CS1 | max1617 AMB Top Temp | 25.00 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Bot Temp | 23.00 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 33.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 31.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 32.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 33.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 34.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 35.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 36.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 37.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 38.83 C | 21.0 sec | OK |
| CSB at CS1 | max1617 AMB Temp | 39.83 C | 21.0 sec | OK |
| CP at CP0 | dmx0 | DMX0 Temp | 19.62 C | 21.7 sec | OK |
| CP at CP0 | dmx1 | DMX1 Temp | 20.54 C | 21.7 sec | OK |
| CP at CP0 | dmx3 | DMX3 Temp | 16.44 C | 21.7 sec | OK |
| CP at CP0 | dmx5 | DMX5 Temp | 22.39 C | 21.7 sec | OK |
| CP at CP0 | amx0 | AMX0 Temp | 25.22 C | 21.7 sec | OK |
| CP at CP0 | amx1 | AMX1 Temp | 27.14 C | 21.7 sec | OK |
| CP at CP0 | rmx | RMX Temp | 20.54 C | 21.7 sec | OK |
| CP at CP0 | darb | DARB Temp | 25.70 C | 21.7 sec | OK |
| CP at CP1 | dmx0 | DMX0 Temp | 17.41 C | 21.3 sec | OK |
| CP at CP1 | dmx1 | DMX1 Temp | 33.03 C | 21.3 sec | OK |
| CP at CP1 | dmx3 | DMX3 Temp | 25.10 C | 21.3 sec | OK |
| CP at CP1 | dmx5 | DMX5 Temp | 18.74 C | 21.3 sec | OK |
| CP at CP1 | amx0 | AMX0 Temp | 25.98 C | 21.3 sec | OK |
| CP at CP1 | amx1 | AMX1 Temp | 18.71 C | 21.3 sec | OK |
| CP at CP1 | rmx | RMX Temp | 21.00 C | 21.3 sec | OK |
| CP at CP1 | darb | DARB Temp | 31.18 C | 21.3 sec | OK |
| EXB at EX2 | max1617 AMB Temp | 26.00 C | 59.3 sec | OK |
| EXB at EX2 | max1617 AMB Temp | 25.00 C | 59.3 sec | OK |
EXB at EX2  sbbc  SBBC Temp  33.83  C  59.3 sec OK
EXB at EX2  axq  AXQ Temp  23.75  C  59.3 sec OK
EXB at EX2  sdim  SDIM Temp  20.46  C  59.3 sec OK
EXB at EX2  sdise  SDISE Temp  21.85  C  59.3 sec OK
EXB at EX2  pcf8591  1.5 VDC  1.51  V  56.6 sec OK
EXB at EX2  pcf8591  3.3 VDC  3.26  V  56.6 sec OK
EXB at EX2  pcf8591  2.5 VDC  2.47  V  56.6 sec OK
EXB at EX2  pcf8591  3.3 VDC HK  3.24  V  56.6 sec OK
CPU at SB2  max1617  PROC 0 Temp  42.00  C  9.6 sec OK
CPU at SB2  max1617  PROC 1 Temp  0.00  C  9.6 sec OK
CPU at SB2  max1617  PROC 2 Temp  0.00  C  9.6 sec OK
CPU at SB2  max1617  PROC 3 Temp  0.00  C  9.6 sec OK
CPU at SB2  sdc  SDC Temp  57.83  C  9.6 sec OK
CPU at SB2  ar  AR Temp  49.16  C  9.6 sec OK
CPU at SB2  dx0  DX0 Temp  50.49  C  9.6 sec OK
CPU at SB2  dx1  DX1 Temp  48.49  C  9.6 sec OK
CPU at SB2  dx2  DX2 Temp  46.50  C  9.6 sec OK
CPU at SB2  dx3  DX3 Temp  43.83  C  9.6 sec OK
CPU at SB2  sbbc 0  SBBC 0 Temp  45.16  C  9.6 sec OK
CPU at SB2  sbbc 1  SBBC 1 Temp  47.16  C  9.6 sec OK
CPU at SB2  pcf8591  1.5 VDC  1.51  V  57.2 sec OK
CPU at SB2  pcf8591  3.3 VDC  3.33  V  57.2 sec OK
CPU at SB2  pcf8591  Core 0 Volt  1.73  V  57.2 sec OK
CPU at SB2  pcf8591  Core 1 Volt  1.14  V  57.2 sec HIGH_MAX
CPU at SB2  pcf8591  Core 2 Volt  1.12  V  57.2 sec HIGH_MAX
CPU at SB2  pcf8591  Core 3 Volt  1.13  V  57.2 sec LOW_MIN
HPCI at IO1  pcf8591  PS0 Temp  48.10  C  48.7 sec OK
HPCI at IO1  pcf8591  PS1 Temp  31.97  C  48.7 sec OK
HPCI at IO1  sdc  SDC0 Temp  67.82  C  48.7 sec OK
HPCI at IO1  ar  AR0 Temp  61.82  C  48.7 sec OK
HPCI at IO1  dx0  DX0 Temp  57.16  C  48.7 sec OK
HPCI at IO1  dx1  DX1 Temp  47.83  C  48.7 sec OK
HPCI at IO1  sbbc  SBBC Temp  37.16  C  48.7 sec OK
HPCI at IO1  max1617a  IOA 0 Temp  52.00  C  48.7 sec OK
HPCI at IO1  max1617a  IOA 1 Temp  43.00  C  48.7 sec OK
HPCI at IO1  pcf8591  1.5 VDC  1.52  V  23.3 sec OK
HPCI at IO1  pcf8591  3.3 VDC  3.28  V  23.3 sec OK
HPCI at IO1  pcf8591  5.0 VDC  5.01  V  23.3 sec OK
HPCI at IO1  pcf8591  +12.0 VDC  12.03  V  23.3 sec OK
HPCI at IO1  pcf8591  -12.0 VDC  -12.01  V  23.3 sec OK
HPCI at IO1  pcf8591  3.3 VDC HK  3.28  V  23.3 sec OK
HPCI at IO1  pcf8591  1.5 CVT0 VDC  1.88  V  23.3 sec OK
HPCI at IO1  pcf8591  1.5 CVT1 VDC  1.74  V  23.3 sec OK
HPCI at IO1  pcf8591  3.3 V_PS0  10.25  A  23.3 sec OK
HPCI at IO1  pcf8591  3.3 V_PS1  10.40  A  23.3 sec OK
HPCI at IO1  pcf8591  5.0 V_PS0  4.02  A  23.3 sec OK
HPCI at IO1  pcf8591  5.0 V_PS1  4.15  A  23.3 sec OK
WPCI at IO8  max1617a  IOA0 Temp  46.00  C  39.9 sec OK
WPCI at IO8  dx0  DX0 Temp  61.16  C  39.9 sec OK
WPCI at IO8  dx1  DX1 Temp  56.49  C  39.9 sec OK
WPCI at IO8  sdc  SDC Temp  67.16  C  39.9 sec OK
WPCI at IO8  sbbc  SBBC Temp  41.16  C  39.9 sec OK
WPCI at IO8  ar  AR Temp  65.82  C  39.9 sec OK
WPCI at IO8  wci  WCI0 Temp  9.65  C  39.9 sec OK
WPCI at IO8  wci  WCI1 Temp  7.71  C  39.9 sec OK

Last modified 20 November 2001
showenvironment(1M) Maintenance Commands

<table>
<thead>
<tr>
<th>Component</th>
<th>Measurement</th>
<th>Value</th>
<th>Time</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>WPCI at IO8</td>
<td>pcf8591 VDC+12</td>
<td>11.95 V</td>
<td>26.2 sec</td>
<td>OK</td>
</tr>
<tr>
<td>WPCI at IO8</td>
<td>pcf8591 VDC-12</td>
<td>-12.01 V</td>
<td>26.2 sec</td>
<td>OK</td>
</tr>
<tr>
<td>WPCI at IO8</td>
<td>pcf8591 HK 3.3</td>
<td>3.26 V</td>
<td>26.2 sec</td>
<td>OK</td>
</tr>
<tr>
<td>WPCI at IO8</td>
<td>pcf8591 VDC 3.3</td>
<td>3.28 V</td>
<td>26.2 sec</td>
<td>OK</td>
</tr>
<tr>
<td>WPCI at IO8</td>
<td>pcf8591 VDC 1.5</td>
<td>1.48 V</td>
<td>26.2 sec</td>
<td>OK</td>
</tr>
<tr>
<td>WPCI at IO8</td>
<td>pcf8591 VDC 2.5</td>
<td>2.49 V</td>
<td>26.2 sec</td>
<td>OK</td>
</tr>
<tr>
<td>WPCI at IO8</td>
<td>pcf8591 VDC 5.0</td>
<td>5.04 V</td>
<td>26.2 sec</td>
<td>OK</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component</th>
<th>Measurement</th>
<th>Value</th>
<th>Time</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schizo0.0</td>
<td>max1617a Slot 0</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>Schizo0.1</td>
<td>max1617a Slot 0</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>EXB at EX4</td>
<td>pcf8591 VDC 1.5</td>
<td>1.51 V</td>
<td>56.0 sec</td>
<td>OK</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>PROC 0 Temp</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>PROC 1 Temp</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>PROC 2 Temp</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>PROC 3 Temp</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>SBB 0 Temp</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>SBB 1 Temp</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>VDC HK 3.3</td>
<td>3.26 V</td>
<td>56.6 sec</td>
<td>OK</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>Core 0 Volt</td>
<td>-1.00 V</td>
<td>56.6 sec</td>
<td>HIGH_MAX</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>Core 1 Volt</td>
<td>1.12 V</td>
<td>56.6 sec</td>
<td>HIGH_MAX</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>Core 2 Volt</td>
<td>1.70 V</td>
<td>56.6 sec</td>
<td>OK</td>
</tr>
<tr>
<td>CPU at SB4</td>
<td>Core 3 Volt</td>
<td>1.13 V</td>
<td>56.6 sec</td>
<td>HIGH_MAX</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component</th>
<th>Measurement</th>
<th>Value</th>
<th>Time</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schizo1.0</td>
<td>max1617a Slot 0</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>Schizo1.1</td>
<td>max1617a Slot 1</td>
<td>N/A</td>
<td>N/A</td>
<td>PRESENCE</td>
</tr>
<tr>
<td>EXB at EX1</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>CPU at SB6</td>
<td>PROC 0 Temp</td>
<td>43.00 C</td>
<td>5.1 sec</td>
<td>OK</td>
</tr>
<tr>
<td>CPU at SB6</td>
<td>PROC 1 Temp</td>
<td>0.00 C</td>
<td>5.1 sec</td>
<td>OK</td>
</tr>
<tr>
<td>CPU at SB6</td>
<td>PROC 2 Temp</td>
<td>0.00 C</td>
<td>5.1 sec</td>
<td>OK</td>
</tr>
</tbody>
</table>

190 SMS 1.2 Last modified 20 November 2001
### CPU at SB6
- `max1617`: PROC 3 Temp: 0.00 C, 5.1 sec, OK
- `sdc`: SDC Temp: 62.49 C, 5.1 sec, OK
- `ar`: AR Temp: 55.16 C, 5.1 sec, OK
- `dx0`: DX0 Temp: 57.16 C, 5.1 sec, OK
- `dx1`: DX1 Temp: 55.16 C, 5.1 sec, OK
- `dx2`: DX2 Temp: 55.83 C, 5.1 sec, OK
- `dx3`: DX3 Temp: 53.83 C, 5.1 sec, OK
- `sbbc 0`: SBBC 0 Temp: 51.83 C, 5.1 sec, OK
- `sbbc 1`: SBBC 1 Temp: 49.16 C, 5.1 sec, OK
- `pcf8591 1.5 VDC`: 1.51 V, 56.0 sec, OK
- `pcf8591 3.3 VDC`: 3.30 V, 56.0 sec, OK
- `pcf8591 Core 0 Volt`: 1.72 V, 56.0 sec, OK
- `pcf8591 Core 1 Volt`: 1.13 V, 56.0 sec, HIGH_MAX
- `pcf8591 Core 2 Volt`: 1.14 V, 56.0 sec, HIGH_MAX
- `pcf8591 Core 3 Volt`: 1.13 V, 56.0 sec, LOW_MIN

### EXB at EX12
- `max1617`: AMB Top Temp: 24.00 C, 27.1 sec, OK
- `max1617`: AMB Bot Temp: 24.00 C, 27.1 sec, OK
- `sbbc`: SBBC Temp: 35.16 C, 27.1 sec, OK
- `axq`: AXQ Temp: 27.01 C, 27.1 sec, OK
- `sdim`: SDIM Temp: 24.62 C, 27.1 sec, OK
- `sdise`: SDISE Temp: 24.59 C, 27.1 sec, OK
- `sdisc`: SDISC Temp: 27.48 C, 27.1 sec, OK
- `pcf8591 1.5 VDC`: 1.51 V, 55.3 sec, OK
- `pcf8591 3.3 VDC`: 3.28 V, 55.3 sec, OK
- `pcf8591 2.5 VDC`: 2.47 V, 55.3 sec, OK
- `pcf8591 3.3 VDC HK`: 3.26 V, 55.3 sec, OK

### FANTRAY
- **FT0**: ON, HIGH, OK, OK, OK, OK, OK, OK, OK, OK, OK, OK
- **FT1**: ON, HIGH, OK, OK, OK, OK, OK, OK, OK, OK
- **FT2**: ON, HIGH, OK, OK, OK, OK, OK, OK, OK, OK
- **FT3**: ON, HIGH, OK, OK, OK, OK, OK, OK, OK, OK
- **FT4**: ON, HIGH, OK, OK, OK, OK, OK, OK, OK, OK
- **FT6**: ON, HIGH, OK, OK, OK, OK, OK, OK, OK, OK
- **FT7**: ON, HIGH, OK, OK, OK, OK, OK, OK, OK, OK

### POWER UNIT
- **PS0**: FAIL, FAIL, FAIL, ON, ON, OK, OK
- **PS1**: FAIL, OK, OK, ON, ON, OK, OK
- **PS2**: OK, OK, OK, ON, ON, OK, OK
- **PS4**: OK, OK, ON, ON, OK, OK
- **PS5**: OK, OK, ON, ON, OK, OK

### POWER
- **PS0**: Current0 0.39 A, N/A
- **PS0**: Current1 0.39 A, N/A
- **PS0**: 48VDC 0.39 V, N/A
- **PS1**: Current0 8.36 A, N/A
- **PS1**: Current1 5.97 A, N/A
- **PS1**: 48VDC 48.60 V, N/A

---

Last modified 20 November 2001

SMS 1.2

191
EXAMPLE 2  Reporting Temperature on Domain A

This example assumes that domain a contains MCPUs at IO6 and IO2.

```
sc0:sms-user:~> showenvironment -p temps -d a

LOCATION DEVICE SENSOR       VALUE UNIT AGE STATUS
--------- -------- ---------- --------- ------ ------ -----
MCPU at I06 max1617 PROC 1 Temp 35.00 C 8.0 sec OK
... ... ... ... ...
MCPU at I02 dx0 DX0 Temp 36.50 C 8.0 sec OK
... ... ... ... ...
```

EXIT STATUS  The following exit values are returned:

0  Successful completion.

1  An invalid domain used.

2  An invalid command line option used.

3  Invalid permission.

4  An internal error occurred.

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSOp</td>
</tr>
</tbody>
</table>
SEE ALSO

addtag(1M)
showfailover – manage or display system controller (SC) failover status

showfailover [-r] [-v]
showfailover [-h]

DESCRIPTION
showfailover(1M) provides the ability 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: <state>

The failover mechanisms can be in one of three states: ACTIVE, DISABLED, and FAILED. See the EXTENDED DESCRIPTION below.

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 described as follows:

ACTIVE Identifies the failover mechanism as being enabled and functioning normally.

DISABLED Identifies that the failover mechanism has been disabled due to the occurrence of a failover or an operator request (for example, setfailover off).
Identifies that the failover mechanism has detected a failure that prevents a failover from being possible.

In addition, `showfailover` displays the state of each of the network interface links monitored by the failover processes. The display format is:

```
network if device name: [GOOD|FAILED]
```

A failure string is returned describing the failure condition. Each failure string has a code associated with it. The following codes and associated failure strings are defined:

<table>
<thead>
<tr>
<th>String</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>No Failure.</td>
</tr>
<tr>
<td>M-SC/S-SC EXT NET</td>
<td>The main and spare SC's external network interfaces have failed.</td>
</tr>
<tr>
<td>S-SC CONSOLE BUS</td>
<td>A fault has been detected on the spare SC's console bus path.</td>
</tr>
<tr>
<td>S-SC LOC CLK</td>
<td>The spare SC's local clock has failed.</td>
</tr>
<tr>
<td>S-SC CLK NOT PHASE LOCKED</td>
<td>The spare SC's clock is not phase locked with the main SC.</td>
</tr>
<tr>
<td>S-SC DISK FULL</td>
<td>The spare SC's system is full.</td>
</tr>
<tr>
<td>S-SC IS DOWN</td>
<td>The spare SC is down and unresponsive.</td>
</tr>
<tr>
<td>S-SC MEM EXHAUSTED</td>
<td>The spare SC's memory/swap space has been exhausted.</td>
</tr>
<tr>
<td>S-SC SMS DAEMON</td>
<td>At least one SMS daemon could not be started/restarted on the spare SC.</td>
</tr>
<tr>
<td>No CSBS Powered on</td>
<td>At least one CSB must be powered on.</td>
</tr>
</tbody>
</table>
showfailover(1M)

Group Privileges
Required

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

Refer to Chapter 2 in the System Management Services (SMS) 1.2
Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  Failover Status Shows Everything is OK

```
sc0> showfailover
SC Failover: ACTIVE
hme0: GOOD
hme1: GOOD
hme2: GOOD
```

EXAMPLE 2  The Spare SC System is Full

```
sc0> showfailover
SC Failover: FAILED
S-SC DISK FULL
hme0: GOOD
hme1: GOOD
hme2: GOOD
```

EXAMPLE 3  Displays the SC Role

```
sc0> showfailover -r
SC: SPARE
```

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:

196       SMS 1.2       Last modified 01 October 2001
Maintenance Commands

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

setfailover(1M)
NAME  
showkeyswitch – display the position of the virtual keyswitch

SYNOPSIS  
showkeyswitch –d domain_id | domain_tag [−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  
−d domain_id  ID for a domain. Valid domain_ids are ’A’...'R' and are case insensitive.
−d 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 or platform service privileges, or domain administrators or configurators privileges for the specified domain to run this command.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES  
EXAMPLE 1  Keyswitch Status for Domain A

sc0sms-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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**SEE ALSO**

addtag(1M), setkeysweak(1M), pcd(1M)
NAME
showlogs – display message log files

SYNOPSIS
showlogs [−F] [−f filename] [−d domain_id | domain_tag] [−p m | c | s] [−v ]

DESCRIPTION
showlogs(1M) displays platform or domain log files. The default is the platform message log. You must have platform group privileges to run the default, otherwise you will receive an error message. Depending on your privileges, you can display the message logs, console logs, or syslog for the platform or a specified domain.

OPTIONS
−F
Outputs only lines which have been appended to the log file since the showlogs command was executed. Similar to the ‘tail –f’ command. Output will continue until interrupted by Control −c.

−d domain_id | domain_tag
Outputs the message log file for the specified domain instead of the platform log. You must have domain privileges to use this option.

−f filename
Places the output of the showlogs command into a specified file.

−h
Help. Displays usage descriptions.

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

−p m | c | s
Specifies display of either the platform (m)essage log or domain (c)onsole log or domain (s)yslog.

m — Displaying the platform message log requires platform group privileges. This is the showlogs default.

c — Displaying the domain console log requires the −d option and domain privileges for that domain.

s — Displaying the domain syslog requires domain privileges for that domain. Syslogs forwarded to the system controller (SC) from
non-domain systems are stored in /var/opt/SUNWSMS/adm/anonymous.

−v
Verboso. Displays all available command information.

EXTENDED DESCRIPTION

Group Privileges
Required
If you have platform administrator, operator, or service privileges, you can display the platform messages log file.
If you have domain administrator/configurator privileges, you can display only those log files for domains for which you have privileges.
Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  Output 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  Output 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  Output 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 850960648900 INFO Observers... c 193] DOMAIN_UP A event has been sent to DXS, rc = 0....
```

EXAMPLE 4  Output 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 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  Output 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 7 13:51:49 agent [received software termination signal]
Sep 7 13:51:49 sms2 agent[6629]: [ID 985882 daemon.alert] syslog
Sep 7 13:51:49 agent *** terminating execution ***
Sep 7 13:51:50 sms2 platform[22481]: [ID 345917 daemon.alert] syslog
Sep 7 13:51:50 platform *** terminating execution ***
Sep 7 14:49:07 sms2 platform[4309]: [ID 745556 daemon.alert] syslog
Sep 7 14:49:07 platform general parsing error
Sep 7 14:49:07 sms2 platform[4309]: [ID 344248 daemon.alert] syslog
Sep 7 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 7 14:49:07 platform couldn't load file
...
```

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..

ATTRIBUTES

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

tail(l)
NAME
showobpparams – display OpenBoot PROM bring up parameters for a domain

SYNOPSIS
showobpparams
  −d domain_id | domain_tag [−v ]

showobpparams −h

DESCRIPTION
showobpparams(1M) allows a domain administrator to display the virtual
NVRAM and REBOOT parameters passed to OpenBoot PROM by
setkeys(c)h(1M). The −d option with domain_id or domain_tag is required.

OPTIONS
The following options are supported:

−d domain_id ID for a domain. Valid domain_ids are ‘A’...’R’ and are case
insensitive.

−d domain_tag Name assigned to a domain using addt(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 Chapter 2 in the System Management Services (SMS) 1.2
Administrator Guide for more information.

EXAMPLES
EXAMPLE 1 Displaying OpenBoot PROM Parameters for Domain A

sc0sms-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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

addtag(1M), setkeysswitch(1M), setobpparams(1M)
NAME  showplatform – display the board available component list and domain state for each of the domains

SYNOPSIS  showplatform [−d domain_id | domain_tag] [−p domains | available | ethernet ]
            [−v ]

            showplatform −h

DESCRIPTION  Show the available component list, domain state and Ethernet address for domains. If a domain_id | domain_tag is specified, only the information for that domain is displayed. If no domain and −p are specified, the available component list, domain states and ethernet addresses for all domains for which you have privileges are displayed.

OPTIONS

−d domain_id  ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.

−d domain_tag  Domain 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  Display specific reports.

    Valid arguments for −p 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.

−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 or for Ethernet addresses, it indicates the domain idprom image file does not exist. You need to 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 Exit OBP — The domain OpenBoot PROM exited.
- OBP Failed — The domain OpenBoot PROM 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 running, boot attempt failed.
- Loading Solaris Failed — OpenBoot PROM running, loading attempt 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, panic flow has started.
- Solaris Panic Debug — Solaris software panicked, and is entering debugger mode.
- Solaris Panic Continue — Exited debugger mode and continuing panic flow.
- Solaris Panic Dump — Panic dump has started.
- Solaris Halt — Solaris software is halted.
- Solaris Panic Exit — Solaris software 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.
- 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 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 Chapter 2 in the *System Management Services (SMS) 1.2 Administrator Guide* for more information.

**EXAMPLES**

**EXAMPLE 1**  
Show the Available Component List and Domain State Information for All Domains

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.

```
sc0:sms-user:~ > showplatform

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 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 Configurations:
showplatform(1M) Maintenance Commands

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  dmnR  sun15-r  Running Solaris

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:b7:3d: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

EXAMPLE 2  Show Available Component List and Domain State for Domain engB

sc0@sms-9net:~$ showplatform -d engB

Available Component List for Domains:

Available for domain engB:
SB4 SB5 SB6
I04 I05

Domain Configurations:

---------------------------
EXAMPLE 3  Showplatform for Domain Administrators

The following example shows 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, E and dmnj.

```
sc0:~$>
```

```
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 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
```

```
Domain Ethernet Addresses:
===========================

Domain ID Domain Tag  Ethernet Address
B  engB  8:0:20:b4:30:8c
```

EXAMPLE 4  Show Available Component List for Domain engB

```
sc0:~$>
```

```
showplatform -d engB -p available
```

Available Component List for Domains:

```
Last modified 05 October 2001  SMS 1.2  211
EXAMPLE 5  Show Domain Status for Domain engB

    sc0:$SMN-USER:/> showplatform -d engB -p domains

Domain Configurations:

<table>
<thead>
<tr>
<th>DomainID</th>
<th>Domain Tag</th>
<th>Solaris Nodename</th>
<th>Domain Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>engB</td>
<td>sun15-b</td>
<td>Keyswitch Standby</td>
</tr>
</tbody>
</table>

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 communicating with the platform configuration daemon (pcd(1M)).

6  An error occurred communicating with the hardware access daemon (hwad(1M)).

7  An error occurred communicating with the task management daemon (tmd(1M)).

8  An internal error occurred.

ATTRIBUTES

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

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNW$SMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

addtag(1M), hwad(1M), pcd(1M), setupplatform(1M), tmd(1M)
NAME showxirstate – display CPU dump information after sending a reset pulse to the processors

SYNOPSIS showxirstate –d domain_id|domain_tag |–f filename [–v ]  

showxirstate [–h ]

DESCRIPTION showxirstate(1M) displays CPU dump information after sending a reset pulse 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.

If domain_id | domain_tag or filename is not specified, showxirstate returns an error.

OPTIONS The following options are supported:

−d domain_id ID for a domain. Valid domain_ids are ‘A’...’R’ and are case insensitive.

−d domain_tag Name assigned to a domain using addtag(1M).

−f filename Name of the file containing a previously generated xir_dump. You must provide the absolute path to the file. 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 Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

Last modified 01 October 2001
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
y    : 00000000.00000000
afs  : 00000000.00000000  aff : 00000000.00000000
pcontext: 00000000.00000000 scontext: 00000000.00000000
pil  : 0x0
p    : 00000000.00000000
afsr : 00000000.00000000 afar : 00000000.00000000
pcr  : 00000000.00000000
gsr  : 00000000.00000000
softint : 0x0000

pawatch: 00000000.00000000
va_watch: 00000000.00000000

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 0x00000000 00000000.00000000 00000000.00000000
0x00 0x00000000 00000000.00000000 00000000.00000000
0x00 0x00000000 00000000.00000000 00000000.00000000
0x00 0x00000000 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
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

Last modified 01 October 2001
SMS 1.2 215
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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

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. In order to create a complete and accurate backup, turn off SMS before running smsbackup. For information on manually starting and stopping SMS refer to the System Management Services (SMS) 1.2 Installation Guide and Release Notes.

Whenever changes are made to the SMS environment, for example by shutting down a domain, you must run smsbackup again in order 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.

OPTIONS

-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 on as a UFS filesystem. Specifying a TMPFS file system, such as /tmp, will cause smsbackup to fail. If you are not certain that your directory_name is mounted as a UFS filesystem, type:
/usr/bin/df -F ufs directory_name

A UFS file system will return directory information. Any other type of file system will return a warning.

EXTENDED DESCRIPTION

Group Privileges Required

You must have superuser privileges to run this command.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  Backing Up SMS to /var/opt/SUNWSMS/bkup

sc0:~user$ smsbackup /var/opt/SUNWSMS/bkup

EXAMPLE 2  Backing Up SMS to a Tape Device 0

sc0:~user$ smsbackup /dev/rmt/0

EXAMPLE 3  Backing Up SMS to a TMPFS System

sc0:~user$ smsbackup /tmp
ERROR: smsbackup fails to backup to /tmp, a TMPFS file system. Please specify a directory that is mounted on a UFS filesystem.
ABORT:

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>
The following file is used by this command:
/var/sadm/system/logs/smsbackup  smsbackup log file

SEE ALSO

smsrestore(1M)
NAME  
  smsconfig – configures the SMS environment

SYNOPSIS
  
  smsconfig
  
  smsconfig -m
  
  smsconfig -m I1 [ domain_id | sc]
  
  smsconfig -m I2 [sc0 | sc1]
  
  smsconfig -m L [sc]
  
  smsconfig -g
  
  smsconfig -a -u username -G admn | oper | svc platform
  
  smsconfig -r -u username -G admn | rcfg domain_id
  
  smsconfig -l domain_id | platform
  
  smsconfig -h

DESCRIPTION
  
  smsconfig(1M) configures and modifies the 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 all three internal enterprise networks.

  Note - Once you have configured or changed the configuration of the MAN network you must reboot the 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_ids 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 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 allows you to customize those groups using the -g option. For more information refer to the System Management Services (SMS) 1.1 Installation Guide and Release Notes.
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. User access will be compromised.

### OPTIONS

- **−a**
  Adds a user to an SMS group and provides read,
  write and execute access for a domain or the
  platform directories. You must specify a valid
  *username*, SMS group and if applicable, a
  *domain_id*

- **−G admn | rcfg**
  Indicates an SMS domain administrator or
  reconfigurator. All groups are case insensitive.

- **−G admn | oper | svc**
  Indicates an SMS platform administrator,
  operator or service personnel. All groups are case
  insensitive.

- **−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 case insensitive. A
  domain can be excluded from the I1 network
  configuration by using the word *NONE* as the
  *net_id*. This applies to the I1 network only.

- **−m I2**
  Configures all interfaces for enterprise network
  I2. Network designation is case insensitive.
−m L

Configures all interfaces for the external community network. Network designation is case insensitive.

−r

Removes a user from an SMS group and denies read, write and execute access for a domain or the platform directories. You must specify a valid username, SMS group and if applicable, a domain_id.

−u username

Indicates user login name.

OPERANDS

The following operands are supported:

domain_id

ID for a domain. Valid domain_ids are 'A'...'R' and are case insensitive.

platform

Specifies the Sun Fire 15K platform and platform specific directories.

SC, SC0, SC1

Interface designation for the Sun Fire 15K SC. Interface designations are case insensitive.

EXTENDED DESCRIPTION

Group Privileges

Required

You must have superuser privileges to run this command.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  Initial Setup

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. An IPv6 network example differs slightly. There will be no prompts for netmasks and /etc/ipnodes will be modified in addition to /etc/hosts.

IP addresses on the external network for failover, hme0 and eri1 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. A domain can be excluded from the I1 network configuration by using the word NONE as the net_id. For more information refer to the System Management Services (SMS) 1.1 Installation Guide and Release Notes.
Once you have configured the MAN network, you must reboot the SC.

```bash
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**

Enter NICs associated with community C1 [hme0 eri1]: [Return]

Enter Logical/Floating IP hostname for community C1 [sun15-sc-C1]: [Return]

Enter IPMP IP address for sun15-sc-C1: **10.1.1.50**

Enter Netmask for community C1: **255.255.255.0**

Enter IPMP hostname for community C1 failover address [sun15-sc0-C1-failover]: [Return]

Enter IPMP IP address for sun15-sc0-C1-failover: **10.1.1.51**

Enter IPMP hostname for hme0 [sun15-sc0-hme0]: [Return]

Enter IPMP IP address for sun15-sc0-hme0: **10.1.1.52**

Enter IPMP hostname for eri1 [sun15-sc0-eri1]: [Return]

Enter IPMP IP address for sun15-sc0-eri1: **10.1.1.53**

<table>
<thead>
<tr>
<th>Hostname</th>
<th>IP Address (platform=sun15)</th>
</tr>
</thead>
<tbody>
<tr>
<td>sun15-sc-C1</td>
<td>10.1.1.50</td>
</tr>
<tr>
<td>sun15-sc0-C1-failover</td>
<td>10.1.1.51</td>
</tr>
<tr>
<td>sun15-sc0-hme0</td>
<td>10.1.1.52</td>
</tr>
<tr>
<td>sun15-sc0-eri1</td>
<td>10.1.1.53</td>
</tr>
</tbody>
</table>

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.s? [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]

<table>
<thead>
<tr>
<th>Hostname</th>
<th>IP Address (platform=sun15)</th>
</tr>
</thead>
<tbody>
<tr>
<td>netmask-i1</td>
<td>255.255.255.224</td>
</tr>
<tr>
<td>sun15-sc-i1</td>
<td>10.2.1.1</td>
</tr>
<tr>
<td>sun15-a</td>
<td>10.2.1.2</td>
</tr>
<tr>
<td>sun15-b</td>
<td>10.2.1.3</td>
</tr>
<tr>
<td>sun15-c</td>
<td>10.2.1.4</td>
</tr>
<tr>
<td>sun15-d</td>
<td>10.2.1.5</td>
</tr>
</tbody>
</table>
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: 10.2.1.20 sun15-sc0-11 #smsconfig-entry#
ADD: 10.2.1.21 sun15-sc1-11 #smsconfig-entry#
ADD: 10.2.1.22 sun15-sc0-C1-failover #smsconfig-entry#
ADD: 10.2.1.23 sun15-sc1-C1-failover #smsconfig-entry#
**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-scl-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-scl-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.
```
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

```
sc0: # smsconfig -m I1 D
Enter the MAN hostname for DB-I1 [ sun15-b ]: domainB-il
I could not automatically determine the IP address of domainB-il.

Please enter the IP address of domainB-il: NONE

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-il IP Address: NONE

Do you want to accept these settings? [y,n] y
Creating ~/.rhosts to facilitate file propagation ... done.

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: NONE domainB-il #smsconfig-entry#

Update the hosts file, "/etc/hosts", with these changes? [y,n] y
Hosts file "/etc/hosts" has been updated.
```

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 [platadm]? 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 [dmbadmn]? [Return]
Enter the name of the Domain C Administrator group [dmncadm]? [Return]
Enter the name of the Domain D Administrator group [dmdadm]? [Return]
Enter the name of the Domain E Administrator group [dmnedadm]? [Return]
Enter the name of the Domain F Administrator group [dmnfadm]? [Return]
Enter the name of the Domain G Administrator group [dmngadm]? [Return]
Enter the name of the Domain H Administrator group [dmnhadm]? [Return]
Enter the name of the Domain I Administrator group [dmniadm]? [Return]
Enter the name of the Domain J Administrator group [dmnjadm]? [Return]
Enter the name of the Domain K Administrator group [dmnkadm]? [Return]
Enter the name of the Domain L Administrator group [dmnladm]? [Return]
Enter the name of the Domain M Administrator group [dmnmadm]? [Return]
Enter the name of the Domain N Administrator group [dmnnadm]? [Return]
Enter the name of the Domain O Administrator group [dmnoadm]? [Return]
Enter the name of the Domain P Administrator group [dmnopadm]? [Return]
Enter the name of the Domain Q Administrator group [dmnqadm]? [Return]
Enter the name of the Domain R Administrator group [dmnradm]? [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 [dmbrcfg]? [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.
EXAMPLE 6  Adding a User to the Domain Administrator Group and Configuring Access to the Domain B Directories

You must specify a valid username and valid SMS group and domain.

    sc0: # smsconfig -a -u fdjones -G admn B
    fdjones has been added to the dmnnAdmn group.
    All privileges to domain B have been applied.

EXAMPLE 7  Adding a User to the Domain Configurator Group and Configuring Access to the Domain C Directories

You must specify a valid username and valid SMS group and domain.

    sc0: # smsconfig -a -u fdjones -G rcfg C
    fdjones has been added to the dmnnRcfg group.
    All privileges to domain C have been applied.

EXAMPLE 8  Configuring Access to the Platform Directories

You must specify a valid username 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 9  Displaying Users with Access to the Domain C Directories

    sc0: # smsconfig -l C
    fdjones
    shea
EXAMPLE 10  Displaying Users with Access to the Platform Directories

sc0: # smsconfig -l platform
fdjones
jtd

EXAMPLE 11  Removing User Access to the Domain C Directories

You must specify a valid username and valid SMS group. If a user belongs to more than one group with access to a domain, they 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 12  Configuring Using an Invalid Groupname

You must specify a valid SMS group.

sc0: # smsconfig -a -u fdjones -G staff D
ERROR: group staff does not exist
ABORTING.

EXAMPLE 13  Mixing Groups and Designations.

You must specify groupnames 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
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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSr</td>
</tr>
</tbody>
</table>

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.

SEE ALSO

mand(1M)

NOTES

The user is required to enter several items of key network information. To facilitate the gathering of this information beforehand, refer to the Sun Fire 15K System Site Planning Guide for setup worksheets.
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 which 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 will fail and the session will most likely hang. To exit, type: ~.

When you finish, there are several ways to end the session depending on whether you logged into the local SC using telnet or rlogin. See the EXTENDED DESCRIPTION section below.

OPTIONS

- `-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 are telnetted in to the local SC this will disconnect the tip session and you will remain logged in to the local SC.
If you `rlogged` in to the local SC, this will disconnect the `tip` session and also disconnect your `rlogin` session.

**Note** - The tilde will not echo to the screen until after the period is pressed.

- `~.`. Disconnect `tip` session.
  - `~.` only works with with `rlogin`. If you are `telnetted` in to the local SC you will receive the error message: `~:` Command not found

  If you are `rlogged` in to the local SC this will disconnect the `tip` session and you will remain logged in to the local SC.

**Note** - The first tilde will not echo to the screen. The second will not echo until after the period is pressed.

`rlogin` also processes tilde-escape sequences whenever a tilde is seen 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 `^j`, then `stty sane`, then `^j`.

**Group Privileges Required**

You must have platform administrator privileges to run this command.

Refer to Chapter 2 in the *System Management Services (SMS) 1.2 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

rlogin(1M), rlogin(1M), tip(1M)
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 wish, afterwards. For information on manually starting and stopping SMS refer to the System Management Services (SMS) 1.2 Installation Guide and Release Notes.

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
−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 will receive an error.

EXTENDED DESCRIPTION
Group Privileges Required
You must have superuser privileges to run this command.

Last modified 04 October 2001

SMS 1.2 235
Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

**EXAMPLES**

**EXAMPLE 1**  Restoring SMS

```
sc# smsrestore sms_backup.1.0.cpio
```

**EXAMPLE 2**  Restoring SMS from Tape Device 0

```
sc# smsrestore /dev/rmt/0/sms_backup.1.0.cpio
```

**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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

**FILES**

The following file is used by this command:

```
/var/sadm/system/logs/smsrestore  smsrestore log file
```

**SEE ALSO**

smsbackup(1M)
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.

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 needed to activate 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:

- Turn off failover and stop SMS before running smsrestore.
- 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 now.

- Otherwise, you can start SMS and turn on failover. For information on manually starting and stopping SMS refer to the System Management Services (SMS) 1.2 Installation Guide and Release Notes.

If any errors occur, smsversion writes error messages to /var/sadm/system/logs/smsversion.

OPTIONS

-h

Help. Displays usage descriptions.

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

The following operands are supported:

version_number  Release number of the target SMS version.

EXTENDED DESCRIPTION

Group Privileges

Required

You must have superuser privileges to run this command.

Refer to Chapter 2 in the System Management Services (SMS) 1.2 Administrator Guide for more information.

EXAMPLES

EXAMPLE 1  One Version of SMS Installed

Displays the active version and exits when only one version of SMS is installed.

sc# smsversion -t
1.1

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.

SMS must be stopped prior to switching versions.

sc# smsversion
smsversion: Active SMS version < 1.1 >
smsversion: SMS version 1.1 installed
smsversion: SMS version 1.2 installed
Please select from one of the following installed SMS versions.
1) 1.1
2) 1.2
3) Exit
Select version: 2
You have selected SMS Version 1.2
Is this correct? [y,n] y
smsversion: Upgrading SMS from <1.1> to <1.2>.
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.1.1.cpio
smsversion: Switching to target version <1.2>. 
EXAMPLE 3  Downgrading SMS Versions

Use of the command-line argument to downgrade SMS versions.

sc# smsversion 1.1
smsversion: Active SMS version < 1.2 >
You have requested SMS Version 1.1

Is this correct? [y,n]  y
smsversion: Downgrading SMS from <1.2> to <1.1>.  
smsversion: SMS version 1.1 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.1.2.cpio  
smsversion: Switching to target version <1.1>.  
smsversion: New Version <1.1> Active  
smsversion: Active SMS version < 1.1 >
To restore previous the SMS configuration setting type:  
smsrestore /var/tmp/sms_backup.1.2.cpio

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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

FILES  The following file is used by this command:
/var/sadm/system/logs/smsversion smsversion log file
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

- 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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>
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

-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:

<table>
<thead>
<tr>
<th>ATTRIBUTE TYPE</th>
<th>ATTRIBUTE VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>SUNWSMSop</td>
</tr>
</tbody>
</table>

SEE ALSO

ssd(1M)