



Sun Management Center Change Manager 1.0 Reference Manual

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

Part No: 817-0576-10
October 2002

Copyright 2002 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

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

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

Sun, Sun Microsystems, the Sun logo, Solaris, Sun4U, and docs.sun.com are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

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

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

Copyright 2002 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés

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

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

Sun, Sun Microsystems, le logo Sun, Solaris, Sun4U, et docs.sun.com sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REpondre A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



021015 @ 4660



Contents

Preface 5

Change Manager System Administration Commands 7

bart(1MCM) 8

changemgr(1MCM) 13

cmgetprop(1MCM) 20

cminst(1MCM) 21

cmsetup(1MCM) 23

Change Manager File Formats 25

bart_manifest(4CM) 26

bart_rules(4CM) 29

cmssp(4CM) 33

ichange.cfg(4CM) 36

Preface

This reference manual contains the man pages for the Sun™ Management Center Change Manager product, henceforth referred to as Change Manager.

Who Should Use This Book

This book is intended for anyone responsible for performing one or more of these Change Manager operations:

- Installing the Change Manager software on the Change Manager server
- Managing deployment objects and audit objects in the Change Manager repository
- Managing hosts on the Change Manager server
- Creating the Solaris™ Flash archives for use with the Change Manager
- Deploying software stacks to managed hosts
- Auditing software on managed hosts

Related Books

- *Sun Management Center Change Manager 1.0 Release Notes*
- *Sun Management Center Change Manager 1.0 Administration Guide*
- *Solaris 9 Installation Guide*
- *Sun Management Center 3.0 Software Installation Guide*
- *Sun Management Center 3.0 Software User's Guide*
- *Sun Management Center 3.0 Configuration and Deployment Guide*

Accessing Sun Documentation Online

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

Ordering Sun Documentation

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

Change Manager System Administration Commands

bart(1MCM)

NAME	bart – basic audit reporting tool
SYNOPSIS	<pre> /usr/bin/bart create [-n] [-R root_directory] [-r rules_file -] /usr/bin/bart create [-n] [-R root_directory] -I [file_name ...] /usr/bin/bart compare [-i attribute] [-p] [-r rules_file -] control-manifest test-manifest </pre>
DESCRIPTION	<p>The bart(1MCM) command is a tool that performs a file-level check of the software contents of a system. Users can optionally specify the files to track and the types of discrepancies to flag by means of a rules file. See <code>bart_rules(4CM)</code>.</p> <p>The bart command performs two basic functions:</p> <p>bart create The manifest generator tool takes a file-level “snapshot” of a system. The output is a catalog of file attributes referred to as a “manifest.” See <code>bart_manifest(4CM)</code>.</p> <p>Users can specify the list of files to be cataloged in three ways. Use <code>bart create</code> with no options, specify the files by name on the command line, or create a rules file with directives that specify which the files to monitor. See <code>bart_rules(4CM)</code>.</p> <p>By default, the manifest generator catalogs all attributes of all files in the root (/) file system. File systems mounted on the root file system are cataloged only if they are of the same type as the root file system.</p> <p>For example, /, /usr, and /opt are separate UFS file systems. /usr and /opt are mounted on /. Therefore, all three file systems are cataloged. However, /tmp, also mounted on /, is not cataloged because it is a TMPFS file system. Mounted CD-ROMs are not cataloged since they are HSFS file systems.</p> <p>bart compare The report tool compares two manifests. The output is a list of per-file attribute discrepancies. These discrepancies are the differences between two manifests: a control manifest and a test manifest. A discrepancy is a change to any attribute for a given file cataloged by both manifests. Note that a new file or a deleted file in a manifest is reported as a discrepancy.</p> <p>The reporting mechanism provides two types of output: verbose and programmatic. Verbose output is localized and presented on multiple lines, while programmatic output is more easily parsable by other programs. See <code>OUTPUT</code>.</p> <p>By default, the report tool generates verbose output where all discrepancies are reported except for modified directory timestamps (the <code>dirmtime</code> attribute).</p>

Note – To ensure consistent and accurate comparison results, both *control-manifest* and *test-manifest* must be built with the same rules file.

Use the rules file to ignore specified files or subtrees when you generate a manifest or compare two manifests. Users can compare manifests from different perspectives by re-running the `bart compare` command with different rules files.

OPTIONS

The following options are supported:

- `-i attribute ...` Specifies the file attributes to be ignored globally. This option produces the same behavior as supplying the file attributes to a global `IGNORE` keyword in the rules file. See `bart_rules(4CM)`.
- `-I [file_name...]` Inputs list of files. The file list can be specified at the command line or read from standard input.
- `-n` Prevents computation of content signatures for all regular files in the file list.
- `-p` Displays manifest comparison output in “programmatic mode,” which is suitable for programmatic parsing. The output is not localized.
- `-r rules_file` Uses *rules_file* to specify which files and directories to catalog, and to define which file attribute discrepancies to flag. If *rules_file* is `-`, then the rules are read from standard input. See `bart_rules(4CM)` for the definition of the syntax.
- `-R root_directory` Specifies the root directory for the manifest. All paths specified by the rules, and all paths reported in the manifest, are relative to *root_directory*.

OPERANDS

The following operands are supported:

- control-manifest* Is the manifest created by `bart create` on the control system.
- test-manifest* Is the manifest created by `bart create` on the test system.

OUTPUT

The `bart create` and `bart compare` commands write output to standard output, and write error messages to standard error.

The `bart create` command generates a system manifest. See `bart_manifest(4CM)`.

When the `bart compare` command compares two system manifests, it generates a list of file differences. By default, the comparison output is localized. However, if the `-p` option is specified, the output is generated in a form that is suitable for programmatic manipulation.

Default Format

```
filename
  attribute    control:xxxx test:yyyy
```

bart(1MCM)

	<p><i>filename</i> Name of the file that differs between <i>control-manifest</i> and <i>test-manifest</i>. For file names that contain embedded whitespace or newline characters, see Quoting Syntax in <i>bart_manifest(4CM)</i>.</p> <p><i>attribute</i> The name of the file attribute that differs between the manifests that are compared. <i>xxxx</i> is the attribute value from <i>control-manifest</i>, and <i>yyyy</i> is the attribute value from <i>test-manifest</i>. When discrepancies for multiple attributes occur for the same file, each difference is noted on a separate line.</p> <p>The following default output shows the attribute differences for the <i>/etc/passwd</i> file. The output indicates that the <i>size</i>, <i>mtime</i>, and <i>contents</i> attributes have changed.</p> <pre>/etc/passwd: size control:74 test:81 mtime control:3c165879 test:3c165979 contents control:daca28ae0de97afd7a6b91fde8d57afa test:84b2b32c4165887355317207b48a6ec7</pre>
Programmatic Format	<pre>filename attribute control-val test-val [attribute control-val test-val] *</pre> <p><i>filename</i> Same as <i>filename</i> in the default format.</p> <p><i>attribute control-val test-val</i> A description of the file attributes that differ between the control and test manifests for each file. Each entry includes the attribute value from each manifest. See <i>bart_manifest(4CM)</i> for the definition of the attributes.</p> <p>Each line of the programmatic output describes all attribute differences for a single file.</p> <p>The following programmatic output shows the attribute differences for the <i>/etc/passwd</i> file. The output indicates that the <i>size</i>, <i>mtime</i>, and <i>contents</i> attributes have changed.</p> <pre>/etc/passwd size 74 81 mtime 3c165879 3c165979 contents daca28ae0de97afd7a6b91fde8d57afa 84b2b32c4165887355317207b48a6ec7</pre>
Manifest Generator	<p>0 Success</p> <p>1 Non-fatal error when processing files; for example, permission problems</p> <p>>1 Fatal error; for example, invalid command-line options</p>
Report Tool	<p>0 No discrepancies reported</p> <p>1 Discrepancies found</p> <p>>1 Fatal error executing comparison</p>

EXAMPLES**EXAMPLE 1** Creating a Default Manifest Without Computing Checksums

The following command line creates a default manifest, which consists of all files in the / file system. The -n option prevents computation of checksums, which causes the manifest to be generated more quickly.

```
bart create -n
```

EXAMPLE 2 Creating a Manifest for a Specified Subtree

The following command line creates a manifest that contains all files in the /home/nickiso subtree.

```
bart create -R /home/nickiso
```

EXAMPLE 3 Creating a Manifest by Using Standard Input

The following command line uses output from the find(1) command to generate the list of files to be cataloged. The find output is used as input to the bart create command that specifies the -I option.

```
find /home/nickiso -print | bart create -I
```

EXAMPLE 4 Creating a Manifest by Using a Rules File

The following command line uses a rules file, rules, to specify the files to be cataloged.

```
bart create -r rules
```

EXAMPLE 5 Comparing Two Manifests and Generating Programmatic Output

The following command line compares two manifests and produces output suitable for parsing by a program.

```
bart compare -p manifest1 manifest2
```

EXAMPLE 6 Comparing Two Manifests and Specifying Attributes to Ignore

The following command line compares two manifests. The dirmtime, lnmtime, and mtime attributes are not compared.

```
bart compare -i dirmtime lnmtime mtime manifest1 manifest2
```

EXAMPLE 7 Comparing Two Manifests by Using a Rules File

The following command line uses a rules file, rules, to compare two manifests.

```
bart compare -r rules manifest1 manifest2
```

ATTRIBUTES

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

bart(1MCM)

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	SUNWbart
Interface Stability	Evolving

SEE ALSO find(1), bart_manifest(4CM), bart_rules(4CM), attributes(5)

NAME	changemgr – Sun Management Center Change Manager command-line interface
SYNOPSIS	<code>/opt/SUNWchange/bin/changemgr</code> <i>command</i> [<i>options</i>] [<i>operands</i>]
DESCRIPTION	<p>The <code>changemgr(1MCM)</code> command is the command-line interface for the Sun Management Center Change Manager, henceforth referred to as Change Manager. This command-line interface performs the same operations that can be performed by using the browser user interface, such as software deployment tasks and system audit tasks.</p> <p>Change Manager commands must be run by an authenticated user.</p> <p>The command-line interface can be used to initiate a Change Manager session. A Change Manager session is a subshell in which you can run Change Manager commands as an authenticated user. You authenticate when you initiate the session. All operations run within the session are owned by the authenticated user.</p> <p>The command-line interface can also run custom scripts that execute multiple Change Manager commands. The script support facilitates the execution of multiple Change Manager operations. Authentication is performed once for the script instead of once per command-line invocation.</p>
OPTIONS	The <code>changemgr</code> command supports several command-line options.
Authentication Options	<p>Other than the <code>changemgr help</code> commands, all commands must be authenticated. In the context of a session, the session's authenticated identity is used.</p> <p>The following authentication options are supported:</p> <p><code>-p file</code> <i>file</i> consists of a single line, which contains the password. If <i>file</i> is <code>-</code>, then the user can supply the password as standard input.</p> <p> If the <code>-p</code> option is not supplied, then the <code>changemgr</code> command prompts the user for his password.</p> <p><code>-u username</code> Specify the user name to authenticate. If the <code>-u</code> option is not supplied, the user is the current UNIX real user ID, as reported by <code>id(1M)</code>.</p>
Common Options	<p>These options are used by more than one command.</p> <p><code>-d domain</code> Specify the Sun Management Center administrative domain on which to operate. In the context of a session, the default is the domain specified by the session, if any. By default, <i>domain</i> is the user's home domain.</p> <p><code>-o format</code> <i>format</i> is a blank-separated list or comma-separated list of property names. If you separate the property names with spaces, make sure that you surround the list of property names with quotes. The specified property values are displayed in a <i>name=value</i> format. If <i>format</i> is specified as <code>all</code>, then all properties are displayed. The output is suitable for programmatic parsing.</p>

changemgr(1MCM)

The output lists each file or folder on a line by itself. The name can be followed by property lines, which consist of a tab, property name, equals sign, and a property value. Each file or folder entry is separated from the next entry by a blank line.

For example, the output is arranged as follows:

```
path
    name=value
    ...

path
    name=value
    ...

...
```

OPERANDS

The following operands are supported:

<i>filepath</i>	An absolute path to a file or a relative (to the current directory) path to a file. This file path is not in the Change Manager repository.
<i>relpath</i>	Path to a file-like object (including a folder) that is relative to the top of the Change Manager repository.
<i>relfilepath</i>	Path to a file-like object (<i>not</i> including a folder) that is relative to the top of the Change Manager repository.
<i>reldirpath</i>	Path to a folder-like object that is relative to the top of the Change Manager repository.
<i>.type</i>	File name suffix that specifies the file type. File type suffixes are: <code>.flar</code> for archives, <code>.miniroot</code> for boot images, <code>.bmft</code> for manifests, <code>.brul</code> for audit rules files, <code>.txt</code> for reports, and <code>.cmsp</code> for shared profiles. Folders do not require a file name suffix.
<i>topopath</i>	Path to a topology object (including a host group) that is relative to the top of the selected administrative domain.
<i>hostpath</i>	Path to a managed host that is relative to the top of the selected administrative domain.
<i>hostname</i>	Network name of a host, for example, <code>netherfield.sun.com</code> .
<i>grouppath</i>	Path to a host group that is relative to the top of the selected administrative domain.

SUBCOMMANDS

The following sections describe the `changemgr` subcommands.

Sessions	<p><code>changemgr session [-u <i>username</i>] [-p <i>file</i>] [-d <i>domain</i>] [<i>command</i> [<i>command-arguments</i>]]</code></p> <p>Run the specified <i>command</i> in the context of a Change Manager session so that individual commands in a script (<i>command</i>) do not need authentication and startup overhead. The authentication and startup overhead is amortized over all of the commands.</p> <p><i>command</i> is normally an <code>sh(1)</code> or <code>ksh(1)</code> script that contains Change Manager commands in the form of the command-line interface.</p> <p>If <i>command</i> is <code>sh</code> or <code>ksh</code>, a subshell is spawned to create an interactive session. You are required to authenticate to initiate the session.</p> <p>If <i>command</i> is not supplied, then an interactive subshell of <code>\$SHELL</code> starts, if known to be compatible. If <code>\$SHELL</code> is not compatible, then an interactive <code>ksh</code> subshell starts.</p> <p>Note – The <code>csh(1)</code> shell cannot be used to run scripts or initiate a session.</p>
File Management Operations	<p><code>changemgr mkdir [-u <i>username</i>] [-p <i>file</i>] <i>new_reldirpath</i>...</code> Create one or more folders in the Change Manager repository.</p> <p><code>changemgr import [-u <i>username</i>] [-p <i>file</i>] <i>filepath</i>.[<i>type</i>] <i>relfilepath</i> . <i>type</i></code> Import a single file, <i>filepath</i>.[<i>type</i>], to the repository as <i>relfilepath</i> . <i>type</i>. The file being imported can have any file suffix, but the file name in the repository <i>must</i> have the appropriate suffix.</p> <p><code>changemgr import [-u <i>username</i>] [-p <i>file</i>] <i>filepath</i> . <i>type</i>... <i>reldirpath</i></code> Import one or more files to the specified folder, <i>reldirpath</i>, in the repository.</p> <p>Because this command uses the original file names when creating the files in the repository, the original names <i>must</i> have the appropriate suffixes.</p> <p><code>changemgr export [-u <i>username</i>] [-p <i>file</i>] <i>relfilepath</i> <i>filepath</i></code> Export a single file, <i>relfilepath</i>, from the repository as <i>filepath</i>.</p> <p><code>changemgr export [-u <i>username</i>] [-p <i>file</i>] <i>relfilepath</i>... <i>dirpath</i></code> Export one or more files to the specified folder, <i>dirpath</i>, outside of the repository.</p> <p><code>changemgr files [-u <i>username</i>] [-p <i>file</i>] [-l] [-d] [-R] [-o <i>format</i>] [<i>relpath</i>...]</code> List the specified files and folders, or the contents of the specified folders. When no path is specified, the objects in the root of the repository are listed.</p> <p>The default output format is one file or folder name per line.</p> <ul style="list-style-type: none"> -d Present information about the folder itself, rather than about the folder's contents. -l Present more information in tabular output. This output is not suitable for programmatic parsing. -R Recursively list the contents of a folder.

changemgr(1MCM)

Topology Operations

`changemgr delete [-u username] [-p file] relpath...`

Delete the specified files and folders.

Note that only empty folders can be deleted.

`changemgr fileset [-u username] [-p file] [-s name=value]... [-s name]... relpath...`

Set properties for the specified files and folders by using the `-s name=value` option. The `-s` option with just the property name deletes the property.

`-s name=value` Specify one or more *name=value* pairs. *name* is the property name, and *value* is the property value. Supply the `-s` option for each property value you want to set. If *value* is blank, then the property is assigned an empty value.

`-s name` Specify one or more property names to delete, where *name* is the property name. Supply the `-s` option for each property you want to delete.

`changemgr filemove [-u username] [-p file] old_relpath... new_dirpath`

Move files and folders to another folder. The original file and folder names are unchanged. The destination folder *must* already exist.

old_relpath can be a folder or a file.

`changemgr filemove [-u username] [-p file] old_relpath.type new_relpath.type`

Rename a file or a folder. The type of the renamed file *must* stay the same.

`changemgr mkgroup [-u username] [-p file] [-d domain] new_grouppath...`

Create one or more host groups.

`changemgr hosts [-u username] [-p file] [-l] [-g] [-R] [-d domain] [-o format] [topopath...]`

List information about *topopath*, which represents the specified managed hosts or host groups. With no path arguments, information is listed about the managed hosts and host groups in the root of the administrative domain.

The default output format is one managed host or host group name per line.

`-g` Present information about the group itself, rather than about the group's contents.

`-l` Present more information in tabular output. This output is not suitable for programmatic parsing.

`-R` Recursively list the contents of a group.

`changemgr add [-u username] [-p file] [-d domain] hostname hostpath`

Register a network host name as a Sun Management Center host name. The host path includes the host group and the host name. The name of the managed host can be different from the network host name.

`changemgr add [-u username] [-p file] [-d domain] hostname... grouppath`

Add the specified managed hosts to the specified host group, with the managed host names equal to the network host names.

Host Operations

`changemgr remove [-u username] [-p file] [-d domain] topopath...`

Remove managed hosts and host groups from the topology.

`changemgr hostset [-u username] [-p file] [-d domain] [-s name=value]... [-s name]... topopath...`

Set properties for the specified managed hosts and host groups by using the `-s name=value` option. The `-s` option with just the property name deletes the property.

`-s name=value` Specify one or more *name=value* pairs. *name* is the property name, and *value* is the property value. Supply the `-s` option for each property value you want to set. If *value* is blank, then the property is assigned an empty value.

`-s name` Specify one or more property names to delete, where *name* is the property name. Supply the `-s` option for each property you want to delete.

`changemgr hostmove [-u username] [-p file] [-d domain] old_topopath... new_grouppath`

Move managed hosts or host groups to another host group. The destination host group *must* already exist.

`changemgr hostmove [-u username] [-p file] [-d domain] old_topopath new_topopath`

Rename a single managed host or host group.

`changemgr update [-u username] [-p file] [-d domain] [-x operation] topopath...`

Update the specified managed hosts to conform to the configuration specified by their properties.

If *topopath* is a host group, all members of the host group are updated.

`-x operation` Specify the action to take after the update completes. If *operation* is `reboot`, then activate the newly installed software stack and reboot. If *operation* is `halt`, then activate the newly installed software stack and halt. The default operation is to reboot the managed host.

`changemgr fallback [-u username] [-p file] [-d domain] [-x operation] topopath...`

Restores the specified managed hosts to their state prior to the last `changemgr update` operation. This action only undoes the last update operation. This action does not change the parameters associated with the managed host. After the fallback operation, the managed host's running configuration will not match the parameters selected for it, which is the case immediately prior the update operation.

If *topopath* is a host group, all members of the host group are restored.

`-x operation` Specify the action to take after the fallback completes. If *operation* is `reboot`, then activate the newly selected software stack and reboot. If *operation* is `halt`, then activate the newly selected software stack and halt. The default operation is to reboot the managed host.

changemgr(1MCM)

Job Management Operations

changemgr reinstall [-u *username*] [-p *file*] [-d *domain*] *topopath*...

Reinstall the specified managed hosts. The reinstallation is equivalent to:

```
# reboot -- net - install
```

If *topopath* is a host group, all members of the host group are reinstalled.

changemgr setup [-u *username*] [-p *file*] [-d *domain*] *topopath*...

Set up files for initial installation. This operation is required before manually running `boot net - install` on the consoles of managed hosts.

If *topopath* is a host group, all files for the members of the host group are set up.

changemgr reboot [-u *username*] [-p *file*] [-d *domain*] *topopath*...

Reboot the specified managed hosts.

If *topopath* is a host group, all members of the host group are rebooted.

changemgr halt [-u *username*] [-p *file*] [-d *domain*] *topopath*...

Halt the specified managed hosts.

If *topopath* is a host group, all members of the host group are halted.

changemgr manifest [-u *username*] [-p *file*] [-d *domain*] -o *relpathprefix* [-r *relfilepath.brul*] *topopath*...

Create manifests for the specified managed hosts.

-o *relpathprefix* Specify the prefix to use when creating manifests. The host name and suffix are appended to the prefix to form the name of the manifest.

-r *relfilepath.brul* Specify the audit rules file to use when building manifests.

changemgr audit [-u *username*] [-p *file*] [-d *domain*] -o *relfilepath.txt* [-r *relfilepath.brul*] *relfilepath.bmft* *topopath*...

Compare managed host contents against a baseline manifest.

-o *relfilepath.txt* Specify the file path of the report.

-r *relfilepath.brul* Specify the audit rules file to use when auditing hosts.

changemgr info [-u *username*] [-p *file*] [-d *domain*] -o *relfilepath.txt* *topopath*...

Get software status information about the specified managed hosts. Store the results in the specified report.

-o *relfilepath.txt* Specify the file path of the report.

changemgr jobs [-u *username*] [-p *file*] [-l] [-o *format*] [*id*...]

Display the status of all outstanding jobs or of specified jobs.

-l Present more information in tabular output. This output is not suitable for programmatic parsing.

changemgr kill [-u *username*] [-p *file*] *id*...

Cancel currently running jobs or pending jobs.

	<p><code>changemgr ack [-u <i>username</i>] [-p <i>file</i>] <i>id</i>...</code></p> <p>Acknowledge the completion of the specified jobs. This action discards the status of the specified jobs.</p> <p>Use this command to purge completed job entries from the job list that were initiated by the browser interface.</p>
Miscellaneous	<p><code>changemgr help</code> Provide a one-line summary of the available subcommands. No authentication is required.</p> <p><code>changemgr help <i>subcommand</i></code> Provide a summary of the specified subcommand. No authentication is required.</p>
SEE ALSO	<p><code>csh(1)</code>, <code>date(1)</code>, <code>ksh(1)</code>, <code>sh(1)</code>, <code>id(1M)</code></p>
EXAMPLES	<p>EXAMPLE 1 Running Commands in an Interactive Change Manager Session</p> <p>The following example shows an interactive Change Manager session. The <code>changemgr session</code> command starts a subshell in which you can run authenticated <code>changemgr</code> commands.</p> <p>This example shows how to purge a completed job from the job queue. This job, <code>IC_1</code>, was initiated from the browser interface. When the tasks are completed, exit the session by typing <code>exit</code> at the subshell prompt.</p> <pre>\$ changemgr session Password: password \$ changemgr jobs -l IC_1 IC_1 succeeded \$ changemgr ack IC_1 \$ changemgr jobs l IC_1 \$ exit</pre> <p>EXAMPLE 2 Running Scripts in a Change Manager Session</p> <p>This example shows how to use the <code>changemgr session</code> command to run a script. The command line runs the script called <code>deploy-web</code>, which contains the following:</p> <pre>\$ cat deploy-web #!/bin/sh changemgr import "\$1" / changemgr filesset -s MediaName=s9.miniroot "\$1" changemgr hostset -s base_config_flar_archive="/\$1" "\$2" changemgr update "\$2" \$</pre> <p>The following command line runs the <code>deploy-web</code> script.</p> <pre>\$ changemgr session deploy-web web.flar host1</pre>

cmgetprop(1MCM)

NAME	cmgetprop – Get Change Manager property value
SYNOPSIS	cmgetprop <i>property-name</i>
DESCRIPTION	<p>The <code>cmgetprop(1MCM)</code> command writes the value of the specified property to standard output. Note that no value is returned when the property is not set.</p> <p>Use the <code>cmgetprop</code> command in deployment finish scripts to get property values. Change Manager finish scripts are stored in the <code>/etc/ichange.d</code> directory.</p> <p>The <code>cmgetprop</code> command is included in the <code>\$PATH</code> supplied to the Change Manager finish scripts.</p>
EXAMPLES	<p>EXAMPLE 1 Using the <code>cmgetprop</code> Command</p> <p>The following line might exist in a Change Manager finish script.</p> <pre>FILENAME='cmgetprop FNAME'</pre> <p>This code assigns the value of the <code>FNAME</code> property to the <code>FILENAME</code> variable.</p>
SEE ALSO	<i>Sun Management Center Change Manager 1.0 Administration Guide</i>

NAME	cminst, cmuninst – Install or uninstall Change Manager software
SYNOPSIS	cminst cmuninst
DESCRIPTION	<p>The cminst command installs the Change Manager software on a Change Manager server or on a master system. A master system is one on which to create a software stack and build the Solaris Flash archive. The cmuninst command uninstalls the Change Manager software.</p> <p>The cminst(1MCM) command must be run after the Sun Management Center software is installed and configured. Change Manager requires that you install the Sun Management Center premier package and add patch 110938-06 to the Change Manager server. Configure the Sun Management Center software on the Change Manager server by running the Sun Management Center es-setup command. See “Installing and Configuring the Change Manager Server” in <i>Sun Management Center Change Manager 1.0 Administration Guide</i>.</p> <p>The cminst command determines whether the system being installed is the Change Manager server or a master system. If cminst detects Sun Management Center server software, it installs the Sun Management Center Web Console, Change Manager server and agent, and Change Manager application components. If, however, cminst detects only Sun Management Center agents, then it installs only the Change Manager agent component.</p> <p>The cmuninst command only removes the Sun Management Center Web Console and Change Manager packages from the Change Manager server. It does not remove Change Manager data, which is the Change Manager database or the contents of the Change Manager repository.</p>
Change Manager Server	<p>The cminst command performs the following tasks when run on the Change Manager server:</p> <ul style="list-style-type: none"> ■ Stops the Sun Management Center server ■ Installs the Sun Management Center Web Console package ■ Installs the Change Manager server and agent components ■ Installs the Change Manager application component ■ Runs es-setup to configure the Change Manager components ■ Creates the <code>/var/opt/SUNWsymon/cfg/ichange.cfg</code> file <p>If the <code>/var/opt/ichange</code> directory does not have enough available disk space for the repository, then edit the <code>ichange.cfg</code> file. Change the value of the <code>cmdataroot</code> parameter to point to another file system that has sufficient disk space.</p> <p>Note – After making changes to the <code>ichange.cfg</code> file, restart the Sun Management Center server by typing es-restart -s.</p>

cminst(1MCM)

- Asks you to provide the same Sun Management Center server seed you supplied when you configured Sun Management Center on the Change Manager server
If you change the value of the seed after running `cminst`, then you must update the `agentseed` parameter in the `ichange.cfg` file. Then, restart the Sun Management Center server.
- Asks you to specify a directory in which to store the Change Manager database
This directory must have at least 500 Mbytes of available disk space.
- Sets up the Change Manager database
- Asks you to specify a directory in which to store the Change Manager repository
This directory might need several Gbytes of available disk space to store files such as Solaris Flash archives, Solaris boot images, and manifests.
- Restarts the Sun Management Center server
- Starts the Sun Management Center web server

The `cmuninst` command performs the following tasks when run on the Change Manager server:

- Stops the Sun Management Center server
- Uninstalls the Sun Management Center Web Console package
- Uninstalls the Change Manager server and agent components
- Preserves the Change Manager database
- Preserves the `ichange.cfg` file
- Preserves the Change Manager repository
- Restarts the Sun Management Center server
- Restarts the Sun Management Center web server

Master System

The `cminst` command installs the Change Manager agent package on the master system and restarts the Sun Management Center agents.

The `cmuninst` command performs the following tasks when run on a master system:

- Stops the Sun Management Center agent
- Uninstalls the Change Manager agent package
- Restarts the Sun Management Center agent

SEE ALSO

Sun Management Center Change Manager 1.0 Administration Guide

NAME	cmsetup – Set up or remove Change Manager data
SYNOPSIS	cmsetup [-c] [-h] [-s] [-u]
DESCRIPTION	<p>The <code>cmsetup(1MCM)</code> command is used to initialize, re-initialize, or remove Change Manager data on the Change Manager server. Change Manager data includes both the metadata maintained in the database tables and the repository where files, such as Solaris Flash archives, Solaris boot images, and manifests, are stored.</p> <p>When Change Manager is installed, the <code>cminst(1MCM)</code> command can either initialize Change Manager data immediately or, at the user's option, defer configuration. If you choose to have <code>cminst</code> initialize data, you will not have to run <code>cmsetup</code>. If you choose to defer configuration, you must run <code>cmsetup</code> manually.</p> <p>The Change Manager database is dependent on the underlying Sun Management Center database. Therefore, you must run <code>cmsetup(1MCM)</code> whenever the Sun Management Center database has been re-initialized, such as by <code>/opt/SUNWsymon/sbin/es-setup</code>. <code>cmsetup</code> <i>must</i> be run after <code>es-setup</code> has completed. You must also run <code>cmsetup</code> after installing or re-initializing the Performance Reporting Manager (PRM) add-on.</p>
OPTIONS	<p>The following options are supported:</p> <ul style="list-style-type: none"> -c Run <code>es-setup</code> to set up or re-initialize Change Manager data. This option must be used the first time that Change Manager software is installed, and whenever you want to re-initialize the Change Manager database. -h Run help. Use this option to obtain help in using this command. -s Run <code>es-setup</code>. This option initializes both Sun Management Center data and Change Manager data. -u Remove (unsetup) Change Manager data. This option removes the Change Manager database, removes all Change Manager data files, and disables boot services to the managed hosts.
SEE ALSO	<code>cminst(1MCM)</code> , <i>Sun Management Center Change Manager 1.0 Administration Guide</i>

cmsetup(1MCM)

Change Manager File Formats

bart_manifest(4CM)

NAME	bart_manifest – system audit manifest file								
DESCRIPTION	<p>The <code>bart(1MCM)</code> command generates a manifest that describes the contents of a managed host. A manifest consists of a header and entries. Each entry represents a single file. Entries are sorted in ascending order by file name. Any nonstandard file names, such as those that contain embedded newline or tab characters, have the special characters quoted prior to being sorted. See Quoting Syntax.</p> <p>Lines that begin with <code>!</code> supply metadata about the manifest. The manifest version line indicates the manifest specification version. The date line shows the date on which the manifest was created, in <code>date(1)</code> form.</p> <p>Some lines are ignored by the manifest comparison tool. Ignored lines include blank lines, lines that consist only of white space, and comments that begin with <code>#</code>.</p> <p>In addition to metadata lines, the header contains the format comment block. This comment block lists the attributes reported for each file type.</p> <p>To see the format of an manifest file, see EXAMPLES.</p>								
Manifest File Entries	<p>Each manifest file entry is a single line of one of the following forms, depending on the file type:</p> <pre> <i>fname</i> D <i>size mode acl dirmtime uid gid</i> [<i>xattr xcontents</i>] * <i>fname</i> P <i>size mode acl mtime uid gid</i> [<i>xattr xcontents</i>] * <i>fname</i> S <i>size mode acl mtime uid gid</i> [<i>xattr xcontents</i>] * <i>fname</i> F <i>size mode acl mtime uid gid contents</i> [<i>xattr xcontents</i>] * <i>fname</i> L <i>size mode acl lnmtime uid gid dest</i> [<i>xattr xcontents</i>] * <i>fname</i> B <i>size mode acl mtime uid gid devnode</i> [<i>xattr xcontents</i>] * <i>fname</i> C <i>size mode acl mtime uid gid devnode</i> [<i>xattr xcontents</i>] * </pre> <p>Each entry begins with <i>fname</i>, which is the name of the file. To prevent parsing problems that are caused by special characters embedded in file names, file names are encoded as described in Quoting Syntax.</p> <p>Subsequent fields represent the following file attributes.</p> <table> <tr> <td><i>type</i></td><td>Type of file. Possible values are as follows:</td></tr> <tr> <td></td><td> <ul style="list-style-type: none"> ■ B for a block device node ■ C for a character device node ■ D for a directory ■ F for a file ■ L for a symbolic link ■ P for a pipe ■ S for a socket </td></tr> <tr> <td><i>size</i></td><td>File size in bytes.</td></tr> <tr> <td><i>mode</i></td><td>Octal number that represents the permissions of the file.</td></tr> </table>	<i>type</i>	Type of file. Possible values are as follows:		<ul style="list-style-type: none"> ■ B for a block device node ■ C for a character device node ■ D for a directory ■ F for a file ■ L for a symbolic link ■ P for a pipe ■ S for a socket 	<i>size</i>	File size in bytes.	<i>mode</i>	Octal number that represents the permissions of the file.
<i>type</i>	Type of file. Possible values are as follows:								
	<ul style="list-style-type: none"> ■ B for a block device node ■ C for a character device node ■ D for a directory ■ F for a file ■ L for a symbolic link ■ P for a pipe ■ S for a socket 								
<i>size</i>	File size in bytes.								
<i>mode</i>	Octal number that represents the permissions of the file.								

<i>acl</i>	ACL attributes for the file. For a file with ACL attributes, this field contains the output from <code>acltotext()</code> .
<i>uid</i>	Numerical user ID of the owner of this entry.
<i>gid</i>	Numerical group ID of the owner of this entry.
<i>dirmtime, lnmtime, mtime</i>	Last modification time, in seconds since 00:00:00 UTC, January 1, 1970, for directories, links, and other files, respectively.
<i>contents</i>	Checksum value of the file. This attribute is only specified for regular files. If you turn off context checking or if checksums cannot be computed, the value of this field is <code>-</code> .
<i>dest</i>	Destination of a symbolic link.
<i>devnode</i>	Value of the device node. This attribute is for character device files and block device files only.
<i>[xattr xcontents]*</i>	Zero or more checksum values for files with extended attributes. The attributes are described in alphabetical order. If you specify the <code>-n</code> option or the <code>IGNORE</code> <code>contents</code> directive, the value of <i>xcontents</i> is <code>-</code> .

Quoting Syntax The rules file supports a quoting syntax for representing nonstandard file names.

When generating an manifest for file names that embed tab, space, or newline characters, the special characters are encoded in their octal forms.

Input Character	Quoted Character
(space)	\(space)
(tab)	\(tab)
(newline)	\(newline)
?	\?
[\[
*	*

EXAMPLES **EXAMPLE 1** Sample Manifest Output

The following is a sample system manifest file. The file entries are sorted by the encoded versions of the file names to correctly handle special characters.

```
! Version 1.0
! Mon Feb 11 10:55:30 2002
# Format:
```

bart_manifest(4CM)

EXAMPLE 1 Sample Manifest Output *(Continued)*

```
# fname D size mode acl dirmtime uid gid [xattr xcontents]*
# fname P size mode acl mtime uid gid [xattr xcontents]*
# fname S size mode acl mtime uid gid [xattr xcontents]*
# fname F size mode acl mtime uid gid contents [xattr xcontents]*
# fname L size mode acl lnmtime uid gid dest [xattr xcontents]*
# fname B size mode acl mtime uid gid devnode [xattr xcontents]*
# fname C size mode acl mtime uid gid devnode [xattr xcontents]*
/etc D 3584 40755 user::rwx,group::r-x,mask::r-x,other::r-x, 3c6803d7 0 3
/etc/.login F 524 100644 user::rw-,group::r--,mask::r--,other::r--, 3c165878
0 3 27b53d5c3e844af3306f1f12b330b318
/etc/.pwd.lock F 0 100600 user::rw-,group::---,mask::---,other::---, 3c166121
0 0 d41d8cd98f00b204e9800998ecf8427e
/etc/.syslog_door L 20 120777 user::rw-,group::r--,mask::rwx,other::r--,
3c6803d5 0 0 /var/run/syslog_door
/etc/autopush L 16 120777 user::r-x,group::r-x,mask::r-x,other::r-x, 3c165863
0 0 ../sbin/autopush
/etc/cron.d/FIFO P 0 10600 user::rw-,group::---,mask::---,other::---, 3c6803d5
0 0
```

SEE ALSO date(1), bart(1MCM), bart_rules(4CM), attributes(5)

NAME	bart_rules – bart rules file
DESCRIPTION	<p>The rules file is a text file that is used by the <code>bart(1MCM)</code> command. The rules file determines which files to validate and which file attributes of those files to ignore.</p> <p>Some lines are ignored by the manifest comparison tool. Ignored lines include blank lines, lines that consist only of white space, and comments that begin with #.</p> <p>The rules file supports three directives: <code>CHECK</code>, <code>IGNORE</code>, and a subtree directive, which is an absolute path name and optional pattern matching modifiers. The rules file uses the directives to create logical blocks.</p>
Syntax	<p>The syntax for the rules file is as follows:</p> <pre>[IGNORE attribute...] * [CHECK] [attribute...] * subtree1 [pattern...] * [IGNORE attribute...] * [CHECK] [attribute...] * subtree2 [pattern...] * subtree3 [pattern...] * subtree4 [pattern...] * [IGNORE attribute...] * [CHECK] [attribute...] * ...</pre>
Rule Blocks	<p>Rule blocks are composed of statements that are created by using directives and arguments. There are three types of blocks.</p> <p>Global Block The first block in the file. The block is considered “global” if it specifies <code>CHECK</code> and <code>IGNORE</code> statements, but no previous subtree statement. A global block pertains to all subsequent blocks.</p> <p>Local block A block that specifies <code>CHECK</code> and <code>IGNORE</code> statements as well as a subtree directive. The rules in this block pertain to files and directories found in the specified subtree.</p> <p>Heir block A block that contains a null <code>CHECK</code> statement, no arguments. This block inherits the global <code>CHECK</code> statements and <code>IGNORE</code> statements.</p> <p>Note – The order in which <code>CHECK</code> and <code>IGNORE</code> statements appear in blocks is important. The <code>bart</code> command processes <code>CHECK</code> and <code>IGNORE</code> statements in the order in which they are read, with later statements overriding earlier statements.</p> <p>Subtree specifications must appear one per line. Each specification must begin with an absolute path name. Optionally, each specification can be followed by pattern-matching arguments.</p>

bart_rules(4CM)

Pattern Matching Statements

When a file being tracked belongs to more than one subtree directive, bart performs the following resolution steps:

- Applies the CHECK and IGNORE statements set in the global block. Note that all CHECK and IGNORE statements are processed in order.
- Finds the last subtree directive that matches the file.
- Processes the CHECK and IGNORE statements that belong to the last matching subtree directive. These statements are processed in the order in which they are read, overriding global settings.

AND Statement

For a given subtree directive, all pattern matching statements are logically ANDed with the subtree. Patterns have the following syntax:

- Wildcards are permitted for both the subtree and pattern matching statements.
- The exclamation point (!) character represents logical NOT.
- A pattern that terminates with a slash is a subtree. The absence of a slash indicates that the pattern is not a directory. The subtree itself does not require an end slash.

For example, the following subtree example includes the contents of /home/nickiso/src except for object files, core files, and all of the SCCS subtrees. Note that directory names that terminate with .o and directories named core are *not* excluded because the patterns specified do not terminate with /.

```
/home/nickiso/src !*.o !core !SCCS/  
CHECK    all
```

OR Statement

Group multiple subtree directives together. Such subtree directives are logically ORed together.

```
/home/nickiso/src !*.o !core  
/home/nickiso/Mail  
/home/nickiso/docs *.sdw  
CHECK    all  
IGNORE   mtime lnmtime dirmtime
```

The files included in the previous example are as follows:

- Everything under /home/nickiso/src except for *.o and core files
- Everything under /home/nickiso/Mail
- All files under /home/nickiso/docs that end in *.sdw

For these files, all attributes are checked except for modification times.

File Attributes

The bart command uses CHECK and IGNORE statements to define which attributes to track or ignore. Each attribute has an associated keyword.

The attribute keywords are as follows:

- acl
- all
- contents
- dest
- devnode
- dirmtime
- gid
- lnmtime
- mode
- mtime
- size
- type
- uid
- xattrs

The all keyword refers to all file attributes. See `bart_manifest(4CM)`.

EXAMPLES

EXAMPLE 1 Sample Rules File

```
# Global rules, track everything except dirmtime.
CHECK    all
IGNORE    dirmtime

# The files in /data* are expected to change, so don't bother
# tracking the attributes expected to change.
# Furthermore, by specifying ``IGNORE contents,`` you save
# time and resources.
/data*
IGNORE    contents mtime size

/home/nickiso f* bar/
IGNORE    acl

# For /usr, apply the global rules.
/usr
CHECK

# Note: Since /usr/tmp follows the /usr block, the /usr/tmp
# subtree is subjected to the ``IGNORE all.``
/usr/tmp
/home/nickiso *.o
/home/nickiso core
/home/nickiso/proto
IGNORE    all
```

The following files are cataloged based on the sample rules file:

- All attributes, except for `dirmtime`, `mtime`, `size`, and `contents`, are tracked for files under the `/data*` subtrees.
- Files under the `/usr` subtree, except for `/usr/tmp`, are cataloged by using the global rules.

bart_rules(4CM)

EXAMPLE 1 Sample Rules File *(Continued)*

- If the /home/nickiso/foo.c file exists, its attributes, except for acl and dirmtime, are cataloged.
- All .o and core files under /home/nickiso, as well as the /home/nickiso/proto and /usr/tmp subtrees, are ignored.
- If the /home/nickiso/bar/foo.o file exists, it is ignored because it is subject to the last block.

SEE ALSO bart(1MCM), bart_manifest(4CM), attributes(5)

NAME	cmisp – Sun Management Center Change Manager shared profile
DESCRIPTION	<p>Shared profiles describe how one or more managed hosts are configured with a software stack. Much of the information described by these profiles is the same as described in an installation profile.</p> <p>A shared profile file name must use the .cmisp suffix, for example, web-server.cmis.</p> <p>The shared profile is a set of properties and associated property values, one property per line. The property format is:</p> <pre>property-name=property-value</pre> <p>Lines that contain only whitespace are ignored. Lines whose first non-whitespace character is # or ! are comments. The rest of the lines in the shared profile describe properties.</p> <p>The property name consists of all the characters in the line starting with the first non-whitespace character and up to, but not including, the first equals sign (=) character.</p> <p>The property value consists of the rest of the line after the equals sign.</p> <p>If you want a backslash character to appear in the property value, escape the backslash with another backslash.</p> <p>The following example shows that the value of the base_config_target_arch property is sun4u.</p> <pre>base_config_target_arch=sun4u</pre>
EXAMPLES	<p>EXAMPLE 1 Default Shared Profile for Creating One Boot Environment</p> <p>The following example shared profile uses the default values to create one boot environment.</p> <pre># # Example shared profile for a system with one boot environment. # # This example shared profile assumes a disk that is no smaller than # 7 Gbytes in size. # # You must also specify the following properties with appropriate # values: # # o base_config_flar_archive # Name of the Solaris Flash archive associated with this # shared profile # o base_config_boot_image # Location of the Solaris boot image associated with the # specified Solaris Flash archive # o base_config_sysidcfg_rootpw</pre>

EXAMPLE 1 Default Shared Profile for Creating One Boot Environment *(Continued)*

```
#      Encrypted root password entry, which can be taken from the
#      password entry in the /etc/shadow file
# o base_config_sysidcfg_timezone
#      Appropriate time zone value from /usr/share/lib/zoneinfo
#
base_config_target_arch=sun4u
base_config_sysidcfg_nameservice=none
base_config_sysidcfg_networkinterface=primary
base_config_sysidcfg_netmask=255.255.255.0
base_config_sysidcfg_ipv6=no
base_config_sysidcfg_defaultroute=none
base_config_sysidcfg_systemlocale=C
base_config_sysidcfg_terminal=vt100
base_config_sysidcfg_timeserver=localhost
base_config_sysidcfg_security_policy=none
base_config_be_0_root_device=rootdisk.s0
base_config_be_0_root_size=free
base_config_be_0_var_device=rootdisk.s3
base_config_be_0_var_size=1024
base_config_local_swap1_device=rootdisk.s1
base_config_local_swap1_size=2048
```

EXAMPLE 2 Default Shared Profile for Creating Two Boot Environments

The following example shared profile uses the default values to create two boot environments.

```
#
# Example shared profile for a system with two boot environments.
#
# This example shared profile assumes a disk that is no smaller than
# 12 Gbytes in size.
#
# You must also specify the following properties with appropriate
# values:
#
# o base_config_flar_archive
#      Name of the Solaris Flash archive associated with this
#      shared profile
# o base_config_boot_image
#      Location of the Solaris boot image associated with the
#      specified Solaris Flash archive
# o base_config_sysidcfg_rootpw
#      Encrypted root password entry, which can be taken from the
#      password entry in the /etc/shadow file
# o base_config_sysidcfg_timezone
#      Appropriate time zone value from /usr/share/lib/zoneinfo
#
base_config_target_arch=sun4u
base_config_sysidcfg_nameservice=none
base_config_sysidcfg_networkinterface=primary
base_config_sysidcfg_netmask=255.255.255.0
base_config_sysidcfg_ipv6=no
```

EXAMPLE 2 Default Shared Profile for Creating Two Boot Environments (Continued)

```
base_config_sysidcfg_defaultroute=none
base_config_sysidcfg_systemlocale=C
base_config_sysidcfg_terminal=vt100
base_config_sysidcfg_timeserver=localhost
base_config_sysidcfg_security_policy=none
base_config_be_0_root_device=rootdisk.s0
base_config_be_0_root_size=free
base_config_be_0_var_device=rootdisk.s3
base_config_be_0_var_size=1024
base_config_be_1_root_device=rootdisk.s4
base_config_be_1_root_size=4096
base_config_be_1_var_device=rootdisk.s5
base_config_be_1_var_size=1024
base_config_local_swap1_device=rootdisk.s1
base_config_local_swap1_size=2048
```

SEE ALSO *Sun Management Center Change Manager 1.0 Administration Guide*

ichange.cfg(4CM)

NAME	ichange.cfg – Sun Management Center Change Manager configuration file		
DESCRIPTION	<p>You can change the behavior of the Change Manager application by modifying certain runtime parameters. These parameters are stored in the application configuration file, <code>ichange.cfg</code>. The configuration file is located in the <code>/var/opt/SUNWsymon/cfg</code> directory.</p> <p>Note – When you make changes to the <code>ichange.cfg</code> file, you must restart the Sun Management Center services before the changes can take effect.</p> <p>Restart the Sun Management Center services by running the following command as superuser:</p> <pre># /opt/SUNWsymon/sbin/es-restart -S</pre>		
File Location Parameter	<p>The <code>cmdataroot</code> parameter specifies the location of the Change Manager file hierarchy. <code>cmdataroot</code> points to the root of the Change Manager file hierarchy.</p> <p>You might want to change the value of this parameter if you are moving the Change Manager repository to a different location.</p> <p>The default value is the <code>/var/opt/ichange</code> directory.</p>		
Sun Management Center Agent Parameter	<p>Following is the Sun Management Center agent parameter:</p> <table><tr><td><code>agentport</code></td><td>Sun Management Center agent port to be used. Any update or reinstallation operations in which host parameters do not explicitly specify a value for the agent port will use this one.</td></tr></table> <p>The default value is 161.</p>	<code>agentport</code>	Sun Management Center agent port to be used. Any update or reinstallation operations in which host parameters do not explicitly specify a value for the agent port will use this one.
<code>agentport</code>	Sun Management Center agent port to be used. Any update or reinstallation operations in which host parameters do not explicitly specify a value for the agent port will use this one.		
Job Execution Parameters	<p>The following parameters describe job execution characteristics:</p> <table><tr><td><code>boottimeout</code></td><td><p>Interval to wait for a reboot, update, or reinstallation to complete. This is equivalent to the time it takes for the following events to occur:</p><ul style="list-style-type: none">■ Complete the entire software installation, including any finish scripts■ The subsequent reboot to return■ Any boot-time startup procedures to run■ The Sun Management Center agent to reestablish communications with the management server<p>If the host does not reboot and reestablish agent communications with the Sun Management Center server within the specified time period, the associated management operation will fail with a timeout error.</p><p>You might need to change this value if any of the following are true:</p></td></tr></table>	<code>boottimeout</code>	<p>Interval to wait for a reboot, update, or reinstallation to complete. This is equivalent to the time it takes for the following events to occur:</p> <ul style="list-style-type: none">■ Complete the entire software installation, including any finish scripts■ The subsequent reboot to return■ Any boot-time startup procedures to run■ The Sun Management Center agent to reestablish communications with the management server <p>If the host does not reboot and reestablish agent communications with the Sun Management Center server within the specified time period, the associated management operation will fail with a timeout error.</p> <p>You might need to change this value if any of the following are true:</p>
<code>boottimeout</code>	<p>Interval to wait for a reboot, update, or reinstallation to complete. This is equivalent to the time it takes for the following events to occur:</p> <ul style="list-style-type: none">■ Complete the entire software installation, including any finish scripts■ The subsequent reboot to return■ Any boot-time startup procedures to run■ The Sun Management Center agent to reestablish communications with the management server <p>If the host does not reboot and reestablish agent communications with the Sun Management Center server within the specified time period, the associated management operation will fail with a timeout error.</p> <p>You might need to change this value if any of the following are true:</p>		

ichange.cfg(4CM)

- A managed host takes an unusually long time to boot.
- Your Sun Management Center topology requires a long time to establish server context for a newly configured agent.
- A particular software stack takes a long time to install.
- Finish and startup scripts take a long time to complete.

The default value is 1800000 milliseconds (30 minutes).

downtimeout Amount of time to wait for a managed host to shut down after a management operation has requested a reboot. This is effectively the time it takes for a system to complete an `init 6` sequence.

If the host does not shut itself down within the specified time, the associated management operation will fail with a timeout error. Thus, you might need to adjust this value if a host or software stack takes an unusually long time to complete its shutdown sequence.

The default value is 300000 milliseconds (5 minutes).

Other Parameters	debug	Control the printing of debug messages to the Sun Management Center server console.
		The default value is <code>false</code> . To turn on the debugging messages, change the value to <code>true</code> .

SEE ALSO *Sun Management Center Change Manager 1.0 Administration Guide*

ichange.cfg(4CM)