Installation Guide for Solaris™ Operating Systems

Sun™ ONE Calendar Server

Version 6.0

816-6707-10
December 2003

Contents

List of Tables

List of Figures

About This Guide

This guide describes how to install and configure Sun™ Open Net Environment (Sun ONE) Calendar Server (formerly iPlanet™ Calendar Server) on Solaris™ Systems. Topics in this chapter include:

For the most recent information about this release, see the Calendar Server 6.0 Release Notes on the following documentation Web site:

`http://docs.sun.com/coll/S1_CalendarServer_60`

Who Should Read This Guide

This guide is intended for Calendar Server administrators and support specialists who are responsible for installing and configuring Calendar Server 6.0.

What You Need to Know

Before you install Calendar Server 6.0, you should be familiar with:

- Basic administrative procedures of the Solaris™ Operating System
- Sun Java™ Enterprise System installer, as described in the *Sun Java Enterprise System Installation Guide*
- Sun ONE Directory Server 5.x, which is used for user authentication and to store user preferences
- Sun ONE Identity Server 6.1, if you plan to use Identity Server features such as CLI utilities for provisioning or single sign-on (SSO)
- Any other Java Enterprise System products such as Sun ONE Portal Server that you are planning to integrate with Calendar Server 6.0

How This Guide is Organized

Table 1 Organization of the Sun ONE Calendar Server Installation Guide

Chapter or Appendix	Description
About This Guide (this chapter)	Describes the audience, requirements, organization, document conventions, and related information.
Chapter 1, “Planning for Installation and Configuration”	Describes how to plan for the installation and configuration of Calendar Server 6.0.
Chapter 2, “Configuring Calendar Server 6.0”	Describes how to configure Calendar Server 6.0 and Sun ONE Directory Server 5.x on Solaris Systems.
Chapter 3, “Migrating Calendar Server Data”	Describes the Calendar Server migration utilities.
Appendix A, “Configuration Worksheets”	Provides worksheets to plan for the configuration of Calendar Server.
Appendix B, “LDAP Directory Server Considerations”	Describes how to update an LDAP directory server schema manually and to resolve conflicting OIDs in the LDAP schema directory.
Appendix C, “Calendar Server 5.x to 6.0 Upgrade/Migration Process”	Describes how to upgrade and migrate to Calendar Server 6.0 from Sun ONE or iPlanet Calendar Server 5.x.
Glossary	Describes Calendar Server terms.
Index	

Document Conventions Used in This Guide

This guide uses conventions for the Solaris Operating System.

Monospaced Font

The `Monospaced Font` typeface is used for text that appears on the computer screen or text that you should type. It is also used for file names, path names, distinguished names, functions, and examples.

Italicized Font

The *Italicized Font* typeface represents text that you enter using information that is unique to your installation. It is used for server directory paths and names. For example, in this guide you will see directory path references in the form:

cal_svr_base`/opt/SUNWics5/cal/`

In these references, *cal_svr_base* represents the base or root directory location where you installed Calendar Server.

Square Brackets [ ]

Square (or straight) brackets [ ] enclose optional parameters. For example, the `-q` and `-m` options are optional in the following syntax for the`ics2migrate` migration utility:

```
ics2migrate [-q] [-m ldap]
```

Vertical Bar (|)

A vertical bar (`|`) separates alternatives in a horizontal list. For example, the `-s`, `-f`, and `-l` options have alternatives you can specify:

```
ics2migrate [-q] [-m db] [-s def|none] [-f def|none]
  [-l min|max] source target
```

Command-Line Prompts

Command line prompts (% for a C-Shell or $ for a Korn or Bourne shell) are not specified in the examples. Depending on the shell you are using, you will see a variety of different command-line prompts. However, you should enter the command as it appears in this document unless specifically noted otherwise.

Related Third-Party Web Site References

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

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

Where to Find Related Information

Calendar Server includes documentation for administrators, developers, and end users. In addition to this guide, the following Calendar Server documents are available in PDF and HTML formats on the documentation Web site:

- *Sun ONE Calendar Server Release Notes*
- *Sun ONE Calendar Server Administrator's Guide*
- *Sun ONE Calendar Server Programmer's Manual*
- *Sun ONE Messaging and Collaboration Schema Reference*
- *Sun ONE Messaging and Collaboration Event Notification Service Manual*

To view these documents, see the following documentation Web site:

`http://docs.sun.com/coll/S1_CalendarServer_60`

For end users, online help is available from Sun ONE Calendar Express.

Chapter 1

Planning for Installation and Configuration

The installation and configuration of Sun ONE Calendar Server 6.0 on Solaris Systems has significant changes from previous Calendar Server releases. To install Calendar Server 6.0 on Solaris systems, you must use the Sun Java Enterprise System installer, which also installs other Sun component products.

To install and configure Sun ONE Calendar Server 6.0, follow these steps:

1. "Gather Your Directory Server Configuration Information" on page 16.
2. "Gather Your Calendar Server 6.0 Configuration Information" on page 18.
3. "Install Calendar Server Using the Java Enterprise System Installer" on page 22.
4. "Configure Calendar Server 6.0" on page 22.

For the most recent information about Calendar Server 6.0, see the Release Notes on the following documentation Web site:

`http://docs.sun.com/coll/S1_CalendarServer_60`

CAUTION If you have Sun™ ONE or iPlanet™ Calendar Server 5.x installed at your site, see Appendix C, "Calendar Server 5.x to 6.0 Upgrade/Migration Process" for information about upgrading to the 6.0 release.

If you need to uninstall Calendar Server 6.0, see "Uninstalling Calendar Server 6.0" on page 23.

Gather Your Directory Server Configuration Information

The Directory Server Setup (`comm_dssetup.pl`) Perl script configures Sun ONE Directory Server 5.x for Calendar Server 6.0 (and Messaging Server 6.0). When you run `comm_dssetup.pl`. you will need to provide the following information. To help you keep track of this information, use "Directory Server Setup Script Worksheet" on page 80.

- What is the Directory Server root path name? The default is `/var/mps/serverroot`.
- If you have multiple instances of Directory Server, which instance do you want to use for Calendar Server 6.0?
- What is the Directory Manager Distinguished Name (DN) and password? The default DN is `"cn=Directory Manager"`.
- Will the Directory Server be used for Users and Groups. That is, do you want to use Directory Server to store both configuration and user data (yes) or configuration data only (no)? The default is both (yes).
- If the Directory Server will be used for Users and Groups, what is the User and Group root suffix? The default is `o=usergroup`.
- Which version of the Sun ONE LDAP Schema do you want to use?
  - Option 1– LDAP Schema, v.1 (the default)
  - Option 1.5–ONE LDAP Schema v.2 Compatibility Mode
  - Option 2–LDAP Schema, v.2 Native Mode

  For more information, see Deciding Which Schema to Use.
- In you plan to use LDAP Schema, v.1, what is the DC Tree root suffix? The default is `o=internet`.
- Do you want to update the schema (yes/no)? The default is yes. If you answer yes, you must have a `config` directory with the schema files.
- Do you want to configure new Directory Server indexes (yes/no)? The default is yes. For Calendar Server 6.0, `comm_dssetup.pl` adds indexes for the `icsCalendar` and `icsCalendarOwned` attributes.
- What is the path to the directory where the schema files are located? The default is `./schema`.

Deciding Which Schema to Use

Calendar Server 6.0 supports Sun ONE LDAP Schema, v.1 and Sun ONE LDAP Schema, v.2 Native Mode.

Use the following guidelines to choose the schema to use for your installation.

Table 1-1 Deciding Which Schema to Use

Scenario	Use This Schema
You are installing Calendar Server 6.0 for the first time.	Sun ONE LDAP Schema, v.2 Native Mode
You plan to integrate Calendar Server 6.0 with other Java Enterprise System products such as Sun ONE Portal Server.	Sun ONE LDAP Schema, v.2 Native Mode
You plan to use either of these Sun ONE Identity Server 6.1 features: • CLI utilities for provisioning (for example, domains and users) • Single sign-on (SSO)	Sun ONE LDAP Schema, v.2 Native or Compatibility Mode
You are upgrading Calendar Server to 6.0 from a 5.x version?	Sun ONE LDAP Schema, v.2 Native Mode or Sun ONE LDAP Schema, v.2 Compatibility Mode, if you plan to use Identity Server 6.1 features or integrate Calendar Server with other Java Enterprise System products or Sun ONE LDAP Schema, v.1, if you don't plan to use Identity Server 6.1 features or integrate Calendar Server with other Java Enterprise System products
You want to use Calendar Server 6.0 csdomain utility for provisioning domains.	Sun ONE LDAP Schema, v.2 Native or Compatibility Mode or Sun ONE LDAP Schema, v.1 if you don't plan to use Identity Server 6.1 features or integrate Calendar Server with other Java Enterprise System products
You don't plan to use either the Identity Server 6.1 or Calendar Server 6.0 CLI utilities for provisioning because you have other tools you prefer to use.	Sun ONE LDAP Schema, v.2 for new Calendar Server 6.0 installations or Sun ONE LDAP Schema, v.1 or v.2 Compatibility Mode for existing Calendar Server installations

Gather Your Calendar Server 6.0 Configuration Information

Before configure Calendar Server 6.0 using, you should gather the following configuration information:

- Administration, User Preferences and Authentication Options
- Email and Email Alarms Options
- Runtime Configuration Options
- Database, Logs, and Temporary Files Directories

You will need this information when you run the Calendar Server configuration program, which is described in Chapter 2, "Configuring Calendar Server 6.0." However, you should determine this information before you run the Java Enterprise System installer to avoid conflicts (such as port numbers) with other component products.

To help you keep track of the configuration information, use the worksheets in Appendix A, "Configuration Worksheets".

Administration, User Preferences and Authentication Options

User Preferences Directory

Sun ONE Calendar Server requires a directory server for user authentication and for the storage and retrieval of user preferences.

Table 1-2 User Preferences Directory Options

Option	Description
LDAP Server Host Name	Host name of the LDAP directory server you are using for user authentication and user preferences. The default is the current host.
LDAP Server Port	Port number that the LDAP directory server listens on. The default is 389.
Base DN	Entry in the LDAP directory used as the starting point from which searches will occur. The default is `o=host.com`.
Directory Manager DN	User name that can make changes in the directory server schema. The default is `cn=Directory Manager`.
Directory Manager Password	Password of the Directory Manager DN. There is no default.

Calendar Server Administrator

The Calendar Server Administrator is the user account that can manage Calendar Server. For example, this account can run the Calendar Server administration utilities to perform functions such as starting and stopping Calendar Server, backing up the calendar database, enabling or disabling users, and so on. The Calendar Server Administrator user account must exist in your user authentication directory server.

Table 1-3 Calendar Server Administrator Options

Option	Description
Administrator User ID	User ID of the Calendar Server Administrator; must be a user in the above LDAP directory server. The default is `calmaster`.
Administrator Password	Password of the Calendar Server Administrator. There is no default.

Email and Email Alarms Options

You can configure Calendar Server to send an email alarm message to a Calendar Server Administrator in case a server problem occurs.

Table 1-4 Email and Email Alarms Options

Option	Description
Email Alarms	Enables or disables email alarms. The default is Enabled.
Administrator Email Address	Email address of the Calendar Server Administrator who will receive the email alarm messages.
SMTP Host Name	Host name of the SMTP server where Calendar Server sends the email alarm messages. The default is the current host.

Runtime Configuration Options

You can configure the following Calendar Server runtime and system resource options.

Table 1-5 Runtime Configuration Options

Option	Description
Service Port	Port number that Calendar Server listens on to provide Web (HTTP) access to users. The default is 80.
Maximum Sessions	Maximum number of Calendar Server sessions to allow concurrently. The default is 5000.
Maximum Threads	Maximum number of Calendar Server threads to allow concurrently. The default is 20.
Number of Server Processes	Maximum number of Calendar Server processes to run concurrently. The default is the number of CPUs on the server where you are installing Calendar Server.
Runtime User ID	UNIX user name under which Calendar Server will run. This user name should not be `root`. If the account does not exist, the configuration program will create it. The default is `icsuser`.
Runtime Group ID	UNIX group under which Calendar Server will run. If the group does not exist, the configuration program will create it. The default is `icsgroup`.

Calendar Server Startup

You can configure the following options to automatically start Calendar Server.

Table 1-6 Calendar Server Startup Options

Option	Description
Start after successful installation	Whether to start Calendar Server automatically after a successful installation. The default is checked.
Start on system startup	Whether to start Calendar Server automatically after a system startup. The default is checked.

Database, Logs, and Temporary Files Directories

Calendar Server creates and stores information in calendar database files, log files, and temporary files in specific directories.

Table 1-7 Database, Logs, and Temporary Files Directories Options

Option	Description
Database Directory	Directory where Calendar Server should create and store the calendar database (*.db) files. The default is: `var/opt/SUNWics5/csdb`
Logs Directory	Directory where Calendar Server writes log files. The default is: `var/opt/SUNWics5/logs`
Temporary Files Directory	Directory where the Calendar Server writes temporary files. The default is: `var/opt/SUNWics5/tmp`

Install Calendar Server Using the Java Enterprise System Installer

On Solaris systems, the Java Enterprise System installer installs the Sun component product packages, including Calendar Server 6.0, and the shared components that are used by the various products.

The Java Enterprise System installer installs Calendar Server 6.0 in the following directory:

cal_svr_base/opt/SUNWics5/cal

CAUTION If you have a previous version of Calendar Server 6.0 installed at your site, the Java Enterprise System installer will overwrite existing Calendar Server files, including any files you have customized.

If you have customized any files, such as XSL, XML, GIF, HTML, configuration (.conf), or time-zone files, back up those files before you run the installer.

For information about the Java Enterprise System installer, refer to the *Sun Java Enterprise System Installation Guide*.

Configure Calendar Server 6.0

After you install Calendar Server 6.0 using the Java Enterprise System installer, you must run the Calendar Server configuration program using the information from "Gather Your Calendar Server 6.0 Configuration Information" on page 18. If you filled out the worksheets in Appendix A, "Configuration Worksheets," use that information to supply values to the configuration program.

For information about the configuration program, see Chapter 2, "Configuring Calendar Server 6.0."

Uninstalling Calendar Server 6.0

To uninstall Calendar Server 6.0 (as well as other component products), you must use the Java Enterprise System uninstaller, which is located in the following directory:

`/var/sadm/prod/entsys/uninstall`

For information about running the uninstaller, refer to the *Sun Java Enterprise System Installation Guide*.

When you uninstall Calendar Server 6.0, the uninstaller stops all Calendar Server processes (if they are running) and then removes the SUNWics5 and SUNWica5 packages.

If you are uninstalling only Calendar Server 6.0, the uninstaller does not remove other installed packages or components because they might be shared by other products.

The uninstaller also generates a log file in the following directory:

`/var/sadm/install/logs/`

An example of a log file is:

`Java_Enterprise_System_uninstall.B`*timestamp*

where *timestamp* identities when the uninstaller was run.

Configuring Calendar Server 6.0

After you install Sun ONE Calendar Server using the Sun Java Enterprise System installer, you must configure Calendar Server as follows:

1. Run the Directory Server Setup Script (comm_dssetup.pl) to configure Sun ONE Directory Server 5.x (if the script has not already been run during the configuration of Messaging Server 6.0).
2. Run the Calendar Server Configuration Program (csconfigurator.sh) to configure your site's specific requirements.

Directory Server Setup Script (comm_dssetup.pl)

The Directory Server Setup Perl script (`comm_dssetup.pl`) configures Sun ONE Directory Server 5.x for Calendar Server 6.0 and Messaging Server 6.0. The `comm_dssetup.pl` script prepares the Directory Server by setting up new schema, index, and configuration data. This section describes:

- Functions of comm_dssetup.pl
- Requirements to Run comm_dssetup.pl
- Running comm_dssetup.pl

NOTE If you also installed Sun ONE Messaging Server 6.0 and have already run `comm_dssetup.pl`, you do not need to run the script again. However, if you install a new version of Messaging Server or Calendar Server, run the new version of `comm_dssetup.pl` to update the directory server schema and to add any new indexes.

Functions of comm_dssetup.pl

Specifically, `comm_dssetup.pl` allows you to specify these options:

- Directory Server 5.x installation directory path and instance you want to use for Calendar Server 6.0 and Messaging Server 6.0.
- Directory Manager Distinguished Name (DN) and password.
- Whether Directory Server 5.x will be used for users and groups. If yes, you must also specify the DC Tree base suffix and a User and Group base suffix for your Organization Tree.
- Whether to use Sun ONE LDAP Schema v.1 or v.2.
- DC Tree base suffix, if you have selected LDAP Schema v.1
- Whether to update your schema according to the version you have selected.
- Whether to add Directory Server indexes to improve the efficiency of directory searches.

Requirements to Run comm_dssetup.pl

The requirements to run `comm_dssetup.pl` include:

- Sun ONE Directory Server 5.x must be installed, configured, and running.
- You must run `comm_dssetup.pl` on the same server where Directory Server 5.x is running.
- To use LDAP Schema, v.2, Sun ONE Identity Server 6.1 must be installed and configured.
- You must run `comm_dssetup.pl` as superuser (`root`).
- If your directory server is split into separate directory instances, run `comm_dssetup.pl` on all instances.
- If you are running a replicated directory server, run `comm_dssetup.pl` against the master and replica (slave) directories.
- You must run `comm_dssetup.pl` before you run the Calendar Server 6.0 Configuration Program.

Running comm_dssetup.pl

To run `comm_dssetup.pl`, follow these steps:

1. On the server where Directory Server 5.x is installed, login as or become superuser (`root`).
2. Start Directory Server 5.x, if necessary.
3. If Calendar Server 6.0 is installed on this same server, change to the `/opt/SUNWics5/cal/sbin` directory.

   Or, if Calendar Server 6.0 is not installed on this server, you must copy the `dssetup.zip` file (which includes the `comm_dssetup.pl` script and supporting files) from the server where Messaging Server 6.0 is installed:

   a. On the server where Directory Server 5.x is installed, create a temporary directory. For example: `var/tmp`.

   a. Copy the *msg_svr_base*`/install/dssetup.zip` file from the server where Messaging Server 6.0 is installed to the temporary directory.

   b. In the temporary directory, unzip the `dssetup.zip` file.
4. Run the `comm_dssetup.pl` script in either Silent Mode or Interactive Mode. To run this script, Sun recommends using the version of Perl included with Directory Server 5.x:

   ds_svr_base`/bin/slapd/admin/bin/perl`

Silent Mode

To run `comm_dssetup.pl` in silent mode, use the following syntax. You must provide all of the required arguments, as described in Table 2-1.

```
perl comm_dssetup.pl
    -i yes|no
    -c DirectoryServerRoot -d DirectoryInstance
    -r DCTreeSuffix -u UserGroupSuffix
    -s yes|no -D "DirectoryManagerDN" -w DirectoryManagerPassword
    -b yes|no -t 1|1.5|2 -m yes|no
    [ -S PathtoSchemaFiles ]
```

Table 2-1 Directory Server Setup Script (comm_dssetup.pl) Options

Option	Description
`-i yes|no`	Answers the question: “Do you want to configure new indexes?”
	`yes`–Add new Directory Server indexes. For Calendar Server 6.0, `comm_dssetup.pl` adds indexes for the `icsCalendar` and `icsCalendarOwned` attributes.
	`no`–Do not add indexes.
`-c` *DirectoryServerRoot*	Directory Server root path name. For example: `/usr/sunone/servers`
`-d` *DirectoryInstance*	Directory Server instance subdirectory. For example: `slapd-varrius`
`-r` *DCTreeSuffix*	DC Tree root suffix. For example: `o=internet`
`-u` *UserGroupSuffix*	User and Group root suffix. For example: `o=isp`
`-s yes|no`	Answers the question: “Do you want to update the schema?”
	`yes`–Update the schema. You must have a `config` directory with the schema files.
	`no`–Do not update schema.
`-D` *DirectoryManagerDN*	Directory Manager Distinguished Name (DN). The value must be enclosed by double quotation marks (") to allow `comm_dssetup.pl` to interpret a value with a space correctly.
	For example: `"cn=Directory Manager"`
`-w` *DirectoryManagerPassword*	Directory Manager DN password.
`-b yes|no`	Answers the question: “Will this directory server be used for users and groups?”
	`yes`–Use this directory to store both configuration and user group data.
	`no`–Use this directory to store only configuration data.
`-t 1|1.5|2`	Sun ONE LDAP Schema version:
	• Option 1–ONE LDAP Schema v.1
	• Option 1.5–ONE LDAP Schema v.2 Compatibility Mode
	• Option 2–ONE LDAP Schema v.2 Native Mode
`-m yes|no`	Answers the question: “Do you want to modify the directory server?”
	yes–Modify the directory server without prompting the user.
	no–Do not modify the directory server without prompting the user.
`-S` *PathtoSchemaFiles*	Path to the directory where the schema files are located. For example: `./schema`

Example

```
perl comm_dssetup.pl -i yes -c /var/mps/serverroot -d slapd-ketu
-r o=internet -u o=usergroup" -s yes -D "cn=Directory Manager" -w password
-b yes -t 1 -m yes
```

When you run in silent mode, `comm_dssetup.pl` displays a summary similar to Step 11. Summary of Settings for Interactive Mode before making actual changes to your Directory Server.

Interactive Mode

To run in interactive mode, you run the `comm_dssetup.pl` script without any arguments and then enter your choices as you are prompted.

1. Welcome and Introduction

```
# perl comm_dssetup.pl

Welcome to the Directory Server preparation tool for Sun ONE Messaging
Server.
(Version 6.0 Revision 0.004)
This tool prepares your directory server for Sun ONE Messaging Server
install.
The logfile is /var/tmp/dssetup_YYYYMMDDHHSS
Do you want to continue [y]:
```

Press Enter to continue, or type no and then press Enter to exit.

NOTE Although the welcome message refers only to Sun ONE Messaging Server, `comm_dssetup.pl` also applies to Sun ONE Calendar Server.

2. Installation Root of Directory Server

```
Please enter the full path to the directory where the Sun ONE Directory
Server was installed.
Directory server root [/var/mps/serverroot]
```

Specify the location of the installation root of the Directory Server.

3. Directory Server Instance

```
Please select a directory server instance from the following list:
[1] slapd-varrius
Which instance do you want [1]:
```

If multiple instances of Directory Server reside on the server, choose the one that will be configured with Calendar Server.

4. Directory Manager Distinguished Name (DN)

```
Please enter the directory manager DN [cn=Directory Manager]:
Password:
```

The Directory Manager DN (cn=Directory Manager) is the administrator who is responsible for the user and group data in the Organization Tree. Be sure that the Directory Manager DN you specify in this script is the same DN that you set up for your Directory Server installation as well for as your Calendar Server configuration.

5. User and Group Directory Server

```
Will this directory server be used for users/groups [Yes]:
```

If you answer Yes, you must also specify a DC Tree base suffix and a User and Group base suffix for your Organization Tree.

If you answer No, it is assumed that this directory instance is used to store only configuration data, and you will skip to the question about updating schema files. After you finish running this script against the configuration directory instance, you need to run this script against the directory instance that stores user and group data before you configure Calendar Server.

6. User and Group Base Suffix

```
Please enter the Users/Groups base suffix [o=usergroup]:
```

The User and Group base suffix is the top entry in the Organization Tree that holds the name space for user and group entries. Be sure that the User and Group base suffix you select is the same as what you specified for Directory Server and Calendar Server.

If you installed Identity Server, be sure the suffix specified in Identity Server installation is the same as what you specify for this question. If you do not use the same suffix, Calendar Server will not recognize your Identity Server installation.

7. Schema Type

```
There are 3 possible schema types:
1 - schema 1 for systems with iMS 5.x data
1.5 - schema 2 compatibility for systems with iMS 5.x data that has been
converted with imsdirmig
2 - schema 2 native for systems using Identity Server
Please enter the Schema Type (1, 1.5, 2) [1]:
```

Choose the version of the schema you are planning to use:

- Option 1–Sun ONE LDAP Schema, v.1
- Option 1.5–Sun ONE LDAP Schema, v.2, Compatibility Mode.
- Option 2–Sun ONE LDAP Schema, v.2, Native Mode

To use Sun ONE LDAP Schema, v.2 (options 1.5 or 2) Sun ONE Identity Server 6.1 must be installed and configured; otherwise, `comm_dssetup.pl` will terminate. You must then rerun the script after Identity Server is installed.

8. Domain Component (DC) Tree Base Suffix

```
Please enter the DC Tree base suffix [o=internet]:
```

In Step 7. Schema Type, if you choose Option 1 or 1.5, you will be asked to provide your DC Tree Base Suffix. If you choose Option 2, you will not be asked this question.

The DC Tree mirrors the local DNS structure and is used by the system as an index to the Organization Tree that contain the user and group data entries. The DC Tree base suffix is the name of the top entry on the DC tree. You can either choose the default (`o=internet`) or another name.

9. Updating Schema Files

```
Do you want to update the schema files [yes]:
```

If you answer Yes, `comm_dssetup.pl` adds new elements to your schema. It is recommended that you update the directory with the new schema files each time you install a new version of Calendar Server or Messaging Server.

10. Configuring New Indexes

```
Do you want to configure new indexes [yes]:
```

If you answer Yes to Step 5. User and Group Directory Server, you will be asked if you want to configure new indexes, which can improve the performance of directory searches.

For Calendar Server 6.0, `comm_dssetup.pl` adds indexes for the `icsCalendar` and `icsCalendarOwned` attributes. It is recommended that you answer Yes.

11. Summary of Settings

Before `comm_dssetup.pl` updates the Directory Server configuration, it displays a summary of your settings and then asks if you want to continue.

```
Here is a summary of the settings that you chose:
  Server Root : /var/mps/serverroot/
  Server Instance : slapd-varrius
  Users/Groups Directory : Yes
  Update Schema : yes
  Schema Type : 1
  DC Root : o=internet
  User/Group Root : o=usergroup
  Add New Indexes : yes
  Schema Directory : ./schema **
  Directory Manager DN : cn=Directory Manager
Now ready to generate a shell script and ldif file to modify the Directory.
No changes to the Directory Server will be made this time.
Do you want to continue [y]:
```

If you chose Option 2 (Sun ONE LDAP Schema, v.2 - Native Mode) in Step 7. Schema Type, the DC Root will be the same value that you entered for the User/Group Root.

To change any of your settings, enter No and re-run the script.

If you continue, `comm_dssetup.pl` creates the following LDIF file and shell script to update the Directory Server indexes and schema:

`/var/tmp/dssetup_`*timestamp*`.ldif`
`/var/tmp/dssetup_`*timestamp*`.sh`

Again, `comm_dssetup.pl` asks if you want to continue. Enter Yes to run the `dssetup_`*timestamp*`.sh` script now or No to exit. If you exit, you will need to run the `/var/tmp/dssetup_`*timestamp*`.sh` script later.

Calendar Server Configuration Program (csconfigurator.sh)

The Calendar Server 6.0 configuration program (`csconfigurator.sh`) allows you to configure Calendar Server. The configuration program creates a new `ics.conf` configuration file in the following directory:

```
/etc/opt/SUNWics5/config
```

The configuration program performs minimal configuration. After you run the program, you might need to perform additional configuration, as described in the *Sun ONE Calendar Server Administrator's Guide*.

The configuration program uses a graphical user interface (GUI). If you run the program remotely, you must set your DISPLAY environment variable properly and allow X-Windows connections from the server to display on your computer. For example, to use the `xhost` utility, execute the following command on your computer: `xhost +`.

To run the configuration program, login as or become superuser (`root`) and change to the `/opt/SUNWics5/cal/sbin` directory. Then, issue this command:

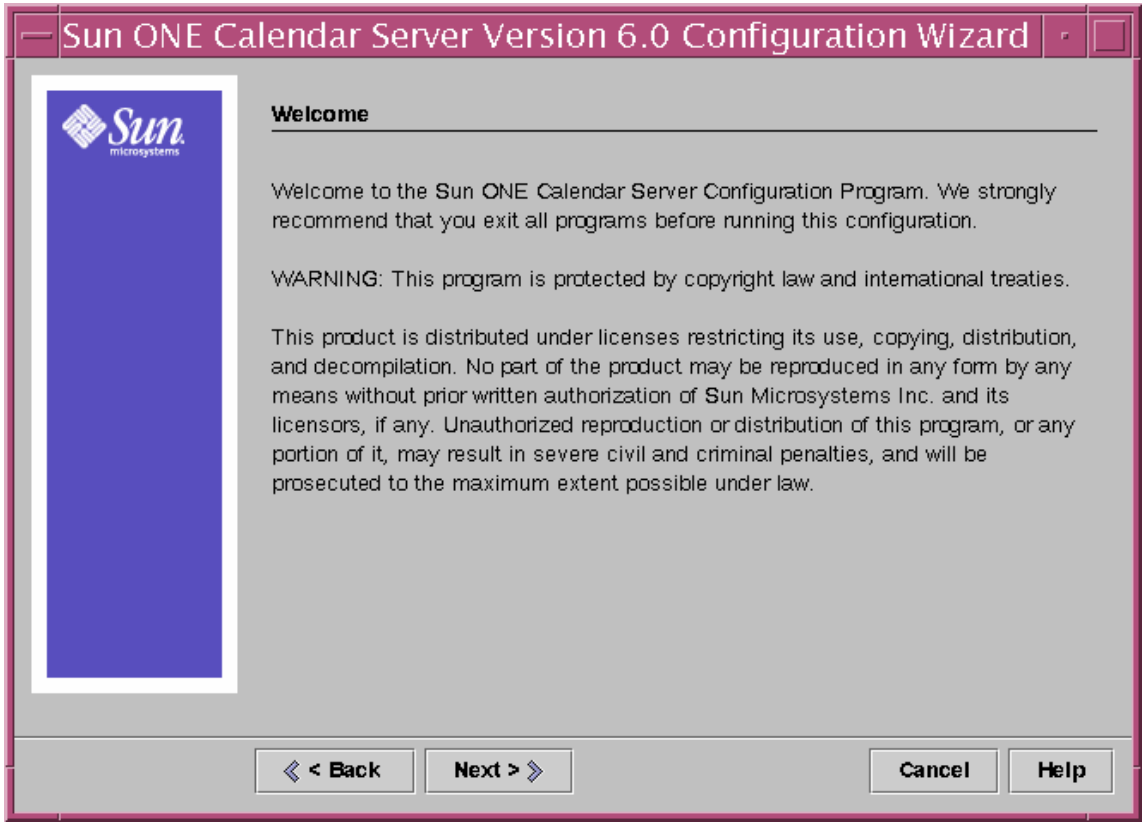
```
# sh ./csconfigurator.sh
```

The configuration program displays the following series of panels:

- Welcome Panel
- Administration, User Preferences and Authentication Panel
- Email and Email Alarms Panel
- Runtime Configuration Panel
- Select Directories Panel
- Ready to Configure Panel
- Configuration Summary Panel

Welcome Panel

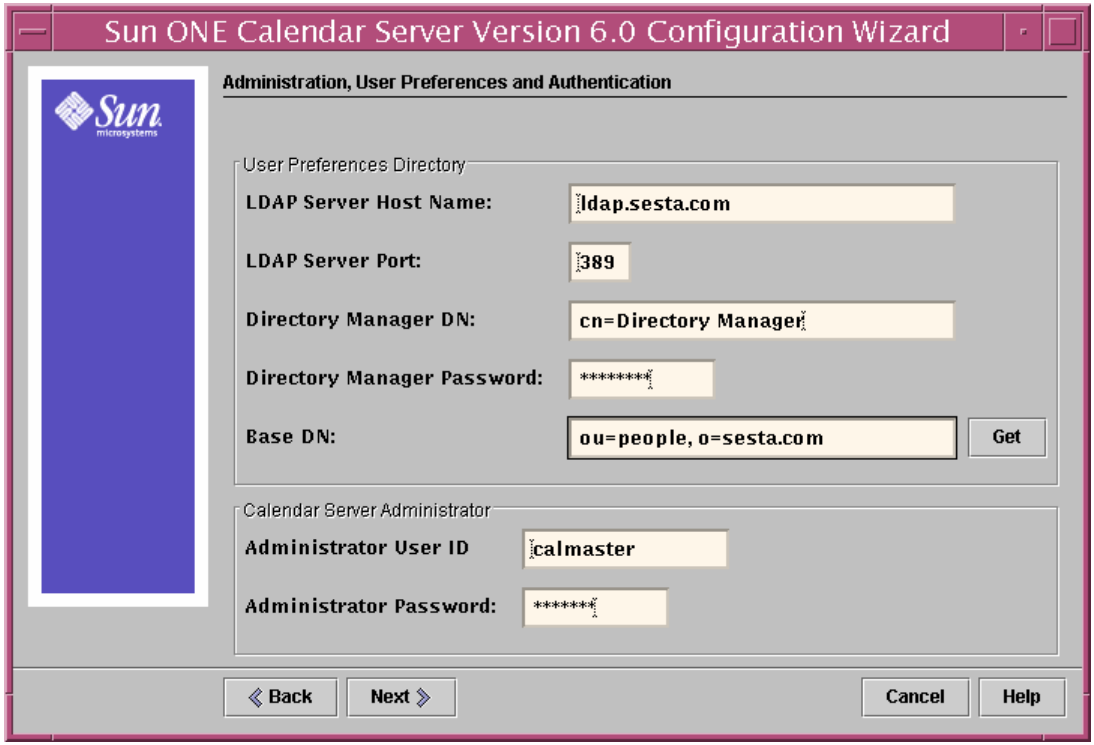
Figure 2-1 Calendar Server Configuration Program Welcome Panel



Click **`Next`** to continue or **`Cancel`** to exit.

Administration, User Preferences and Authentication Panel

Figure 2-2 Calendar Server Configuration Program Administration, User Preferences and Authentication Panel



User Preferences Directory Options

- **LDAP Server Host Name**–Host name of the LDAP directory server you are using for user authentication. Default: current host
- **LDAP Server Port**–Port number that the LDAP server listens on. Default: 389

- **Base DN**–Entry in the LDAP directory used as the starting point from which searches will occur. Default: o=*host*.com.

  To return the Base DN for the LDAP server, click **`Get`**.

  Note Before you click **`Get`**, you must enter the Directory Manager DN and password to authenticate to the directory server.

- **Directory Manager DN**–User name that can make changes in the directory server schema. Default: cn=Directory Manager.
- **Directory Manager Password**–Password of the Directory Manager DN. Default: None

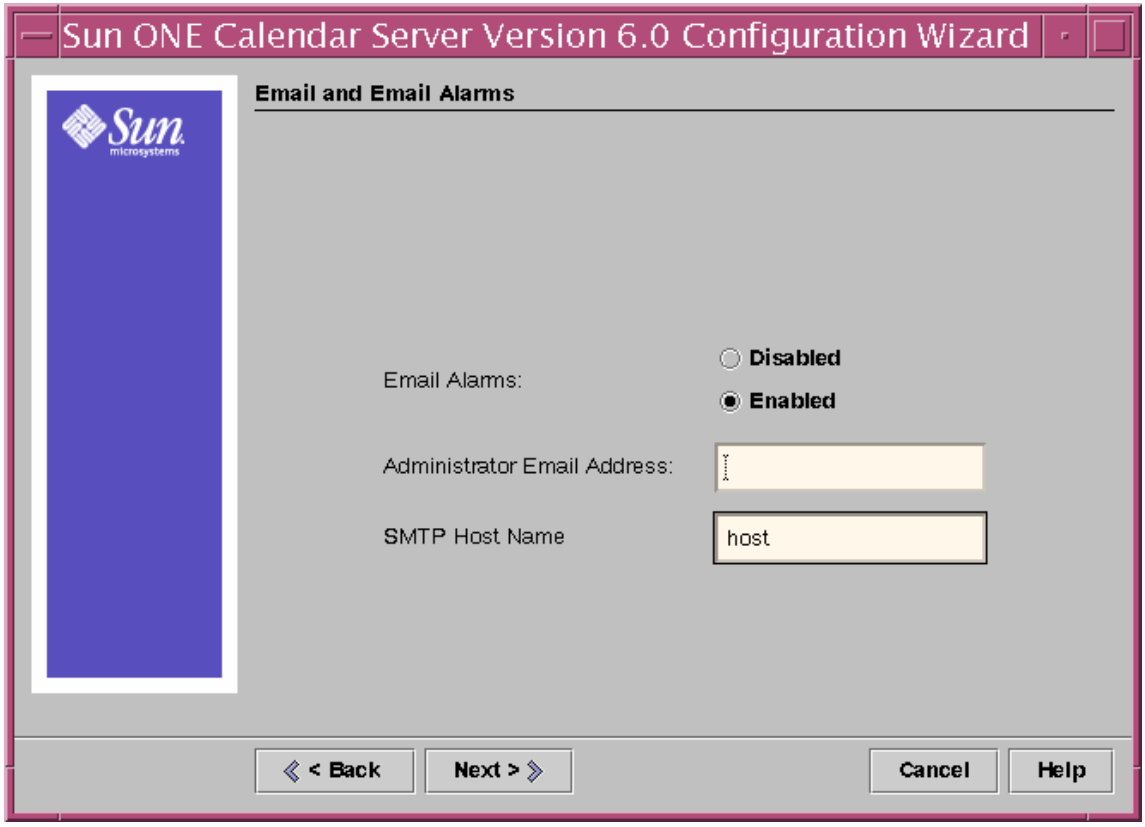
Calendar Server Administrator Options

- **Administrator User ID**–User ID of the Calendar Server Administrator; must be a user in the above LDAP directory server. Default: `calmaster`
- **Administrator Password**–Password of the Calendar Server Administrator. Default: None

Click **`Next`** to continue, **`Back`** to return to the previous panel, or **`Cancel`** to exit.

Email and Email Alarms Panel

Figure 2-3 Calendar Server Configuration Program Email and Email Alarms Panel

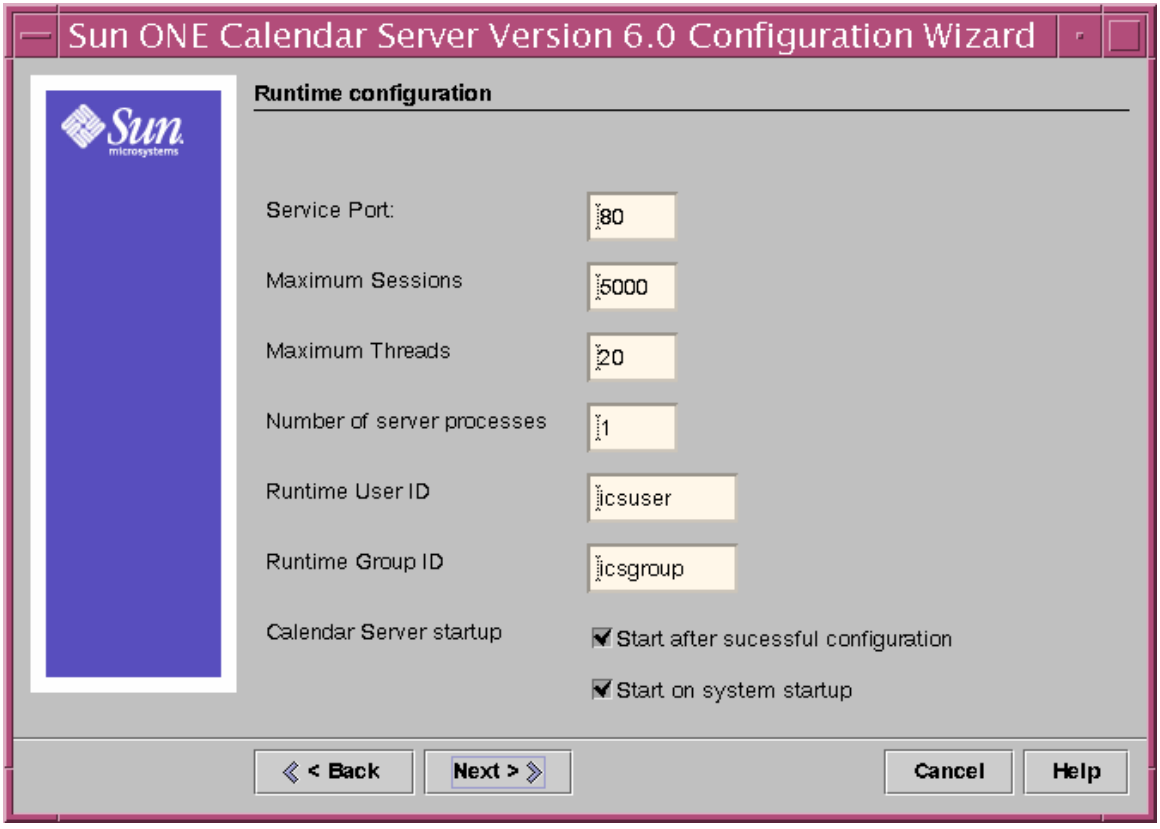


- **Email Alarms**–Specifies whether Calendar Server should send an email alarm message to a Calendar Server administrator in case a server problem occurs. Default: Enabled.
- **Administrator Email Address**–Email address of the Calendar Server Administrator who will receive the email alarm messages. Default: None.
- **SMTP Host Name**–Host name of the SMTP server where email alarm messages should be sent. Default: current host.

Click **`Next`** to continue, **`Back`** to return to the previous panel, or **`Cancel`** to exit.

Runtime Configuration Panel

Figure 2-4 Calendar Server Configuration Program Runtime Configuration Panel



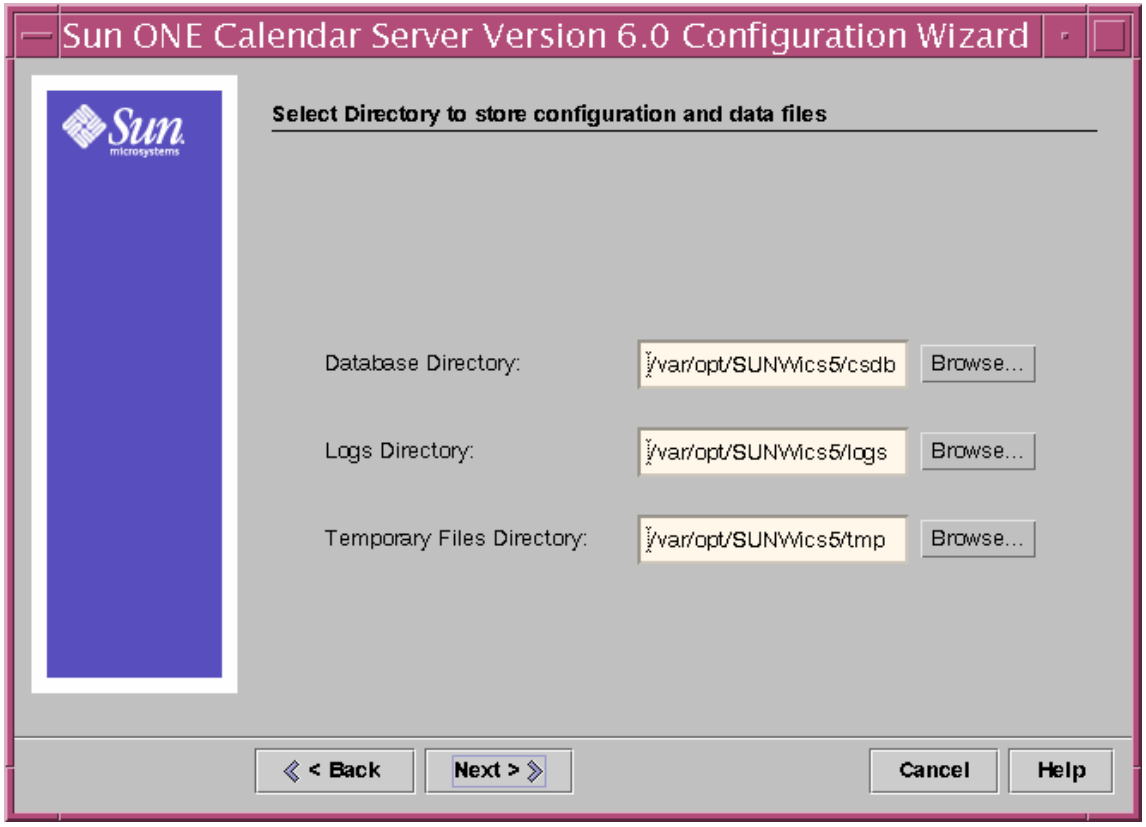
- **Service Port**–Port number that Calendar Server listens on to provide Web (HTTP) access to users. Default: 80.
- **Maximum Sessions**–Maximum number of concurrent Calendar Server sessions. Default: 5000
- **Maximum Threads**–Maximum number of concurrent Calendar Server threads. Default: 20

- **Number of Server Processes**–Maximum number of Calender Server processes to run on the server. Default: Number of CPUs on the server where you are installing Calendar Server
- **Runtime User ID**–UNIX user name under which Calendar Server will run. This name should not be `root`. If the account does not exist, the configuration program will create it. Default: `icsuser`
- **Runtime Group ID**–UNIX group under which Calendar Server will run. If the group does not exist, the configuration program will create it.
  Default: `icsgroup`
- **Calendar Server Startup Options**
  - **Start after successful installation**–Specifies whether to start Calendar Server automatically after a successful installation. Default: checked
  - **Start on system startup**–Specifies whether to start Calendar Server automatically after a system startup. Default: checked

Click **`Next`** to continue, **`Back`** to return to the previous panel, or **`Cancel`** to exit.

Select Directories Panel

Figure 2-5 Calendar Server Configuration Program Select Directories Panel

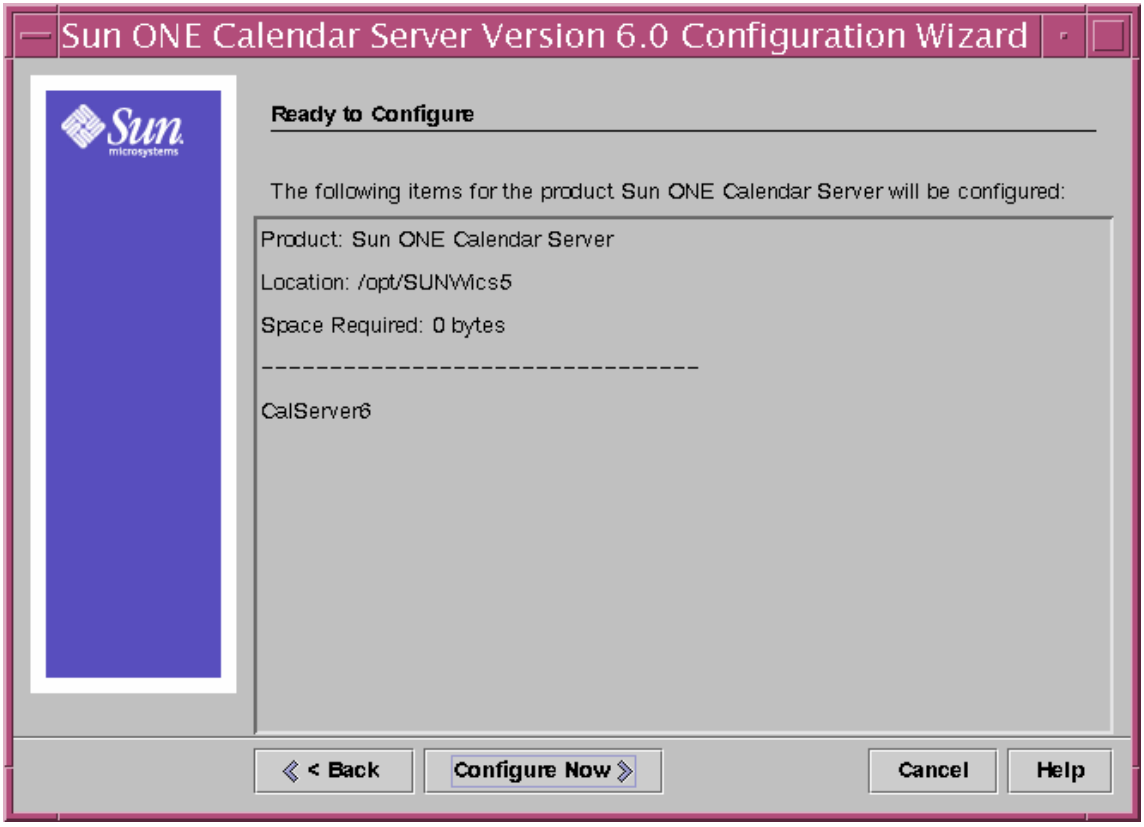


- **Database Directory**–Directory where Calendar Server should create and store the calendar database files. Default: `/var/opt/SUNWics5/csdb`
- **Logs Directory**–Directory where Calendar Server writes log files. Default: `/var/opt/SUNWics5/logs`
- **Temporary Files Directory**–Directory where the Calendar Server writes temporary files. Default: `/var/opt/SUNWics5/tmp`

For each directory, you can accept the default directory, type the full path of another directory, or click **`Browse...`** and choose a directory. Then, Click **`Next`** to continue, **`Back`** to return to the previous panel, or **`Cancel`** to exit.

Ready to Configure Panel

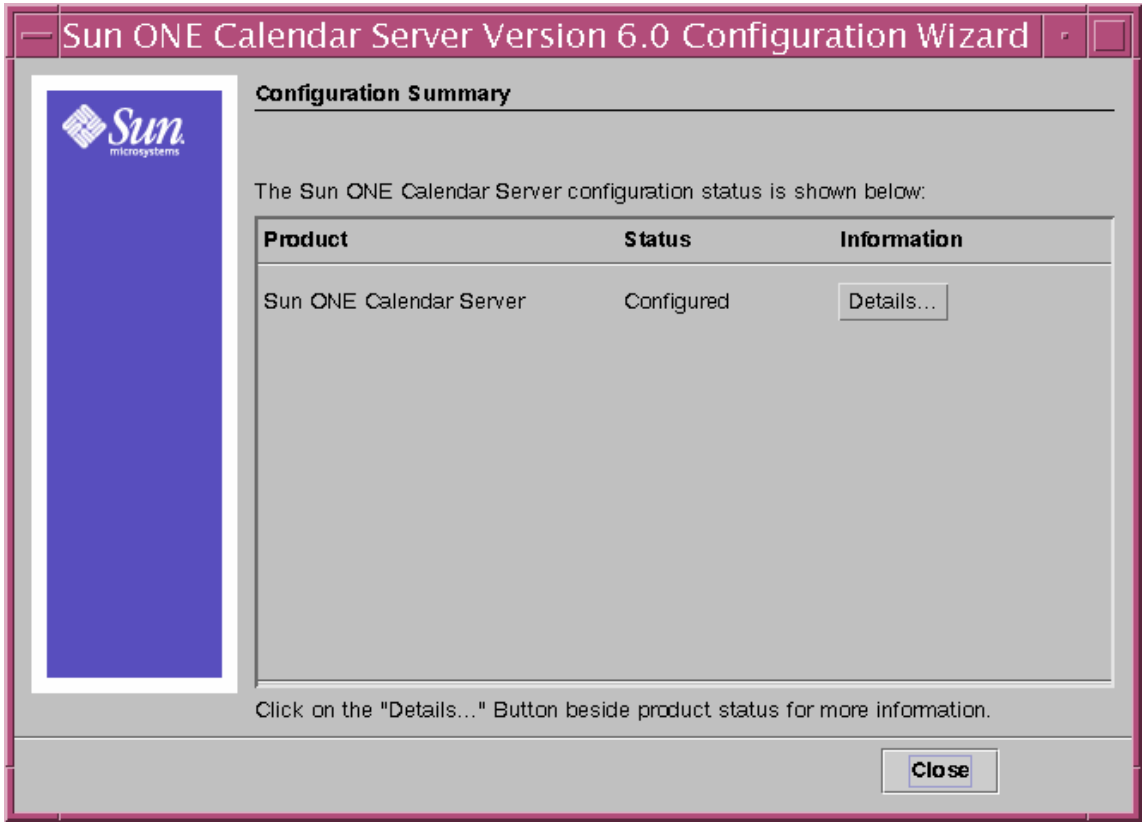
Figure 2-6 Calendar Server Configuration Program Ready to Configure Panel



Click **`Configure Now`** to configure Calendar Server, **`Back`** to return to the previous panel, or **`Cancel`** to exit.

Configuration Summary Panel

Figure 2-7 Calendar Server Configuration Program Summary Panel



Click **`Details...`** to view the details of the configuration summary or **`Close`** to exit the configuration program.

Chapter 3

Migrating Calendar Server Data

Sun™ ONE Calendar Server 6.0 provides the following migration utilities:

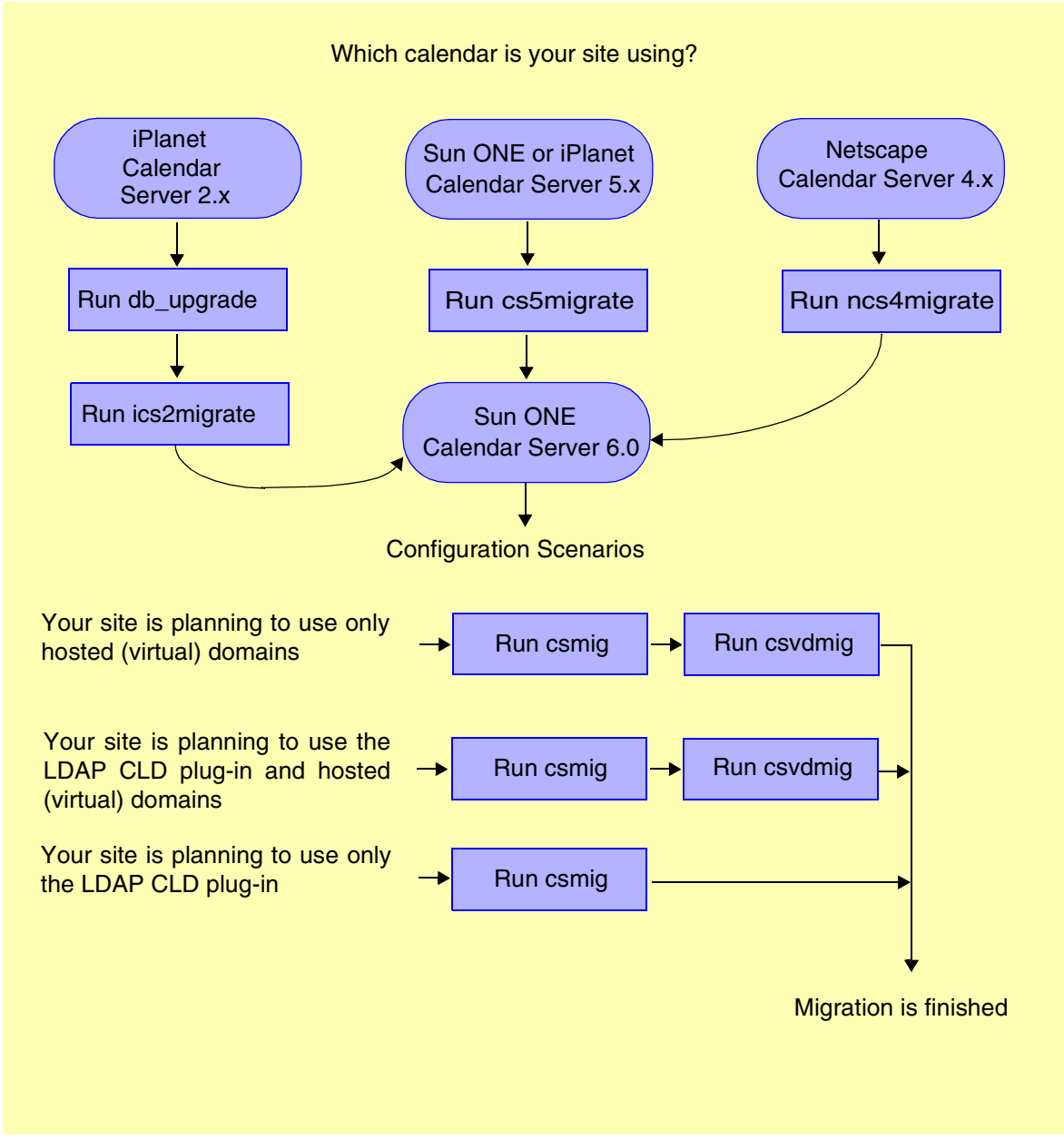
- cs5migrate Utility – Migrates a Calendar Server 5.x database to Calendar Server 6.0 and upgrades the calendar database from Berkeley DB version 2.6 to version 3.2.9.
- csmig Utility – Assigns an owner to each calendar in the calendar database and maps each calendar ID (`calid`) to an owner, if needed, which allows support for hosted (virtual) domains and the LDAP Calendar Lookup Database (CLD) plug-in.
- csvdmig Utility – Upgrades a Calendar Server 6.0 site to use hosted (virtual) domains.
- ics2migrate Utility – Migrates data from iPlanet Calendar Server 2.x.
- ncs4migrate Utility – Migrates data from Netscape Calendar Server 4.x.
- csrename Utility – Renames calendar users in the calendar database and in the LDAP directory server (Calendar Server attributes with the “ics” prefix).

Figure 3-1 shows a roadmap for running the Calendar Server migration utilities.

CAUTION Before you run a migration utility, it is very important to first check with your Sun Microsystems technical support or sales account representative to ensure that you have the latest version of the utility.

If your site is configured for limited virtual domain mode or multiple instances of Calendar Server, contact your Sun Microsystems sales account representative for an evaluation of your migration requirements and to ensure that you have the specific migration utility that supports those requirements.

Figure 3-1 Roadmap for Running the Calendar Server Migration Utilities



cs5migrate Utility

If you are upgrading from Calendar Server 5.x to Calendar Server 6.0, you must run the `cs5migrate` utility before you can run Calendar Server 6.0. The `cs5migrate` utility performs these functions:

- Migrates the following Calendar Server 5.x files to Calendar Server 6.0:
  - Calendar database files (`ics50calprops.db`, `ics50journals.db`, `ics50alarms.db`, `ics50events.db`, `ics50todos.db`, and `ics50gse.db`)
  - Session database (`session.db`)
- Upgrades the calendar database from Berkeley DB version 2.6 to version 3.2.9.
- Writes the migration status to `cs5migrate.log`. and any errors to the `cs5migrateError.log` and `cs5migrateException.log`.

Migration Time

The `cs5migrate` migration time can vary, depending on several factors. First, `cs5migrate` must access the LDAP directory server to update the schema attributes, so the network connection to the LDAP server can greatly affect the migration time. If possible, run `cs5migrate` with a fast network connection to the LDAP server and when other network traffic is minimal.

Migration Scenario–On a Sun Fire™, UltraSPARC™ III Cu, 12 CPUs, 750 MHz, 12 GB memory, floating point processor, running Solaris 8 OS with 20 GB swap file space, `cs5migrate` migrated the following Calendar Server 5.x calendar database in approximately 1 hour and 15 minutes:

- Calendar database size: approximately 600 MB.
- Number of calendars: 8726
- Number of events: 272412
- Number of tasks: 4490
- Number of alarms: 13583
- Number of Group Scheduling Engine (GSE) entries: 0

cs5migrate Syntax

The `cs5migrate` utility has the following syntax:

```
cs5migrate [-q] [-d] [-r] [-l min|max] source-directory target-directory
```

`-q` specifies quiet mode. `cs5migrate` does not display information if the migration is successful. Any errors, however, are displayed.

`-d` specifies dry run mode. A dry run reports what `cs5migrate` would do during an actual migration, but `cs5migrate` does not migrate any data or upgrade the database.

`-r` specifies to create the master component for recurring events.

`-l min|max` specifes the log mode and level of detail for the migration log (`cs5migrate.log`).

Note The `-t` option is not implemented in the current release.

source-directory is a required parameter that specifies the directory that contains the Calendar Server 5.x database files.

target-directory is a required parameter that specifies an existing directory where `cs5migrate` should create the new Calendar Server 6.0 database files.

Important You must create the *target-directory* before you run `cs5migrate`.

Migration Process

Before you run `cs5migrate`, perform these steps:

- Backup your Calendar Server 5.x database using a utility such as `csbackup`, the Sun StorEdge Enterprise Backup™ software, or Legato Networker®.
- It is also recommended that you rebuild your calendar database using the `csbd rebuild` command before you migrate. For information see, Chapter 5, "Managing Calendar Server Databases," in the *Sun ONE Calendar Server Administrator's Guide*.
- If necessary, enable alarms by setting the `caldb.serveralarms` parameter in the `ics.conf` file to "yes".

- If you need to move your Calendar Server 5.x database to another server, you can simply copy the database (*.db) files to the new server, if he files are not too large. Otherwise, create a tar file of the database files, copy the tar file to the new server, and then untar it.

To run `cs5migrate`, follow these steps:

1. On Solaris and other UNIX systems, login as the user and group under which Calendar Server is running, such as `icsgroup` and `icsuser`.
2. If necessary, stop Calendar Server using the `stop-cal` command.
3. If necessary, create the *target-directory*. The *target-directory* must exist before you run `cs5migrate`.
4. Run `cs5migrate`. For the syntax, see cs5migrate Syntax.

   For example, on a Solaris system:

   ```
   ./cs5migrate -q -l max /var/opt/SUNWics5/csdb511
     /var/opt/SUNWics5/csdb60
   ```

   In this example, the `/var/opt/SUNWics5/csdb60` directory must exist before you migrate.

   For the migration status, view the `cs5migrate.log` file. If errors occur during the migration or if calendar database entries cannot be migrated, `cs5migrate` writes them to `cs5migrateerror.log`.
5. After `cs5migrate` is finished, the `caldb.berkeleydb.homedir.path` parameter in the `ics.conf` file must point to the migrated database because `cs5migrate` does not modify the `ics.conf` file.

   Either reset this parameter to point to the migrated database directory, or move the migrated database files to the directory indicated by the parameter.
6. If you are using the LDAP data cache option (`local.ldap.cache.enable = "yes"`) or the CLD cache option (`caldb.cld.cache.enable = "yes"`), create the `ldap_cache` and `cld_cache` directories in the target directory after you run `cs5migrate`.
7. Check the permissions for the migrated database files. If you ran `cs5migrate` as `icsuser`, you shouldn't have any access problems. If you ran as superuser (`root`), which is not recommended, you might need to reset the permissions.
8. Restart Calendar Server using the `start-cal` command.

csmig Utility

The `csmig` utility assigns an owner to each calendar in the calendar database and maps each calendar ID (`calid`) to an owner, if needed.

The `csmig` utility supports hosted (virtual) domains and the LDAP Calendar Lookup Database (CLD) plug-in. Calendars in the migrated database are then accessible using this plug-in. The LDAP CLD plug-in provides horizontal scalability of the calendar database by allowing calendars to be distributed over a number of back-end servers. For information about the LDAP CLD plug-in, see the *Sun ONE Calendar Server Administrator's Guide*.

This document describes these topics:

csmig Functions

The `csmig` migration utility performs these functions:

- `csmig` migrates both user and resource calendars in the current calendar database (*.db files) specified by the `caldb.berkeleydb.homedir.path` parameter. In the new destination target database, `csmig` updates entries required by the LDAP CLD plug-in in the calendar properties (calprops), events, todos (tasks), and group scheduling engine (gse) database files.

  `csmig` writes only to the destination target database; it does does not write to your existing calendar database.

- `csmig` updates LDAP attributes for all relevant LDAP entries, including `icsSubscribed`, `icsCalendar`, `icsCalendarOwned`, `icsFreeBusy`, `icsSet`, and `uid` (for resource calendars) . `csmig` creates the `icsDWPHost` attribute for each calendar in the LDAP directory server database. `icsDWPHost` specifies the host name of the back-end server where a calendar resides.

- csmig assigns an owner to each calendar in the calendar database and maps each calendar ID (calid) to an owner, if needed. All default calids are kept as is, and no changes are made. Other calendars are mapped as follows:
  - User calendars that don't have valid owners will be owned by the user passed to csmig by the -c option. For example, if jsmith doesn't have an owner, it will be converted to orphan:jsmith, if orphan is specified as the -c option.
  - Resource calendars that don't have an owner will be owned by the resource user passed to csmig by the -r option.
  - If a resource calendar has any colons in the name, the colons are converted to underscores.

  For example, a calendar named football with owner bkamdar will be converted to bkamdar:football. A calendar tchang:soccer with the owner bkamdar will be converted to bkamdar:tchang_soccer. (Only one colon should be in the calid.) A resource calendar named auditorium:room1 will be converted to auditorium_room1.

csmig Requirements

The requirements for using csmig are:

- The calendar database must not be corrupted. Use the csdb check command to check your calendar database, and if necessary, run the csdb rebuild command to rebuild the database. For information about these commands, see the Sun ONE *Calendar Server Administrator's Guide*.
- You must have sufficient disk space for the new destination target database and if applicable, your backup database.
- To run csmig, log in as icsuser (or as the Calendar Server runtime user ID specified during configuration). If you run csmig as superuser (root), you might need to reset the permissions for the migrated files.

  You must also have privileges to manage the attributes of calendar users in the LDAP directory server that stores user preferences.
- Calendar Server must be stopped.

csmig Syntax

The `csmig` utility has the following syntax:

```
csmig [ -t DestinationDB ] [ -b Backend-DWPHost ]
      [ -o OutputFile ]  [ -e ErrorFile ] [ -m MappingFile ]
        -c calendarOwner -r resourceOwner { migrate|dryrun }
```

`-t` *DestinationDB* specifies the destination target database that `csmig` generates. The default is `MigratedDB`.

`-b` *Backend-DWPHost* specifies the name of the DWP back-end host server. This name must match the DWP back-end host server name specified in the `ics.conf` file.

`-o` *OutputFile* specifies an output file that captures the `csmig` output to the screen as well as any errors that occur. The default is `MigrateOut`.

`-e` *ErrorFile* is the file where `csmig` writes any errors or database entries that cannot be resolved. If database entries cannot be resolved, they are not written to the destination database. The default is `MigrateError`.

`-m` *MappingFile* is an output mapping file generated in `dryrun` mode that list recommended changes for updating entries in the LDAP schema. For example:

```
Old calid = jsmith New calid = jsmith:basketball
```

The mapping file provides only a list of recommended changes to the LDAP schema. `csmig` does not actually make the changes to the schema.

In `migrate` mode, *MappingFile* is not used.

`-c` *calendarOwner* specifies the owner for user calendars that don't have owners.

`-r` *resourceOwner* specifies the owner for resource calendars that don't have owners.

csmig Migration Steps

After you have installed Calendar Server 6.0 on all of the servers in your configuration, you must run `csmig` to migrate your existing Calendar Server and LDAP data to the new Calendar Server 6.0 and LDAP data, which is required for the LDAP CLD plug-in to work properly. Here are the recommended steps to migrate calendar data using `csmig`:

1. Configure Your LDAP Directory Server–Adding indexes can greatly improve the performance of your migration and calendar searches on LDAP data.
2. Perform a Test Dry Run–A dry run reports what csmig would do during a migration, but it does not migrate any actual data. After the dry run, you can correct any errors and determine a plan to handle any unresolved calendars.
3. Migrate Your Production Data–During a production run, `csmig` migrates the calendar database (.db files) and LDAP data (user and group preferences data), `icsSubscribed`, `icsCalendar`, `icsCalendarOwned`, `icsFreeBusy`, `icsSet`, and `uid` (for resource calendars). After the migration, all calendar resources will have an LDAP entry created.

Configure Your LDAP Directory Server

To improve performance, consider adding the following two new indexes to the `slapd.ldbm.conf` file:

- `index icscalendar pres,eq,sub`–Used by the migration process for searching the `icsCalendar` attribute.
- `index icscalendarowned pres,eq,sub`–Not required for the migration process but is used to perform a calendar search on LDAP data (for a subscribe operation) when the LDAP CLD plug-in is enabled.

For information about creating indexes in the `slapd.ldbm.conf` file, refer to your directory server documentation.

Perform a Test Dry Run

A test dry run performed on a staging server reports what would be migrated, but it does not perform the actual migration of your production database. A dry run allows you to determine a plan for migrating your production database. For example, you can decide how you want to handle “orphan” calendars, which are calendars that don’t have an owner.

To perform a test dry run using `csmig`, follow these steps:

1. Log in as `icsuser` (or as the Calendar Server runtime user ID specified during configuration). If you run `csmig` as superuser (root), you might need to reset the permissions for the migrated files.
2. Install Calendar Server 6.0 (if necessary) on the staging server.
3. Copy a snapshot of your calendar database to the staging server.
4. Install an LDAP server to mimic the production LDAP environment. Install a snapshot of the LDAP database on this server with the new indexes in the `slapd.ldbm.conf` file.
5. Change to the *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory.
6. Consider creating a catchall `calid` for user calendars that don't have an owner. For example, on Solaris systems, the following command creates a user with the `calid` of `orphan`:

   ```
   ./csuser -g orphan -s adminuser -y password -l en -c orphan create orphan
   ```

7. Stop the Calendar Server using the `stop-cal` command (if necessary).
8. Run the `csdb check` command to check your database for corruption. If corruption is indicated, run `csdb rebuild` to rebuild the database.
9. Run `csmig` with the `dryrun` option. For example, on a Solaris system, enter:

   ```
   ./csmig -b sesta.com -o csmig.out -e csmig.errors -m csmig.map
   -c orphan -r calmaster dryrun
   ```

   This command assigns user calendars without an owner to `orphan` and resource calendars without an owner to `calmaster`.

   Check the output mapping file (`csmig.map`). The mapping file lists recommended changes for updating entries in the LDAP schema.
10. Check the output, mapping, and error files. Resolve any LDAP issues or errors that you find. Determine how you will handle any unresolved calendars before the actual migration. Several options are:
    - Delete any unneeded calendars before you migrate.
    - Assign owners to any unresolved calendars.
    - Allow `csmig` to assign owners to the calendars during migration using the `-c` and `-r` options.

11. It is highly recommended that you migrate your calendar database on your staging server before you migrate your actual production calendar database. This steps allows you to see exactly how your data will be migrated and to correct any problems before you migrate your production database.

    For example, on a Solaris system, the following command migrates the calendar database to the `/var/opt/SUNWics5/testcsdb/` directory:

    ```
    ./csmig -t /var/opt/SUNWics5/testcsdb/ -b sesta.com -o csmig.out
    -e csmig.errors -m csmig.map -c orphan -r calmaster migrate
    ```

12. After the test migration is finished, copy the migrated database to the `/csdb` directory specified by the `caldb.berkeleydb.homedir.path` parameter. Or, edit this parameter to point to the new location of the migrated database. Then perform these checks:

    - Run `csdb check` on the new calendar database. The number events and todos in the migrated database should match the pre-migration totals.
    - Search for `icsCalendarOwned` entries and make sure that the entries match the pre-migration number of calendars.
    - Login to Calendar Express and verify some of the calendars in the migrated database.

If the test migration is successful, you are ready to migrate your production database.

Migrate Your Production Data

To migrate your production database using `csmig`, follow these steps:

1. Log in as `icsuser` (or as the Calendar Server runtime user ID specified during configuration). If you run `csmig` as superuser (`root`), you might need to reset the permissions for the migrated files.
2. Change to the *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory.
3. Stop the Calendar Server using the `stop-cal` command (if necessary).
4. Backup the following data:
    - Calendar database (`.db` files).
    - LDAP data: `slapd` database directory and LDAP database.
    - `ics.conf` file. This step is not actually required, but it can be useful if you need to revert to your original configuration.

5. Run `csmig` with the `migrate` option. For example, on a Solaris system, the following command migrates the calendar database to the `/var/opt/SUNWics5/newcsdb/` directory:

   ```
   ./csmig -t /var/opt/SUNWics5/newcsdb/ -b sesta.com -o csmig.out
   -e csmig.errors -m csmig.log -c orphan -r calmaster migrate
   ```

6. Check for any unresolved calendars in the error file and resolve them according to your plan from Step 10 under Perform a Test Dry Run.

7. Copy the new migrated database to the `/csdb` directory specified by the `caldb.berkeleydb.homedir.path` parameter. Or, edit this parameter to point to the new location of the migrated database.

8. Run the `csdb check` command to check your migrated database. If any corruption is indicated, run `csdb rebuild` to rebuild the database.

9. Enable the LDAP CLD plug-in by making any necessary changes to the following configuration parameters in the `ics.conf` file:

   - `service.dwp.enable = "yes"`
   - `service.dwp.port = "9779"`
   - `csapi.plugin.calendarlookup = "y"`
   - `csapi.plugin.calendarlookup.name = "*"`
   - `caldb.cld.type = "directory"`
   - `caldb.dwp.server.default = "`*default-server-name*`"`
   - `caldb.dwp.server.`*server-hostname*`.ip = "`*server-hostname*`"` (for each back-end server including the local server)
   - `caldb.cld.cache.enable = "yes"` (if you want to use the CLD cache option)
   - `caldb.cld.cache.homedir.path` specifies the location of the CLD cache directory. The default is *cal_svr_base*`/var/opt/SUNWics5/csdb/cld_cache`.

     Check that this directory is correct, or if you want a different location for the CLD cache, modify this parameter.

   For information about setting configuration parameters for the LDAP CLD plug-in, see the *Sun ONE Calendar Server Administrator's Guide*.

10. Restart the Calendar Server using the `start-cal` command.

11. Log in to the Calendar Server and verify that your configuration is working by checking several of the migrated calendars. To disable alarms while you are making your checks, set each of the following parameters in the `ics.conf` file to "no":

    - `caldb.serveralarms = "no"`
    - `caldb.serveralarms.dispatch = "no"`
    - `service.ens.enable = "no"`
    - `service.notify.enable = "no"`
    - `ine.cancellation.enable  = "no"`
    - `ine.invitation.enable = "no"`
    - `service.admin.alarm = "no"`

csmig Tips and Troubleshooting

The section describes the following tips and trouble shooting solutions:

- The csmig dry run calendar owner is not the owner I want for a calendar
- The LDAP calendar search doesn't work correctly
- The csmig dry run indicates duplicate calendar names
- How do I assign orphan calendars to different owners?
- How do I move calendar users to another back-end server?

The csmig dry run calendar owner is not the owner I want for a calendar

For example, a calendar named `tchang:myCalendar` has the owner as `jsmith` in the calendar database, and the `csmig` dry run shows the mapping as `jsmith:tchang_myCalendar`. I would like to keep this calendar name as `tchang:myCalendar` and assign the owner as `tchang`.

Solution

Before the migration, use the `cscal` utility to change the owner of the calendar `tchang:myCalendar` to `tchang`. Once this is done, the migration will map this calendar to `tchang:myCalendar` and add `icsCalendarowned` to tchang's LDAP entry.

The LDAP calendar search doesn't work correctly

After migration, the LDAP calendar search is enabled, but the calendar search dialog does not return any results or returns only partial results.

Solution

Enabling the LDAP calendar search allows Calendar Server to search `(&(objectclass=icscalendaruser)(icscalendarowned=*substr*))`.

Manually run two different searches on the LDAP data with the following filters and compare the output:

- ldapsearch with filter `(&(objectclass=icscalendaruser)(icscalendarowned=*substr*))`
- ldapsearch with filter `(icscalendarowned=*substr*)`

Since the server uses the filter that includes `icsCalendaruser` objectclass, the LDAP server might have been deployed with the schema check disabled, and some calendar entries may have been provisioned without the `icsCalendaruser` objectclass.

The csmig dry run indicates duplicate calendar names

The `csmig` dry run mapping file and output file indicate that there is a duplicate calendar name. For example, in the original database, `jsmith` owns the following calendars:

- `basketball` with 5 events
- `jsmith:basketball` with 10 events

The dry run indicates that during a migration, the two calendars will be merged, and the resulting calendar will be

- `jsmith:basketball` with owner `jsmith` and 15 total events

The output file will include the following warning message:

```
Error modifying calendar properties, error=2
```

Solution

If you don't want the two calendars to be merged, change the owner of `basketball` to a user other than `jsmith` before the migration. This will preserve the data integrity of the two separate calendars.

How do I assign orphan calendars to different owners?

By default `csmig` assigns all orphan calendars to a single owner, but I would like to assign different owners for some orphan calendars.

Solution

`csmig` doesn't accept the mapping file in the command line. However, you can assign owners to the orphan calendars in the original database before the migration. Check the dry run mapping file for all orphan calendars. Then use the `cscal` utility to assign owners to the orphan calendars before the migration. Run csmig in `dryrun` mode again to verify the new owners.

How do I move calendar users to another back-end server?

How do I move users from one back-end server to another?

Solution

To move a calendar user, you export each of the user's calendars on the original server and then import the calendars on the second server. After the calendars are moved, you can delete the calendars on the original server. For detailed steps about moving users, see the *Sun ONE Calendar Server Administrator's Guide*.

csvdmig Utility

The `csvdmig` utility modifies the Calendar Server database and LDAP directory server database for sites that want to use hosted (virtual) domains. The `csvdmig` utility adds the domain name to the user ID as follows:

- The format of calendar IDs (calids) is changed as:

  `userid[:calendar-name]` to `userid@domain[:calendar-name]`

- Access Control List (ACL) access rules are changed as:

  `userid` to `userid@domain`

- The LDAP directory server user entries for the Calendar Server attributes are modified as:

  `userid[:calendar-name]` to `userid@domain[:calendar-name]`.

CAUTION The `csvdmig` utility does not actually migrate data from one location to another location. It modifies the calendar database and LDAP directory server in their current locations

Therefore, before you run `csvdmig`, backup both your Calendar Server database and LDAP directory server database.

csvdmig Syntax

The `csvdmig` utility has the following syntax:

```
csvdmig [-t DestinationDB] [-c ConfigFile] [-e ErrorFile] [-m MappingFile]
  migrate [DB | LDAP]
```

`-m` *MappingFile* is an input parameter that specifies a mapping file. The default is `MigrateMapping`.

The mapping file is an input text file that maps existing users to their respective domains. You must create the mapping file before you run `csvdmig`. Specify one entry per line with a space between the old and new values. For example:

```
user1 user1@sesta.com
user2 user2@siroe.com
user3 user3@sesta.com
...
user-n user-n@siroe.com
```

`-c` *ConfigFile* is an input parameter that specifies a Calendar Server configuration file. The default is the `ics.conf` file.

`-t` *DestinationDB* is an output parameter that specifies the location of the migrated database. The default is `MigratedDB`.

`-e` *ErrorFile* is an output parameter that specifies the name of the error file for errors that cannot be resolved. The default is `MigrateError`.

`DB | LDAP` specifies whether to modify the Calendar Server database (DB) or the LDAP directory server (LDAP). The default is the calendar database (DB).

csvdmig Examples

- Migrate the LDAP directory server data using default values:

  ```
  csvdmig migrate LDAP
  ```

- Migrate the Calendar Server database:

  ```
  csvdmig -t targetDB -e errorFile -m mappingFile migrate
  ```

ics2migrate Utility

The `ics2migrate` migration utility migrate iPlanet Calendar Server 2.x calendar data and LDAP user preferences to Calendar Server 6.0.

This section describes:

- Migration Requirements
- What Gets Migrated?
- Migration Process
  1. Upgrade the 2.x Calendar Database
  2. Migrate the Data
  3. Check the Migration Results
- Migration Examples

Migration Requirements

Calendar Server 2.x to 6.0 migration requires the following hardware and software:

- The source machine has the Calendar Server 2.x data that you plan to migrate.
- The target machine is where the migrated data will be created. This machine must have Calendar Server 6.0 (or later) installed.
- `ics2migrate` utility–Before you migrate, first check with your Sun technical support representative or account manager to ensure that you have the latest version of the utility.

The source machine and destination machines can be different servers or the same server. For a list of supported platforms refer to the Sun ONE Calendar Server Release Notes.

What Gets Migrated?

The following table lists the Calendar Server 2.x data and describes how `ics2migrate` migrates the data to Calendar Server 6.0.

Table 3-1 Migration of Calendar Server 2.x Data

Calendar Server 2.x Data	Migration Results for Calendar Server 6.0
Calendar Properties (`calprops`)	Updates the Calendar Server `calprops` database.
Events	Updates the Calendar Server `events` database.
Todos	Updates the Calendar Server `todos` database
Alarms	Updates the `alarms` database while writing events and todos.

The following table lists the Calendar Server 2.x LDAP attributes and describes how `ics2migrate` migrates the attributes to Calendar Server 6.0.

Table 3-2 Migration of LDAP Attributes

Calendar Server 2.x LDAP Attribute	Calendar Server 6.0 LDAP Attribute
`nswcalUser *`	`icsCalendarUser *`
`nswcalCalID`	`icsCalendar`
`nswcalExtendedUserPrefs`	`icsExtendedUserPrefs`
`ceCalList **`	`icsSubscribed`
`ceAgendaList **`	`icsSet`
`ceDefaultAgenda **`	`icsDefaultSet`
`ceDefaultTZID **`	`icsTimeZone`
`ceFirstDayWeek **`	`icsFirstDay`

* Objectclass

** Originally part of `nswcalExtendedUserPrefs`

Migration Process

The `ics2migrate` steps are:

- Upgrade the 2.x Calendar Database
- Migrate the Data
- Check the Migration Results

CAUTION Before you run `ics2migrate`, back up your calendar database using a utility such as `csbackup`, the Sun StorEdge Enterprise Backup™ software, or Legato Networker®.

Backing up your calendar database is very important because `db_upgrade` upgrades the database in its current directory. If a problem occurs during the upgrade, your database could be left in an unrecoverable state.

Upgrade the 2.x Calendar Database

Calendar Server 6.0 requires Berkeley DB version 3.2.9 from Sleepycat Software. Before you run `ics2migrate`, you must use the Berkeley DB `db_recover` and `db_upgrade` utilities to upgrade your calendar database to version 3.2.9. Calendar Server 6.0 includes the Berkeley DB utilities in the following directory:

cal_svr_base`/opt/SUNWics5/cal/tools/unsupported/bin`

For more information about the Berkeley DB utilities, refer to the following web site:

`http://www.sleepycat.com/docs/utility/index.html`

To upgrade your database to version 3.2.9:

1. On Solaris and other UNIX systems, login as the user and group under which Calendar Server is running, such as `icsgroup` and `icsuser`.
2. If necessary, stop the 2.x Calendar Server.
3. Back up your calendar 2.x database, if you have not already done so.

4. Remove (delete) any old share (__db_*name*.share) or log (log.*) files from the following directories:

   cal_svr_base/opt/SUNWics5/cal/lib/http

   cal_svr_base/var/opt/SUNWics5/csdb

5. Run the db_upgrade utility to upgrade your 2.x calendar database to version 3.2.9. If you are not in the same directory with the 2.x calendar database, use the -h option to point to the database files.

   Notes You must run db_upgrade on all 2.x database files (alarms.db, calprops.db, events.db, and todos.db). You must also run db_upgrade on all front-end and back-end servers in your Calendar Server configuration, even if a server is not directly connected to a calendar database.

6. Locate the Calendar Server 2.x caldb.conf file in the csdb directory with the database files and change the first line in the file as follows:

   Old value: caldb.version "1.0.0 [BerkeleyDB]"

   New value: caldb.version= "1.0.0 [BerkeleyDB]"

   Note If this file is not in the csdb directory, create it using a text editor and then set the first line to the new value.

Migrate the Data

Follow these steps to run where ics2migrate:

1. Change to the directory where ics2migrate is located.
2. Run ics2migrate using the syntax in ics2migrate Syntax.
3. After migration, make sure that the caldb.berkeleydb.homedir.path parameter in the ics.conf file points to the migrated database.
4. Run the csdb check command and, if necessary, the csdb rebuild command to rebuild your calendar database.

ics2migrate Syntax

To migrate both the Calendar Server 2.x database and LDAP user preferences

```
ics2migrate [-q] [-s def|none] [-f def|none] [-l min|max] source target
```

To migrate only the Calendar Server 2.x database

```
ics2migrate [-q] [-m db] [-s def|none] [-f def|none] [-l min|max] source
target
```

To migrate only the LDAP user preferences

```
ics2migrate [-q] [-m ldap]
```

NOTE To display the syntax, type `ics2migrate` without any options.

Table 3-3 lists the `ics2migrate` options with a description of each option.

Table 3-3 ics2migrate Options

ics2migrate Option	Description
`[-q]`	Run in quiet mode. If the migration is successful, `ics2migrate` does not display information on the console. If the migration fails, `ics2migrate` displays only errors. The default is verbose mode.
`[-m db|ldap]`	`db` – Migrate only the calendar database. `ldap` – Migrate only the LDAP user preferences. The default is to migrate both the calendar database and LDAP user preferences.
`[-s def|none]`	`def` – Grant scheduling access to only a user's default calendar. `none` – Deny scheduling access to all users' calendars. The default is to grant scheduling access to all calendars.
`[-f def|none]`	`def` – Grant free/busy access to only a user's default calendar. `none` – Deny free/busy access to all users' calendars. The default is to grant free/busy access to all calendars.
`[-l min|max]`	`min` – Log the minimum data migration statistics: calendar ID, primary owner, and number of events and todos for each calendar. `max` – Log the maximum data migration statistics: minimal statistics plus the number of attendees and alarms for each event and todo. `ics2migrate` logs statistics to `ics2migrate.log` in the *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory. By default, `ics2migrate` displays migration statistics on the console and does not generate a log file.
source	Directory where the Calendar Server 2.x database files are located. `source` is required for database migration if the `-m db` option is specified, or if the `-m` option is omitted.
target	Directory where the Calendar Server 6.0 database files are located. `target` is required for database migration if the `-m db` option is specified, or if the `-m` option is omitted.

Check the Migration Results

After you have finished the migration, check the results:

- Check the `ics2migrate.log` file for the following messages (depending on your migration choices):

  ```
  Database migration successfully completed
  LDAP user preference migration successfully completed
  ```

- If you suspect a possible database corruption, run the `csdb` utility `check` command.

  The `check` command scans a calendar database for corruption. If the `check` command finds an inconsistency that cannot be resolved, it reports the situation in its output. If necessary, you can then run the `csdb` utility `rebuild` command to rebuild the calendar database (`caldb`).

  For documentation about the `csdb` utility `check` and `rebuild` commands, see the *Sun ONE Calendar Server Administration Guide* on the documentation web site.

Migration Examples

Migrate Both Calendar Database and LDAP User Information

Migrate both the LDAP user information and the Calendar Server 2.x database. The Calendar Server 2.x database is stored in the `/var/opt/SUNWicsrv/2x_db` directory and the 6.0 database is in the `/var/opt/SUNWics5/50_db` directory.

Grant scheduling and free/busy access to all calendars and log minimal migration statistics in a log file named `ics2migrate.log`.

```
ics2migrate /var/opt/SUNWicsrv/2x_db /var/opt/SUNWics5/50_db -l min
```

Migrate in Quiet Mode

Perform the same migration as the previous example, except operate in quiet mode. `ics2migrate` does not display migration statistics on the console or generate a log file.

```
ics2migrate -q /var/opt/SUNWicsrv/2x_db /var/opt/SUNWics5/50_db
```

Migrate Only the Calendar Database

Migrate only the 2.x calendar database stored in the `2x_db` directory (relative to the current directory) and create a 6.0 database in the `/var/opt/SUNWics5/50_db` directory.

```
ics2migrate -m db 2x_db /var/opt/SUNWics5/50_db
```

Migrate Only LDAP User Information

Migrate only the Calendar Server 2.x LDAP user information to version 6.0 format.

```
ics2migrate -m ldap
```

Migrate Both Calendar Database and LDAP User Information

Migrate both LDAP and calendar database information in the specified directories. Grant scheduling access only to each user's default calendar, deny free/busy access to all calendars on the server, and do not generate statistical information to a log file.

```
ics2migrate -s def -f none 2x_db 50_db
```

ncs4migrate Utility

This section describes how to migrate Netscape Calendar Server 4.x calendar data to Sun ONE Calendar Server using the `ncs4migrate` migration utility.

Netscape Calendar Server 4.x calendars are also known as CS&T calendars for the developer Corporate Software & Technologies Int. Inc.

If you need a copy of the `ncs4migrate` utility, contact your Sun technical support representative or account manager. When you get `ncs4migrate` copy it to your *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory.

This section includes the following information:

Migration Requirements

The migration requires the following hardware and software:

- Source machine — This machine (or machines) has the Netscape Calendar Server 4.0 (or later) data that you plan to migrate.
- Target machine — This machine has the Calendar Server 5.0 database that you plan to migrate to. It should be running Calendar Server 5.0 Patch 4 (or newer).

The source machine and target machine can be different servers or the same server. For a list of supported platforms refer to the Sun ONE Calendar Server Release Notes.

What Gets Migrated?

The following table describes how `ncs4migrate` migrates Netscape Calendar Server 4.0 data to Calendar Server 6.0.

Table 3-4 Migration of Netscape Calendar Server 4.0 Data

Netscape Calendar Server 4.0 Data Item	Calendar Server 5.0 Migration Results
Meetings, events, and notes of resources and users	Migrated as events.
Tasks	Migrated as todos (tasks).

Access (security) rights	Ignored during migration. Designates and Designate Rights are not migrated.
	
	For user's calendars and resource calendars, `ncs4migrate` uses the access control strings in the `ics.conf` file as follows:
	
	For user's calendars, `ncs4migrate` uses `calstore.calendar.default.acl` and sets the privacy settings in the Calendar Server 5.0 calendar as:
	
	• Calendar owner: Availability, Schedule, Read, Delete, and Modify
	
	• All other users: Availability and Schedule
	
	For resource calendars, `ncs4migrate` uses `resource.default.acl` and sets the privacy settings in the Calendar Server 5.0 calendar as:
	
	• Resource owner: Availability, Schedule, Read, Delete, and Modify
	
	• All other users: Availability, Schedule, and Read
	
	For a description of privacy settings and how to change them, see the Calendar Express online Help.
	
	Note Before you migrate, check the strings in the `ics.conf` file to make sure they are correct as follows:
	
	The correct string for `calstore.calendar.default.acl` is:
	
	`@@o^a^r^g;@@o^c^wdeic^g;@^a^sf^g;@^c^^g`
	
	The correct string for `resource.default.acl` is:
	
	`@@o^a^r^g;@@o^c^wdeic^g;@^a^rsf^g;@^c^^g`

File attachments	Ignored during migration; warning is generated in log file.
Groups	Not migrated.

Migration Steps

Backup the Calendar Server 5.0 Database

Before you migrate, it is recommended that you perform these steps to ensure the integrity of your calendar database:

1. Backup your calendar database using a utility such as `csbackup`, the Sun StorEdge Enterprise Backup™ software, or Legato Networker®.

   For information, see the *Sun ONE Calendar Server Administrator's Guide*.

2. Run the `csdb` utility `check` command on your calendar database to check for any database corruption. If the `check` command detects any corruption, run the `csdb` utility `rebuild` command to rebuild the database.

   For documentation about the `csdb` and `csbackup` utilities, see the *Sun ONE Calendar Server Administrator's Guide*.

Prepare to Migrate

Before you run the `ncs4migrate` utility, perform these steps on the target machine:

1. Log in as or become superuser (`root`) or as a user who has administrator rights to the system.

2. Change to the *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory.

3. Create a text file named `ncs4dirpaths.dat` and specify the fully qualified directory path to the Netscape Calendar Server 4.0 database. For example:

   `/apps/ncs/calendar/unison/db/nodes/N0/perm`

   To locate the directory that contains the Netscape Calendar Server 4.0 database, search for the `unison.dbd` file.

   If necessary, fulfill any requirements to allow `ncs4migrate` to access the node and read the directory where the Netscape Calendar Server 4.0 database is located.

NOTE Do not use variables such as `$CAL_HOME` in the pathname. Variables are not be resolved during migration.

For information about creating an `ncs4dirpaths.dat` file for data on multiple nodes, see Migrating Data From Multiple Nodes.

4. If you plan to migrate selected users, create a user filter file named `ncs4userfilter.dat` in the *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory. `ncs4userfilter.dat` is a text file that specifies the users you want to migrate. Each line identifies a user in either of the following formats:

   - `node-number:user id` in Netscape Calendar Server calendar system (`nscalxitemid` attribute)
   - user's `UID` attribute

   For example, several entries in a `ncs4userfilter.dat` file might be:

   ```
   caluser1
   caluser2
   10000:00256
   10000:00257
   ```

   You can use both formats in the same `ncs4userfilter.dat` file.

5. Make sure that the LDAP server is running.
6. To prevent updates to the calendar database during the migration, stop the Calendar Server. The Netscape Calendar Server, however, can be either running or stopped.

You are now ready to migrate the Netscape Calendar Server 4.0 data.

Migrate the Data

On the target machine, perform these steps:

1. While logged in as superuser (`root`)or a user who has administrator rights to the system, change to the *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory (if necessary).
2. Type `ncs4migrate` on the command line.

The `ncs4migrate` utility then displays its welcome menu with the options shown in Table 3-5.

Note: Although `ncs4migrate` displays the (E)xport and (I)mport options, these options are not supported and should not be used.

Table 3-5 ncs4migrate Utility Options

ncs4migrate Option	Description
(E)xport	Export Netscape Calendar Server 4.0 calendar database to intermediate files.
(I)mport	Import the data from intermediate files into the calendar database.
(S)kip	Skip intermediate files. Just migrate one record at a time from Netscape Calendar Server 4.0 to Calendar Server 5.0.
(L)ogging = ON|OFF	Set Logging. Logging filename is `ncs4migrate_`*`yyyymmdd-hhmmss`*`.log`. Default is ON.
(V)erbose = ON|OFF	Set Verbose log. Default is OFF.
	To save disk space, we recommend leaving as OFF.
(D)ebug = ON|OFF	Set Debug log. Default is OFF.
(Q)uiet = ON|OFF	Set for screen output. Default is OFF.
(T)erminate = TRUE|FALSE	Terminate if a user in the Netscape Calendar Server 4.0 database is not in LDAP. Default is FALSE.
(O)nly = TRUE|FALSE	Migrate only users in the user filter file `ncs4userfilter.dat`. Default is FALSE.
	If O and M are TRUE, `ncs4migrate` migrates any event that has any participant in the filter file as either an owner or attendee. All attendees will have the event migrated to their calendars.
(M)igrate = TRUE|FALSE	Migrate users in the user filter file. Default is FALSE.
(B)ypass = TRUE|FALSE	Bypass migration for users in the user filter file. Default is FALSE.
(A)ny = TRUE|FALSE	Any combination of Netscape Calendar Server security access levels produces a grant in Calendar Server. Default is TRUE. FALSE means all 3 access levels need to be present; see (H)elp.
(U)ser	Display user filter file `ncs4userfilter.dat`. Use O option to turn filtering ON|OFF. Default is OFF.
(P)ath	Path file for Netscape Calendar Server 4.0 databases. Filename is `ncs4dirpaths.dat`.
(H)elp	Display Help screen
(E)xit	Exit the program.

3. From the `ncs4migrate` menu, specify the S option to migrate all users. Or, if you are migrating specific users in a user filter file (`ncs4userfilter.dat`), specify the O option.
4. Monitor the migration log file to check the migration status. See Checking the Migration Log File for more information.
5. After the migration is finished, check the migrated calendar database as described in Check the Migrated Data.

Migrating Data From Multiple Nodes

To migrate Netscape Calendar Server 4.0 data from multiple nodes, perform these steps on the target machine:

1. While logged in as superuser (`root`)or as a user who has administrator rights to the system, copy the Netscape Calendar Server 4.0 database directory from each node to the machine where you plan to run `ncs4migrate`. (Each Netscape Calendar Server 4.0 directory should contain a `unison.dbd` file.)

   You can also migrate the Netscape Calendar Server 4.0 data directly from each node; however, you must first fulfill any requirements to allow `ncs4migrate` to access the Netscape Calendar Server 4.0 data on the other nodes.
2. Change to the *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory.
3. In the `ncs4dirpaths.dat` file, specify a directory pathname for data from all nodes. For example, the following `ncs4dirpaths.dat` file incudes directory paths for three nodes:

   ```
   /apps/ncs/calendar/unison/db/nodes/N0/perm
   /apps/ncs/calendar/unison/db/nodes/N1/perm
   /apps/ncs/calendar/unison/db/nodes/N2/perm
   ```
4. To run the migration utility, type `ncs4migrate` on the command line.
5. From the `ncs4migrate` menu, specify the S option to migrate all users. Or, if you are migrating specific users in a user filter file (`ncs4userfilter.dat`), specify the O option.
6. Monitor the migration log file to check the migration status. See Checking the Migration Log File for more information.
7. After the migration is finished, check the migrated calendar database, as described in Check the Migrated Data.

Checking the Migration Log File

The `ncs4migrate` utility generates a log file with the following name in the *cal_svr_base*`/opt/SUNWics5/cal/sbin` directory:

`ncs4migrate_`*`yyyymmdd-hhmmss`*`.log`

where *`yyyymmdd-hhmmss`* is a timestamp that indicates when the migration started.

If the `ncs4migrate` utility is taking a long time to run, check that the log file is increasing in size as an indication that the utility is still running.

NOTE To prevent the log file from becoming too large, consider omitting the `ncs4migrate` verbose (V) option.

Check the Migrated Data

After the migration is finished, perform these steps on the target machine:

1. Run the `csdb` utility `check` command to scan the calendar database to determine if any corruption has occurred. If the `check` command detects any corruption, run the `csdb` utility `rebuild` command to rebuild the database.

   For documentation about the `csdb` utility `check` and `rebuild` commands, see the *Sun ONE Calendar Server Administration Guide* on the documentation web site.

2. If necessary, restart the Calendar Server.

   Users can access the migrated calendar database using Calendar Express.

csrename Utility

The csrename utility renames calendar users as follows:

- Calendar database files–Renames users (user IDs) in the calendar database files and then writes the new database files to a destination directory. The existing calendar database files are not modified.
- LDAP directory server–Converts the user IDs in the Calendar Server LDAP attributes (that is, attributes with the "ics" prefix). The LDAP directory server is modified in place.

The csrename utility is located in the following directory:

cal_svr_base/opt/SUNWics5/cal/sbin

Before you run csrename, you must first:

- Create an input mapping file (-m option) for the users you want to convert.
- Provision any new users in the LDAP directory server, if necessary.
- Stop Calendar Server.

To you run csrename, you must log in as icsuser (or as the Calendar Server runtime user ID specified during configuration). If you run csrename as superuser (root), you might need to reset the permissions for the new database files.To modify the LDAP directory server attributes, you must also have administrative rights for that directory.

If your has a front-end/back-end server configuration, you must run csrename on each back-end server.

csrename Syntax

Use the following syntax o run csrename:

```
csrename [-t DestinationDB ] [-c ConfigFile ] [-e ErrorFile ] -m MappingFile
rename [DB|LDAP]
```

-t *DestinationDB* specifies the destination directory where csrename generates the new database with the converted user names. The default is MigratedDB.

After csrename is finished, the caldb.berkeleydb.homedir.path parameter in the ics.conf file must point to the destination database. Either reset caldb.berkeleydb.homedir.path to point to the destination database directory, or move the destination database files to the directory indicated by the parameter.

-c *ConfigFile* is an input parameter that specifies a Calendar Server configuration file. The default is the ics.conf file.

csrename uses the caldb.berkeleydb.homedir.path parameter in the configuration file to determine the location of the input calendar database. The default location of the calendar database is *cal_svr_base*/var/opt/SUNWics5/csdb.

-e *ErrorFile* is the file where csrename writes any errors or database entries that cannot be resolved. The default is MigrateError.

-m *MappingFile* specifies an input mapping file. The default is MigrateMapping.

The input mapping file is a text file that maps existing user IDs to new user IDs.

You must create the mapping file before you run csrename. Specify one entry per line with a space between the old and new values. For example:

```
tchang tc897675
jsmith js963123
...
bkamdar bk548769
```

DB|LDAP specifies the database that gets updated:

- DB converts user IDs in the new calendar database only (default).
- LDAP converts user IDs in both the new calendar database and the LDAP directory server attributes.

csrename Examples

- Rename users based on the mapping file named DBMapFile and create the new calendar database in the destination directory named newcalDB:

  ```
  csrename -t newcalDB -m DBMapFile rename DB
  ```

- Rename users based on values in the mapping file named NewNames, create the new calendar database in the destination directory named NewDB, and modify the Calendar Server attributes in the LDAP directory server:

  ```
  csrename -t NewDB -m NewNames rename LDAP
  ```

Appendix A

Configuration Worksheets

This appendix contains the following worksheets to help you keep track of the information you need to run the Calendar Server configuration program, which is described in Chapter 2, "Configuring Calendar Server 6.0."

- Directory Server Setup Script Worksheet
- Calendar Server Configuration Worksheets
  - Administration, User Preferences and Authentication Panel Worksheet
  - Email and Email Alarms Worksheet
  - Runtime Configuration Worksheet
  - Database, Logs, and Temporary Files Directories Worksheet

Directory Server Setup Script Worksheet

Table 3-6 Directory Server Setup Script (comm_dssetup.pl) Worksheet

Option	Description
`-i yes|no`	Add new Directory Server indexes (yes/no). Default: yes Your value: _____
Directory Server Root	Directory Server root path name. Default: `/var/mps/serverroot` Your value: _____
Directory Server Instance	Directory Server instance subdirectory. Default: none Your value: _____
DC Root	DC Tree root suffix. Default: `o=internet` Your value: _____
User and Group Base Suffix	User/Group root suffix. Default: `o=usergroup` Your value: _____
`-s yes|no`	Update schema (yes/no). Default: yes Your value: _____
Directory Manager DN	Directory Manager Distinguished Name (DN). Default: `"cn=Directory Manager"`. Your value: _____
Directory Manager DN Password	Directory Manager DN password. Default: none. Your value: _____
`-b yes|no`	Use this directory to store both configuration and user data (yes) or configuration data only (no). Default: yes Your value: _____

Table 3-6 Directory Server Setup Script (comm_dssetup.pl) Worksheet

Option	Description
`-t 1|1.5|2`	Sun ONE LDAP Schema version: • Option 1–v.1 • Option 1.5–v.2 Compatibility Mode • Option 2–v.2 Native Mode Default: 1 Your value: _____
`-s` *PathtoSchemaFile*	Path to the directory where the schema files are located. Default: `./schema` Your value: _____

Calendar Server Configuration Worksheets

Administration, User Preferences and Authentication Panel Worksheet

Table A-1 Administration, User Preferences and Authentication Panel Worksheet

Option	Description
LDAP Server Host Name	Host name of the LDAP directory server you are using for user authentication. Default: current host. Your value: _____
LDAP Server Port	Port number that the LDAP server listens on. Default: 389. Your value: _____
Base DN	Entry in the LDAP directory used as the starting point from which searches will occur. Default: `o=host.com`. Your value: _____
Directory Manager DN	User name that can make changes in the directory server schema. Default: `cn=Directory Manager`. Your value: _____
Directory Manager Password	Password of the Directory Manager DN. Default: None Your value: _____
Administrator User ID	User ID of the Calendar Server Administrator. This user must be a user in the above LDAP directory server. Default: `calmaster`. Your value: _____
Administrator Password	Password of the Calendar Server Administrator. Default: None Your value: _____

Email and Email Alarms Worksheet

Table A-2 Email and Email Alarms Worksheet

Option	Description
Email Alarms	Specifies whether Calendar Server should send an email alarm message to a Calendar Server administrator in case a server problem occurs. Default: Enabled. Your value: _____
Administrator Email Address	Email address of the Calendar Server Administrator who will receive the email alarm messages. Default: None. Your value: _____
SMTP Host Name	Host name of the SMTP server where email alarm messages should be sent. Default: Current host. Your value: _____

Runtime Configuration Worksheet

Table A-3 Runtime Configuration Worksheet

Option	Description
Service Port	Port number that Calendar Server listens on to provide Web (HTTP) access to users. Default: 80. Your value: _____
Maximum Sessions	Maximum number of Calendar Server sessions. Default: 5000. Your value: _____
Maximum Threads	Maximum number of Calendar Server threads. Default: 20. Your value: _____

Table A-3 Runtime Configuration Worksheet

Option	Description
Number of Server Processes	Maximum number of Calendar Server processes. Default: Number of CPUs on the server where you are installing Calendar Server. Your value: _____
Runtime User ID	UNIX user name under which Calendar Server will run. Default: `icsuser`. Your value: _____
Runtime Group ID	UNIX group under which Calendar Server will run. Default: `icsgroup`. Your value: _____
Calendar Server Startup	Start after successful installation. Default: Checked. Your value: _____ Start on system startup. Default: Checked. Your value: _____

Database, Logs, and Temporary Files Directories Worksheet

Table A-4 Database, Logs, and Temporary Files Directories Worksheet

Option	Description
Database Directory	Directory where Calendar Server should create and store the calendar database files. Default: `var/opt/SUNWics5/csdb` Your value: _____
Logs Directory	Directory where Calendar Server writes log files. Default: `var/opt/SUNWics5/logs` Your value: _____
Temporary Files Directory	Directory where the Calendar Server writes temporary files. Default: `var/opt/SUNWics5/tmp` Your value: _____

Appendix B

LDAP Directory Server Considerations

Sun™ ONE Calendar Server 6.0 stores and manages calendars, calendar properties, access control information, events, todos (tasks), and alarms. To manage storage for user information, however, the Calendar Server requires a directory service to perform operations such as user authentication and the storage and retrieval of user preferences.

This appendix describes:

- Updating an LDAP Server Schema Manually
  - Sun ONE or iPlanet Directory Server
  - Netscape Directory Server
- Resolving Conflicting OIDs in the LDAP Schema Directory

For the LDAP directory servers supported by Calendar Server 6.0, refer to the Calendar Server 6.0 Release Notes on the following Web site:

http://docs.sun.com/coll/S1_CalendarServer_60

If your users are already stored in an LDAP directory, the simplest solution for deploying the Calendar Server is to upgrade your directory server to the Sun ONE Directory Server.

For information about installing and configuring a Directory Server, see:

http://docs.sun.com/coll/S1_DirectoryServer_52

If you are using another directory server, or if the installation program cannot update your directory server, you must modify your schema manually to allow your users to access Calendar Server 6.0.

Updating an LDAP Server Schema Manually

In some instances, you might need to update the following directory servers manually:

- Sun ONE or iPlanet Directory Server
- Netscape Directory Server

Sun ONE or iPlanet Directory Server

The Sun ONE Directory Server 5.2 and iPlanet Directory Server 5.1 LDAP schema extensions used by the Calendar Server are defined in the `60iplanet-calendar.ldif` file.

The Calendar Server installation program installs this file in the `/opt/SUNWics5/cal/config` directory.

Note If you update your LDAP server schema manually and then later upgrade Calendar Server, you must manually update the LDAP server schema again. Calendar Server cannot automatically update a directory server schema after the schema has previously been updated manually.

To update Sun ONE or iPlanet Directory Server manually:

1. Install Calendar Server 6.0.
2. Stop Calendar Server, if it is running.
3. Stop Directory Server, if it is running.
4. Copy the `60iplanet-calendar.ldif` file to the following directory on the server where your directory server is running:

   dir_svr_base`/slapd-`*hostname*`/config/schema`

   where *dir_svr_base* is the Directory Server installation directory and *hostname* identifies the server.
5. Restart the Directory Server. If you receive OID errors, see Resolving Conflicting OIDs in the LDAP Schema Directory.
6. Restart Calendar Server.

Netscape Directory Server

For Netscape Directory Server 4.12 or 4.16, the LDAP schema extensions used by the Calendar Server are defined in the following files:

- `um50-common-schema.conf` defines the LDAP attributes and object classes shared by Sun ONE products.
- `ics50-schema.conf` defines the LDAP attributes and object classes used by the Calendar Server.

These files are available in the `/opt/SUNWics5/cal/config` directory.

To update Netscape Directory Server 4.12 or 4.16 manually:

1. Install Calendar Server 6.0.
2. Copy the LDAP schema files (`um50-common-schema.conf` and `ics50-schema.conf`) from the `/opt/SUNWics5/cal/config` directory to the following directory on the server where your directory server is running:

   server-root`/slapd-`*hostname*`/config`

   where *hostname* is the name of the server.

   For example, on Solaris and other UNIX systems:

   `/usr/Netscape/Server4/slapd-sesta/config`
3. Stop Calendar Server, if it is running.
4. Stop Directory Server, if it is running.
5. Edit the `ns-schema.conf` file (in the same directory in which you copied the `um50-common-schema.conf` and `ics50-schema.conf` files. At end of the file, if they are not already present, add the following lines to include these files.

   On Solaris and other UNIX systems:

```
include /netscape/server4/slapd-hostname/config/um50-common-schema.conf
include /netscape/server4/slapd-hostname/config/ics50-schema.conf
```

   On Windows 2000 systems:

```
include "C:\Netscape\Server4\slapd-hostname\config\um50-common-schema.conf"
include "C:\Netscape\Server4\slapd-hostname\config\ics50-schema.conf"
```

   where *hostname* is the name of the server where the directory server is running.

NOTE Be sure to add the lines in the order shown above so that `um50-common-schema.conf` is included before `ics50-schema.conf`.

6. Restart the Netscape Directory Server. If you receive OID errors, see Resolving Conflicting OIDs in the LDAP Schema Directory.
7. Restart Calendar Server.

Resolving Conflicting OIDs in the LDAP Schema Directory

If your LDAP schema directory contains conflicting OIDs, the Directory Server does not know which OID to use and returns an error message. For example, the following message indicates a conflicting OID for the `icsCalendarUser` object class for iPlanet Directory Server 5.1:

```
[24/Jul/2002:23:45:28 -0700] dse - The entry cn=schema in file
/export/iplanet/servers/slapd-ical/config/schema/99user.ldif is invalid,
error code 20 (Type or value exists) - object class icscalendaruser: The
name does not match the OID. Another object class is already using the name
or OID.

[24/Jul/2002:23:45:28 -0700] dse - Please edit the file to correct the
reported problems and then restart the server.
```

This problem can occur when you install Calendar Server 6.0 and you also had an older Calendar Server release that dynamically updated your LDAP server schema `99user.ldif` file.

To resolve the conflicting OIDs, you must edit the `99user.ldif` file and remove the older OIDs. For Calendar Server 6.0, the following table shows the specific OIDs that might cause problems.

Table B-1 Calendar Server OIDs in the LDAP Schema Directory

Object Class	Old OID	New OID
icsCalendarUser	2.16.840.1.113730.3.2.141	1.3.6.1.4.1.42.2.27.9.2.2
icsCalendarResource	2.16.840.1.113730.3.2.143	1.3.6.1.4.1.42.2.27.9.2.3
icsCalendarDomain	2.16.840.1.113730.3.2.144	1.3.6.1.4.1.42.2.27.9.2.4

After you edit the `99user.ldif` file, restart the Directory Server.

Appendix C

Calendar Server 5.x to 6.0 Upgrade/Migration Process

This appendix describes how to upgrade/migrate Sun™ ONE or iPlanet™ Calendar Server 5.x to Calendar Server 6.0, including:

- Upgrade/Migration Process for upgrading to Calendar Server 6.0
- XSL Tips for merging changes into the Calendar Server 6.0 XSL files

Upgrade/Migration Process

To upgrade to Calendar Server 6.0:

1. On the server where Calendar Server 5.x is installed, change to the `/opt/SUNWics5/cal/sbin` directory and stop Calendar Server 5.x:

   ```
   ./stop-cal
   ```

2. Backup the following Calendar Server 5.x directories and files:
   - Calendar database directory–Default location is: `/var/opt/SUNWics5/csdb`.
   - Calendar Server logs directory–Default location is: `/var/opt/SUNWics5/logs`.
   - `ics.conf` configuration file–Default is: `/opt/SUNWics5/cal/bin/config/ics.conf`.
3. Backup any Calendar Server 5.x files that you have customized, including:
   - User interface (UI) XSL files in `/opt/SUNWics5/cal/bin/data`

- Mail formatting files–Default location is:
  `/opt/SUNWics5/cal/bin/config/`*language*`/`
- `default.html` file–Default location is:
  `/opt/SUNWics5/cal/bin/html/`*language*`/`
- HTML files–For example, in:
  `/opt/SUNWics5/cal/bin/html/`*language*`/`
- Time-zone files such as `timezones.ics` and `change_timezone.xml`.

  The *language* directory includes the files for the language you are using. For example, en–English, de–German, es–Spanish, fr–French, ja–Japanese, zh-TW–Traditional Chinese, or zh-CN–Simplified Chinese.

4. Change to the `/opt/SUNWics5/cal/` directory and uninstall Calendar Server 5.x. You must be running as superuser (`root`). For example:

   ```
   ./uninst
   ```

5. Remove all files under the `/opt/SUNWics5` directory.
6. Run the Java Enterprise System installer to install Calendar Server 6.0 in the `/opt/SUNWics5` directory. For information, refer to the *Sun Java Enterprise System Installation Guide*.
7. Change to the `/opt/SUNWics5/cal/sbin` directory and run the `comm_dssetup.pl` utility to configure your Sun ONE Directory Server 5.x to work with Calendar Server 6.0. For information about `comm_dssetup`, refer to the "Directory Server Setup Script (comm_dssetup.pl)" on page 25.
8. Run the Calendar Server 6.0 configuration program (`csconfigurator.sh`) to create a new `ics.conf` configuration file. For more information, refer to "Calendar Server Configuration Program (csconfigurator.sh)" on page 35
9. If you want to keep your 5.x configuration, copy the `ics.conf` file from your backup location or tape to the `/etc/opt/SUNWics5/config/ics.conf` file.
10. Set the UNIX user and group IDs (default is `icsuser` and `icsgroup`) and permissions for the `ics.conf` file. For example:

    ```
    chown icsuser:icsgroup /opt/SUNWics5/cal/config/ics.conf
    chmod 600  /opt/SUNWics5/cal/config/ics.conf
    ```

11. Run the `cs5migrate` utility to migrate your Calendar Server 5.x database to 6.0. For information, refer to the "cs5migrate Utility" on page 47.

12. If the `cs5migrate` migration is successful, copy the migrated 6.0 database to the `/var/opt/SUNWics5/csdb` directory. Make sure that the owner of all files in the `/csdb` directory is `icsuser` and `icsgroup` (or the user ID and group ID you have specified in the `ics.conf` file).
13. The Calendar Server 6.0 XSL. XML, HTML, and mail formatting files have significant changes from the 5.x release. If you saved the 5.x files in Step 3, you must manually merge your customization changes from the 5.x files to the 6.0 files. For tips for merging changes into XSL files, see "XSL Tips" on page 95.
14. Change to the `/opt/SUNWics5/cal/sbin` directory and start Calendar Server 6.0:

    ```
    ./start-cal
    ```

15. Login to Calendar Server 6.0 using Calendar Express and make sure that you are able to view the Calendar Server components (events and todos).

XSL Tips

If you are merging changes into the Calendar Server 6.0 XSL files, follow these tips and guidelines to avoid common mistakes:

- Do not use the wrong syntax when accessing an attribute. For example:

```
./element@attribute
```

This should be:

```
./element/@attribute
```

The slash (/) before the ampersand (@) is very important.

- Do not put a slash (/) at the end of a node. For example:

```
/node1/node2/
```

  This should be:

```
node1/node2
```

- Do not use non-matching open and close braces. For example, to have different background colors depending on a value, you might write multiple `<TD bgcolor={$bgcolor_variable}>` within different `<xsl:when>` statements, all finally ending with a single `</TD>`.

  If the parser complains about non-matching braces, you might write `<TD>` as `<TD>`.

  Instead, however, use either of these methods:

  - Define a new variable and set its value depending on the different states and then use this new variable in the `<TD>` line.
  - Or, define `bgcolor` separately using `<xsl:attribute>` for that `<TD>`.

  In either case you end up with just one `<TD>` line. In general, do not use "linearized" tags, such as `< TR>`.

- Do not close tags before they should be closed. For example:

```
<INPUT type="checkbox" name="ceGroupInviteAll"/>
    <xsl:choose>
      <xsl:when test="/calendar/usrctx/userprefs@ceGroupInviteAll='0'">
        </xsl:when>
          <xsl:otherwise>
          <xsl:attribute name="checked"/>
      </xsl:otherwise>
    </xsl:choose>
```

Notice the slash (/) at the end of the first line. It closes the INPUT tag; thus the attribute being checked later is ignored by most parsers. The correct way to do this is remove the last slash (/) in the first line and put </INPUT> when finished. For example:

```
<INPUT type="checkbox" name="ceGroupInviteAll">
    <xsl:choose>
      <xsl:when test="/calendar/usrctx/userprefs@ceGroupInviteAll='0'">
        </xsl:when>
          <xsl:otherwise>
          <xsl:attribute name="checked"/>
      </xsl:otherwise>
    </xsl:choose>
</INPUT>
```

- Do not use undeclared variables. Make sure variables are declared before they are used. Some parsers ignore undeclared variables, but other parsers flag them as errors.
- Do not use spaces in variable names. For example:

```
<xsl:variable name="test ">
```

- Do not use multiple variables with same name. Do not redeclare the same variable at different levels.
- Do not use position(.). This should be position().
- Do not use "//node". For example:

```
<xsl:value-of
select="//panel[@name='weekgrid']/weekcal[1]/weekday[1]/command[2]"/>
```

Do not use the double slash (//).

- Do not use entities such as ` ` and `&`. They should be declared as entities in the `DOCTYPE` line. For example:

```
<!DOCTYPE xsl:stylesheet [<!ENTITY nbsp " ">]>
```

- Do not use hex values. Values such as `•` must be declared as an entity in the `DOCTYPE` line. For example:

```
<!DOCTYPE xsl:stylesheet [<!ENTITY nbsp " "> <!ENTITY bullet
"&#149;">]>
```

  To use `#149` within the `xsl`, write it as:

```
<xsl:text>&bullet;</xsl:text>
```

- Do not use a wrong `stylesheet` tag. Use the following for any new `xsl` file you write:

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
```

- Do not use comments within `<xsl text>`. For example:

```
<xsl:text><!-- This is a comment --></xsl:text>.
```

  Do not write comments inside the `<xsl:text>` node.

- Do not use very complex patterns in template rules. Instead of very complicated and long rules, use “and” and “or” and try to use `<xsl:choose>` within the rules to make them simpler.
- Do not use counting to check for existence. To check if a node exists, instead of checking if `count(node)` is greater than zero, just check if node exists.
- Do not use too many XSLT functions. Try to reduce use of XSLT functions such as `string-length`, especially in heavily used templates.
- Do not use `for-each`. Try to use `apply-templates` where possible instead of `for-each`. This leads to a better and more natural style for XSLT.
- Do not use preceding `[-sibling]` or following `[-sibling]` axes.
- Do not evaluate the same node multiple times in the same template. When the same node needs to be evaluated multiple times, do it once and save the result in a variable.
- Do not use angle brackets in conditional expressions. Use `<` and `>` instead of “<” and “>” in conditional expressions.
- Do not use `value=""` for radio buttons and check boxes: For example:

```
<xsl:attribute name="checked" value=""/>
```

This should be:

```
<xsl:attribute name="checked"/>
```

- Do not use non-existent nodes and attributes. For example, if `usrctx` and `intervalMinutes` do not exist under the `/calendar/usrctx` node:

```
/usrctx/@intervalMinutes
```

- Do not use `xsl` code between open and close `BR`. For example:

```
<BR> <xsl:if>.......</xsl:if></BR>
```

  For an empty line use only `<BR/>`.

- Do not use too many spaces or too much indentation. Although it makes the `xsl` more readable, spaces can make the `xsl` behave differently.
- Do not use too many variables in the `i18n.xsl` file. Also, do not declare variables that are not used in the application.

Glossary

access control entry (ACE) A string that provides access control for calendars, calendar properties, and calendar components such as events and todos (tasks). An example of an ACE is`jsmith^c^wd^g`.

access control list (ACL) A set of access control entry (ACE) strings that collectively provide access control for calendars, calendar properties, and calendar components such as events and todos (tasks). An example of an ACL with three ACEs, with each ACE separated by a semi-colon is `@@o^a^r^g;@@o^c^wdeic^g;@^a^sf^g`.

alarm event An event generated and sent by the Calendar Server Event Notification Service (ENS). When an alarm event occurs, a message reminder is sent to specific recipients.

authentication The verification of a user ordinarily done using a user ID and a corresponding password. Knowledge of the password is assumed to guarantee that the user is authentic. The Calendar Server requires a directory service such as an LDAP server for user authentication.

base DN The distinguished name (DN) that identifies the starting point of a search in an LDAP directory. Also known as a search base. For example, `ou=people,o=sesta.com`.

Berkeley DB A transactional database intended for high-concurrency read-write workloads and for applications that require transactions and recoverability. The Calendar Server uses the Berkeley DB from Sleepycat Software Inc. for storing calendar data.

Calendar Express A Web-based calendar client program that provides access to the Calendar Server for end users.

calendar group A collection of several calendars that can help a user manage more than one calendar.

calendar ID (`calid`) A unique identifier associated with a calendar in the Calendar Server database. The format for a calendar ID is `userid[:calendar]` where `userid` is the user ID and `calendar` is the calendar name.

Calendar Lookup Database (CLD) A plug-in that determines the physical location of a calendar when the calendar database is distributed over two or more back-end servers. Calendar Server provides the LDAP CLD plug-in and the algorithmic CLD plug-in.

Calendar Server Application Programming Interface (CSAPI) A programmatic interface that provides the capability to modify or enhance the feature set of the Calendar Server. CSAPI modules are plug-ins that are loaded from the `cal/bin/plugins` directory when the Calendar Server is started.

Calendar Access Protocol (CAP) A standard Internet protocol for calendaring based on requirements identified by the Internet Engineering Task Force (IETF).

common name (cn) An attribute that identifies the person or object defined by the entry in an LDAP directory.

component state A set of attributes that describe a calendar event such as a meeting. In WCAP, the `compstate` parameter allows fetch commands to return events by component state. For example, `compstate` might be REPLY-DECLINED (attendee has declined a meeting) or REQUEST_NEEDS-ACTION (attendee has not taken action on a meeting yet).

Calendar User Agent (CUA) An application that a calendar client uses to access the Calendar Server.

default calendar The calendar a user first sees after logging into Calendar Express. Usually, the calendar ID of a default calendar is the same as the user's user ID. For example, `wchang@sesta.com` would have a default calendar named `wchang`.

directory service A centralized repository of directory information for use by other servers. The Calendar Server requires that a calendar user be stored in a directory server such as an LDAP server. The Calendar Server then uses the directory server for user authentication and for the storage and retrieval of user preferences. See also LDAP (Lightweight Directory Access Protocol).

distinguished name (DN) A string representation that uniquely identifies a user, system, or organization. A DN identifies an entry in an LDAP directory from which searches will occur. Also known as a search base. For example, `ou=people,o=sesta.com`.

Database Wire Protocol (DWP) A Calendar Server proprietary protocol that allows multiple servers to be linked together within the same Calendar Server system to form a distributed calendar store. The Calendar Servers uses DWP to retrieve remote data stored in the calendar database.

event A entry with an associated date and time in a calendar. For example, an event might be a new meeting or appointment on a calendar.

Event Notification Service (ENS) A generic service that accepts reports of server-level events that can be categorized and then notifies other servers that have registered interest in certain categories of events.

Extensible Markup Language (XML) A flexible programming language developed by the World Wide Web Consortium (W3C) to create common information formats and share both the format and the data on the Web, intranets, and elsewhere. XML is extensible because, unlike HTML, the markup symbols are unlimited and self-defining. The Calendar Server uses XML and XSL to generate the Calendar Express user interface.

Extensible Style Language (XSL) A language used to create style sheets for XML. XSL describes how data sent over the Web using the XML is to be presented to the user. The Calendar Server uses XSL and XML to generate the Calendar Express user interface.

Group ID (GID) On UNIX systems, the group for Calendar Server files such as counters and logs. The GID is stored in the `ics.conf` file in the `local.servergid` parameter.

GMT (Greenwich Mean Time) The mean solar time of the meridian of Greenwich, England, and the time standard against which all other time zones in the world are referred. GMT is not affected by Daylight Savings Time or Summer Time.

Group Scheduling Engine (GSE) The Calendar Server process that handles group scheduling. The GSE enables a user to schedule events with other calendar users on the same server or on a different server. The other user can then modify, cancel, or reply to the event.

High Availability (HA) A configuration that enables two Solaris servers to run a single instance of Calendar Server that remains continuously available after any single point of failure in hardware (disk, server, or network) or software has occurred in either of the servers.

horizontal scalability The Calendar Server's capability to run on a single server or as a group of processes that are spread across multiple servers with a variety of configuration options.

Hypertext Transfer Protocol (HTTP) A standard protocol that allows the transfer of hypertext documents over the Web. The Calendar Server uses HTTP as its primary transport.

instance A Calendar Server configuration of one or more server processes. Multiple Calendar Server instances can be configured per server.

ISO 8601 An ISO (International Organization for Standardization) standard that specifies the numeric representation of date and time. The Calendar Server uses ISO 8601 standard notations to represent date, time, and duration strings.

LDAP (Lightweight Directory Access Protocol) A directory service protocol defined by the Internet Engineering Task Force (IETF) used for the storage, retrieval, and distribution of information, including user profiles, distribution lists, and configuration data.

LDAP server A software server that maintains an LDAP directory and services queries to the directory. The Calendar Server uses Sun ONE Directory Server or iPlanet Directory Server, which are implementations of an LDAP server.

notification A message describing an event occurrence. An example of a notification in Calendar Server is a reminder for an upcoming meeting.

notification service A service that receives subscriptions and notifications from other servers and then relays notifications to specific subscribers. The Calendar Server `csnotifyd` service sends notifications of events and todos (tasks) using the Event Notification Service (ENS) as the broker for the events.

permissions The settings that control the access to a calendar. For example, in Calendar Express, permissions include Availability, Invite, Read, Delete, and Modify. Calendar Server administrators set permissions as access control entry (ACE) strings using command-line utilities. See also access control entry (ACE) and access control list (ACL).

plug-in An accessory program that can be loaded and then used as part of the overall system. For example, the Calendar Server can use a plug-in to access a non-LDAP directory service.

resource calendar A calendar associated with a resource such as a meeting room or equipment such as a notebook computer or overhead projector.

service A component of an overall system. The Calendar Server has the following services: Administration Service (csadmind). HTTP Service (cshttpd), Notification Service (csnotifyd), Event Notification Service (enpd), and Distributed Database Service (csdwpd).

server root A directory location relative to other files on a server. For example, the default Calendar Server installation on Solaris systems uses the path `/opt/SUNWics5/cal/` as the server root.

SHTML (Server-side Include Hypertext Markup Language) An HTML file that includes embedded server-side includes (SSIs).

Single Sign-on (SSO) A authentication mechanism that enables a user to log in once and then access multiple applications. These applications form a circle of trust that use each other's cookies as verification of authority so that the user does not have to login to each application separately.

task In Calendar Express on the client side, a component of a calendar that specifies something to be done. On the server side, a task is also called a todo.

time zone A geographical region that uses the same time. There are 25 hourly time zones from -12 through +12 (GMT is 0). Each time zone is measured relative to GMT. Most time zones have localized designations in three-letter abbreviations. The Calendar Server also identifies time zones using a time zone ID (TZID) such as America/Los_Angeles or Asia/Calcutta.

todo On the server side, a a component of a calendar that specifies something to be done. In Calendar Express on the client side, a todo is called a task.

user ID (UID) A unique string identifying a user to a system. The Calendar Server identifies each user by a user ID.

Universal Principle Name (UPN) The value for a logged-in user that includes the login name combined with the domain to which the user belongs. For example, user `bill` in domain `sesta.com` has the UPN `bill@sesta.com`.

WCAP (Web Calendar Access Protocol) A high-level, command-based protocol used by clients to communicate with the Calendar Server.

Zulu time A military designation for GMT and UTC (Coordinated Universal Time).

Index

D

E

G

H

I

L

M

N

P

R

S

U

W

X