# Sun Internet™ Mail Server™ 3.5 Administrator's Guide


THE NETWORK IS THE COMPUTER™

A Sun Microsystems, Inc. Business
901 San Antonio Road
Palo Alto, CA 94303 USA
650-960-1300 fax 650-969-9131

Part No.:805-4378-10
Revision A, September 1998

# Contents

# Figures

# Tables

# Preface

*Sun™ Internet Mail Server™ 3.5 Administrator's Guide* begins where *Sun Internet Mail Server 3.5 Installation Guide* ends. Use this guide in conjunction with *Sun Internet Mail Server 3.5 Reference Guide* to fine-tune the default configuration, maintain, monitor, and troubleshoot your mail server after installing the software and loading the user and distribution lists from your existing email system.

## SIMS User Registration

Register as a user of the Sun Internet Mail Server 3.5 (SIMS) to receive information about new releases, upgrade offers, and promotions. To register, press the *Registration* button at the Administration Console login page. Fill-in the form requesting your name, address, e-mail address, and other information, and press *Send*. When Sun receives the completed registration form, we will email an acknowledgment back to you. You must provide an email address in order to receive a confirmation notice.

## Error Conditions

Registration errors are rare, but the following table describes the possible error messages and the required action.

**TABLE P-1** Registration Error Conditions and Required Action

| Error Message | Required Action |
|---|---|
| “The server could not set a locale to encode the mail. There was no locale supplied and the server could not set the default properly.” | Make sure that you start registration from the Admin console screen. |
| “The server could not obtain the <LOCALE> locale that you registered from to properly format the mail. It is necessary to have the same locale installed on the server that you registered from.” | Either make sure the locale installed on server is the same as the locale you are registering from on your client or, type in registration in us-ascii. |
| “The mail program on the server could not be opened.” | There was an error involving the sendmail program. Make sure that /usr/lib/sendmail is on your system and properly configured. |
| “There was not enough memory to process the mail.” | You've run out of swap space. Shut down applications or increase swap and try again. |

# Who Should Use This Book

This book is intended for the following two audiences:

- System administrators experienced with Solaris™ who manage a network comprised of Sun Workstations™, PCs, Macintoshes, or IBM mainframes that share resources. This system administrator has previous experience planning, installing, configuring, maintaining, and troubleshooting an enterprise email system.
- Moderately technical system administrators with some Solaris experience who manage a network comprised of Sun Workstations, PCs, and Macintoshes that share resources. This system administrator does not have previous experience planning, installing, configuring, maintaining, and troubleshooting an email system.

# Before You Read This Book

Before performing the tasks described in this book, you should have installed the mail server software and loaded the users and distribution lists from your existing email system if applicable per information provided in the *Sun Internet Mail Server 3.5 Installation Guide*.

# How This Book Is Organized

**Chapter 1, "Concepts,**" provides conceptual information on the SIMS components and features.

**Chapter 2, "The Administration Console Road Map,"** provides overview information on the Administration Console as well as a road map for Admin Console documentation.

**Chapter 3, "User Management"** describes how to add, delete, or modify user, group or organizational units in the Sun Directory Service.

**Chapter 4, "Internet Message Transport Agent (IMTA) Administration"** provides step-by-step instructions for changing the message transport characteristics of SIMS.

**Chapter 5, "Message Store Administration"** describes step-by-step instructions for changing the Sun Message Store characteristics of SIMS.

**Chapter 6, "Sun Directory Services Administration"** provides step-by-step instructions for viewing and modifying the Sun Directory Services (SDS) also known as the LDAP Server or simply the Directory Service.

**Chapter 7, "Maintenance,"** provides procedures and background concepts that enable you to perform scheduled or as-needed maintenance.

**Chapter 8, "Troubleshooting,"** describes tools that enable you to troubleshoot your mail server, and provides some troubleshooting procedures.

**Chapter 9, "Secure Sockets Layer (SSL) Support in SIMS,"** describes how to use the SSL security features supported by SIMS.

**Chapter 10, "User Administration,"** describes procedures available to users such as starting a vacation program, changing passwords, mail forwarding, and so on.

**Appendix A, "Configuring SIMS as a Proxy Message Access Server"**describes SIMS message access proxy.

**Appendix B, “Replication Configuring—Examples”** provides examples of how to configure replication.

**Appendix C, “Populating the Directory Examples”** describes three examples of populating the directory.

**Appendix D, “SIMS Directory Schema and Directory Information Tree”** describes SIMS directory information tree (DIT) requirements and the type and format of objects and attribute values for directory entries required by SIMS.

**Appendix E, “Error Messages”** lists error messages and the appropriate actions.

**Glossary** is a list of words and phrases found in this book and their definitions.

# Related Information

The *Sun Internet Mail Server 3.5 Administrator’s Guide* is a companion document to the following manuals in the SIMS documentation set:

*Sun Internet Mail Server 3.5 Installation Guide*—describes the planning and installation procedures for the SIMS software on Solaris SPARC and Intel-based x86 systems. In particular, it describes the installation of the software using the graphical user interface.

*Sun Internet Mail Server 3.5 Reference Guide*—provides detailed information on command line options, administrator-editable configuration files, system architecture, supported standards, and location of software files.

*Sun Messaging Connectivity Services Channel Guides*—describes how to connect SIMS with cc:Mail, Microsoft Mail, or PROFS mails systems.

For additional up-to-date product information, refer to the Sun Internet Mail Server web site. The URL is `http://www.sun.com/sims`. At this location, you will find the related information:

- Press releases and data sheets
- Technical white papers
- Product documentation
- Product demos
- Product Frequently Asked Questions (FAQs)
- Links to third party client software

# Topics Not Covered

*Sun Internet Mail Server 3.5 Administrator's Guide* does not cover the following topics:

- Solaris administration topics
- HotJava™ topics

# What Typographic Changes Mean

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

**TABLE P-2** Typographic Conventions

| Typeface or Symbol | Meaning | Example |
|---|---|---|
| `AaBbCc123` | The names of commands, files, and directories; on-screen computer output | Edit your `.login` file.<br>Use `ls -a` to list all files.<br>`machine_name% You have mail.` |
| **`AaBbCc123`** | What you type, contrasted with on-screen computer output | `machine_name% su`<br>`Password:` |
| *AaBbCc123* | Command-line placeholder: replace with a real name or value | To delete a file, type `rm` *filename.* |
| *AaBbCc123* | Book titles, new words or terms, or words to be emphasized | Read Chapter 6 in *User's Guide.* These are called *class* options.<br>You *must* be root to do this. |

# Path Convention for Screen Navigation

The following is an example of a navigation path. Navigation paths are shown at the beginning of each task. The navigation path is used in the Admin Console graphical user interface to move from the main Admin Console screen to the screen where the task is performed.

AdminConsole>Sun Message Store>Purge Options

Using the navigation path above, begin at the main Admin Console screen, shown immediately after log in. Then, click on Sun Message Store to view the next screen. Written directions accompanying the path direct you to click on the Purge Option listing of the Sections List. Follow the written instructions for configuring the purge options.

# Shell Prompts in Command Examples

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

**TABLE P-3** Shell Prompts

| Shell | Prompt |
|---|---|
| C shell prompt | `machine_name%` |
| C shell superuser prompt | `machine_name#` |
| Bourne shell and Korn shell prompt | `$` |
| Bourne shell and Korn shell superuser prompt | # |

# Graphical User Interface Conventions

This section describes terminology and other conventions used when discussing the Administration Console, a graphical user interface.

## Terminology

The following table defines terms used in procedures associated with the Administration Console.

**TABLE P-4** Graphical User Interface Terminology

| Term | Explanation | Example |
|---|---|---|
| Check box | A yes / no or on / off control. A square box that appears highlighted and pushed in when on or pushed out when off. Usually, all check boxes in a group can be selected. | To enable the logging of each message, click the check box. |
| Radio button | A yes / no or on / off control. A diamond or circle that appears highlighted and pushed in when on or pushed out when off. Usually, only one radio button in a group can be selected. | You can enable the channel to ignore nonstandard encoding headers by clicking the appropriate radio button. |
| Click | Press and release a mouse button without moving the pointer. | Click the radio button. |
| Double-click | Click a mouse button twice quickly without moving the pointer. | Double-click the SMTP channel name from the list of channels to display the SMTP property book. |

# Product Packaging

The consolidated Sun Internet Mail Server components are comprised of two main product packages:

- Departmental Edition
- Enterprise Edition

# What Is Sun Internet Mail Server?

Sun Internet Mail Server (SIMS) is an extensible framework of independent modules that create an enterprise-wide, open standards-based, scalable electronic message handling system.

A message handling system is the combination of message user and transfer agents, message stores, and access units that together provide electronic mail.

The components of this message handling system are:

- **Message Access and Store**. Message Access and Message Store are the repositories of user messages, and the means to retrieve and process those messages. Sun Internet Mail Server supports both the Internet Message Access Protocol version 4 (IMAP4) and the less flexible but widely-implemented Post Office Protocol version 3 (POP3).

  The primary message store for Sun Internet Mail Server is the Sun Message Store. The mail server also retains support for the Solaris Mailbox Format Store to ease migration for sites with an installed base of traditional `/var/mail` clients.
- **Internet Message Transfer**. The Internet Message Transfer Agent (IMTA) is responsible for the routing, transfer, and delivery of Internet mail messages. The Sun Internet Mail Server includes a fast, scalable, and flexible IMTA. This replaces the Sendmail utility that is bundled with most UNIX® systems and was used in Sun Internet Mail Server 2.0.
- **Sun Directory Services**. The directory is the central repository for meta-information: user profiles, distribution lists, and other system resources. Sun Internet Mail Server is bundled with a dedicated Lightweight Directory Access Protocol (LDAP) directory service.
- **Administration Services**. The administration of Sun Internet Mail Server is a GUI-based configuration, and monitoring environment. Sun Internet Mail Server is based on the Java™ Management Application Program Interface (JMAPI) framework.
- **Sun Messaging Connectivity Services**. Provides batch-mode connectivity to proprietary message transfer systems including the “LAN Mail” systems Lotus cc:Mail and Microsoft Mail, and the mainframe-base IBM OfficeVision (PROFS).

# Key Features

Key features of the product are:

- Client/server architecture
- High performance and high scalability
- Internet open standards
- Native support for nomadic and disconnected operation
- Integrated directory service
- Dependable transfer and delivery
- Support for non-Internet email systems
- Comprehensive easy-to-use GUI-based administration

The consolidated SIMS components are comprised of two main product packages:

- Departmental Edition
- Enterprise Edition

## Departmental Edition

The *Departmental Edition*, referred to as the Sun Internet Mail Server - Departmental Edition, is intended for local departmental environments that do not need the scalability and extensive configurability of an enterprise server. The departmental package performs its own routing and delivery within a local office or department, but hands off interdepartmental mail to an enterprise or backbone server. The package is simple to install and configure, and requires minimal operator intervention. The key features of the Departmental Edition are:

- Multi-threaded IMAP4 and POP3 servers. These are optimized for up to 500 simultaneously connected IMAP users.
- Internet Message Transfer Agent (IMTA). The IMTA is restricted to two external Simple Mail Transfer Protocol (SMTP) channels. One channel is dedicated to the local intranet, and the other to a “smart host” backbone or firewall connection.
- Server daemon processes that are spawned by the multiprocessing `inetd` utility instead of the Dispatcher.
- Multiprocessing LDAP directory service.
- Full featured Sun Message Store (SMS).
- Centralized GUI Administration Console.
- Interface to Solaris Mailbox Format (`/var/mail`) message store.
- Supports Internet standards mail protocols.
- Integrated backup and restore.
- SSL protocol security for IMAP and POP servers.
- IMAP and POP proxy daemons.
- MAPI providers.

- Sun Web Access clients.

## Enterprise Edition

The Enterprise Edition, also referred to as the Sun Internet Mail Server - Enterprise Edition, provides a full featured messaging server for large user communities, enterprise backbone management, and Internet firewall applications. The key features of the Sun Internet Mail Server - Enterprise Edition include all the features of the Departmental Edition plus the following:

- Multi-threaded IMAP4 and POP3 servers. These servers have higher performance and a much smaller footprint than the multiprocessing servers in the standard package. These servers support as many as 10,000 simultaneous connections and 100,000 mailboxes on an E3000-class server.
- Full-featured IMTA. This includes extensive address rewriting and channel management facilities.
- Multiprocessing LDAP directory service.
- Server daemon processes that are managed by the Dispatcher instead of `inetd`, for better scalability.
- Pipe channels that support extensibility of the IMTA through native UNIX system scripts.
- High performance databases that replace internal flat files for the user cache, distribution lists, mappings, and forwarding.
- Multiprocessing hardware support.
- Anti-spamming.
- Anti-relay.
- Multiple configurable channels.
- Sun SDK APIs available for custom channel development.
- SMCS (Sun Messaging Connectivity Services), for the Microsoft Mail, cc:Mail, and PROFS channels.
- Asymmetric HA.
- MAPI providers.
- Sun Web Access clients.

CHAPTER 1

# Concepts



This chapter provides conceptual information on SIMS components and architecture.

## Mail Server Authorities

The Sun Internet Mail Server features two levels of security: one level from the directory service and another from the UNIX® file system. Specifically, you must access permission from the following authorities:

- Directory service – Requires user name and encrypted password to access the Administration Console.
- UNIX file system:
  - `Inetmail` (owner of Sun Internet Message Store and configuration files) – Requires `root` permission to invoke SIMS utilities and commands from the command line.
  - Solaris format mailbox files (owned by individual users) – Requires either ownership of the mailbox to be manipulated or `root` permission.

Note that when a client machine binds to a directory server, the password used in the bind request is passed in clear text rather than encrypted text.

# SIMS Administration



The administration of the SIMS is based on the Java™ Management Application Programming Interface (JMAPI) architecture. JMAPI is a collection of Java language classes that enable a diverse set of autonomous applications to be brought together under a common look, feel, and behavior everywhere they run. Such is the case with the various SIMS components being integrated into one centralized administrative service.

The administrative service enables you to fine-tune the default configuration, maintain, monitor, and troubleshoot the SIMS components. It is composed of the following elements:

- Administration Console
- Administration Server
- SIMS components
- Remote Method Invocation (RMI)

FIGURE 1-1 illustrates these elements and the following sections provide further explanation.

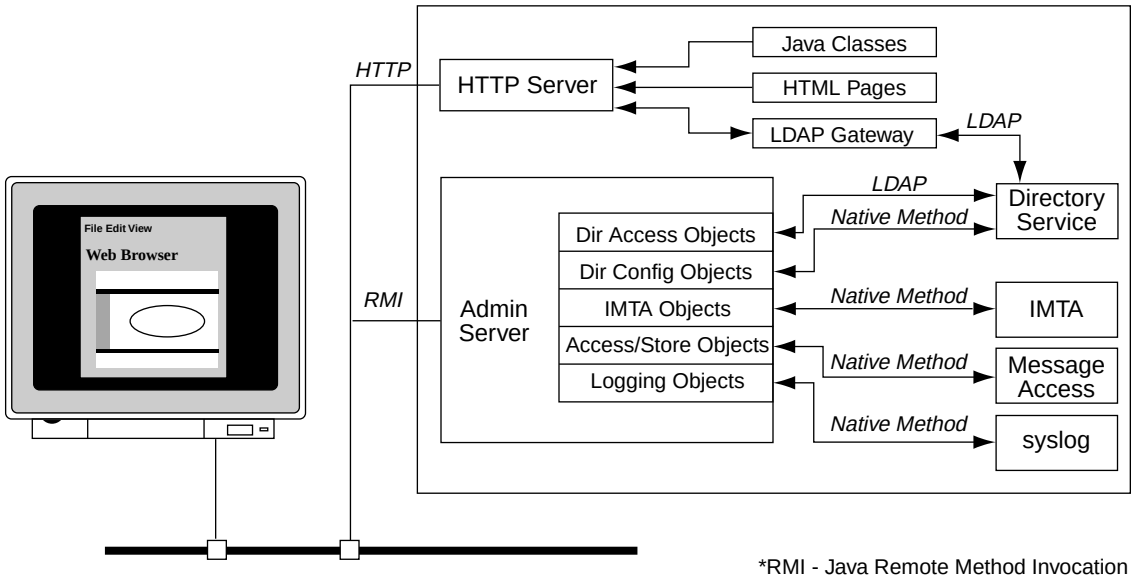



**FIGURE 1-1** Sun Internet Mail Server Administrative Architecture

You can administer SIMS from the same machine that all SIMS components are installed on or, if desired, remotely from any machine on the network. The remote machine must be able to run the HotJava browser provided with SIMS. Other browsers or other versions of HotJava may not work with the Administration Console.

# Administration Console

The Administration Console or *Admin Console* provides the graphical user interface that enables you to configure, maintain, monitor, and troubleshoot the SIMS components. The Admin Console runs on the HotJava browser provided with SIMS. FIGURE 1-2 shows a sample Admin Console page.

---

**Note –** Only one administrator can be logged on to the Administration Console at any given time.

---

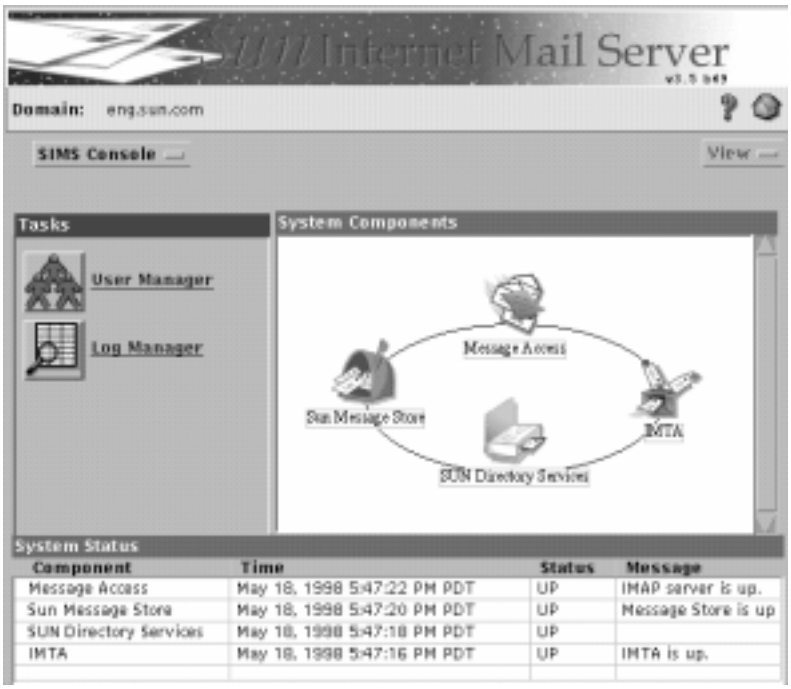



**FIGURE 1-2** Sample Administration Console Page

The applet contains interfaces to the managed objects that reside in the Administration Server. For more information of managed objects, refer to “How Administrative Service Works” on page 5.

For an overview on the Admin Console including information on how to navigate through its pages refer to “The Administration Console Road Map” on page 49.

## Administration Server

The Administration Server contains two relevant elements:

- Managed objects
- HyperText Transfer Protocol (HTTP) server

A managed object is a collection of configurable attributes, for example, a collection of attributes for the directory service. (Note that a managed object does not necessarily map to a SIMS component.)

An HTTP server resides on the Administration Server to provide bootstrap capabilities for certain Java elements. A HyperText Markup Language (HTML) file causes an initial applet and managed objects to be loaded from the Administration Server to the browser. After the initial applet takes control, it uses its managed object interfaces to communicate with the Administration Server.

You do not need to configure or interact in any way with the Administration Server.

## Sun Internet Mail Server Components

Each SIMS component except the directory service (specifically, the directory access object that manages the user and group entries) communicates with the Administration Server using that component's native format. The directory access object interfaces with the Administration Server using the Lightweight Directory Access Protocol (LDAP). (The directory configuration object that manages the configuration of the directory service itself interfaces the Administration Server using the native format.)

## RMI

The Java Remote Method Invocation (RMI) enables the Administration Server and Console, which are running in either different address spaces on the same machine or on different machines, to communicate. RMI enables the remote managed objects that reside on the Administration Server to be manipulated by the managed object interfaces that reside in the Admin Console.

## How Administrative Service Works

The key to understanding SIMS administration is to understand the Java Management Application Programming Interface (JMAPI) concept of managed objects. Managed objects are an abstraction of the actual SIMS component or service they represent. As such, managed objects act as a collection of configurable attributes, for example, a collection of attributes for the directory service component. SIMS managed objects reside on the Administration Server. The configurable attributes that reside on the Administration Server are manipulated by the managed object interfaces on the Administration Console.

For example, imagine that you want to modify the configuration of some aspect of the directory service component. You perform this modification using the Administration Console's graphical user interface. The managed object interface that resides in the Admin Console invokes the specified operation on the managed object that resides in the Administration Server. The managed object in turn calls LDAP to implement the specified operation on the directory component itself.

# Internet Message Transfer Agent



The Internet Message Transfer Agent (IMTA) routes, transports, and delivers Internet mail messages for SIMS.

## Internet Mail Messages

An Internet mail message is composed of the following elements:

- Envelope
- Message

For complete information on message envelopes and headers, respectively, refer to RFCs 821 and 822.

### Message Envelope

The IMTA uses the contents of the envelope to make routing decisions. It does not use the content of the message. The content of the envelope is primarily defined RFC 821. It includes the originator address, the recipient(s) address(es), and envelope ID. The IMTA supports additional envelope information related to SMTP service

extensions published after RFE 821 such as notary (the ability to specify requested Delivery Status Notification for each recipient, see RFE 1891) and original recipient addresses

## Message

RFC 822 defines a message as headers and contents.

### *Message Headers*

An Internet mail message starts with one or more headers. Each header is composed of a field name followed by a colon then a value. Values can be generated by the composer of a message, the mail client, and IMTAs. Headers contain the following types of information about the message:

- Delivery information (for example, TO, CC, BCC, From, reply-to, in-reply-to)
- Summaries of the content (for example, subject, keywords, comments).
- Information that enables tracing of a message if problems occur (for example, message-ID, references, received, return-path).
- Information specific to the Multipurpose Internet Mail Extensions (MIME). For more information on MIME and MIME-specific headers, refer to Message Content / MIME.

RFC 822 defines several headers. Not all defined headers need to be present in a message. In fact, only a few headers are required for any type of message.

The administrator of a mail client can construct a template of desired headers that the composer of a new, forwarded, or replied-to message fills in. The following is an example of a simple template of headers for a new message:

To:
Subject:
Date:
Cc:

The mail client can automatically generate some headers (From, reply-to, message-ID, and references).

Before the message is submitted to the IMTA, the mail client can add a date header. If the From header contains multiple email addresses or if the email address is different than the submitting mail client, then the sender header is added.

An IMTA can also add headers to a message. Each IMTA that accepts a message adds a received header to that message. The last IMTA to accept the message and to actually deliver the message to the message store adds a return-path header. The received and return-path headers provides information that enables you to trace the routing path taken by the message if a problem occurs.

### *Message Content/MIME*

A blank line separates the headers and the content or body of the message. The content or body of the message provides the data that the originator of the message intends to transmit to the recipient.

SIMS supports Multipurpose Internet Mail Extensions (MIME). Whereas RFC 822 is limited to handling text messages of a single body part, MIME extends RFC 822 to handle multiple body parts.

Therefore, the content of a message can include text as well as images, audio, video, and binary or application-specific files. The text included in the message content has the following characteristics:

- Unstructured or structured
- Unlimited line length or overall length
- Non-ASCII character sets, which allows non-English language text
- Multiple fonts

If included, the images, audio, video, and binary or application-specific files appear as attachments.

MIME defines the following headers that can appear at the start of an Internet mail message:

- MIME-version – Specifies a version number to indicate that a message format conforms to the MIME standard
- Content-type – Specifies the type or subtype of data in the content or body of a message. Possible values include the following
  - Text – Indicates data that is principally text
  - Multi-part – Indicates a message consisting of multiple body parts, each having its own data type
  - Application – Indicates either application or binary data
  - Message – Indicates an encapsulated message
  - Image – Indicates still image (picture) data
  - Audio – Indicates audio or voice data
  - Video – Indicates video or moving image data, possibly with audio as part of the composite video data format

- Content-transfer-encoding – Specifies how the data is encoded so that the data can traverse Internet Message Transfer Agents (IMTAs) outside of the SIMS email system that may have data or character set limitations
- Content-ID – Specifies an ID for a message content or body
- Content-description – Text that provides descriptive information about the message content or body

For complete information on MIME, refer to RFCs 1521, 2045, 2046, 2047, 2048, and 2049.

## Channels

A channel is an interface with another SIMS component, another email system, or mail user agent. The actual hardware connection or software transport or both may vary widely from one channel to the next.

Each channel consists of up to two channel programs and an outgoing message queue for storing messages that are destined to be sent to one or more of the interfaces associated with the channel. Channel programs perform two functions:

- They transmit messages to other interfaces, deleting them from their queue after they are sent.
- They accept messages from other interfaces, placing them or *enqueueing* them into channel queues. Note that while a channel program only removes messages from its own queue, it can enqueue messages in any queue, including its own.

A channel program that initiates a message transfer to another interface on its own is called a *master program*. A program that accepts transfers initiated by another interface is called a *slave program*. A channel can be served by a master program, a slave program or both. Some of the default channels provided are:

- SMTP channel: TCP/IP-based message delivery and receipt.
- Pipe channel: used for alternative message delivery programs.
- Message Store channel: delivers mail to the Sun Message Store.
- Reprocessing channel: used for messages that are resubmitted due to transient failures during delivery.
- Defragmentation channel: reassembles partial messages into the original complete message.
- Conversion channel: performs body part-by-body part conversion on messages.
- Local channel: delivers mail to `/var/mail` for backward compatibility.

### SMTP Channels

SMTP channels differ according to the set of SMTP hosts with which they communicate. One type of SMTP channel is dedicated to relaying mail to and from one single particular host. For example, if the IMTA is installed within a Firewall, all mail addressed to external users must transit through the Firewall host. Therefore, an SMTP channel must be configured to handle messages between the IMTA and this Firewall host. This type of channel is referred to as an *SMTP router channel*.

A second type of SMTP channel is dedicated to relaying messages between the IMTA and a group of SMTP hosts defined by the knowledge of the DNS. In this type of channel, envelope recipient addresses are used to determine the destination host(s) of the message. This type of channel is called *SMTP intranet* or *SMTP internet channel*, depending on whether they are used to exchange mail with hosts inside or outside of your mail network.

### Local Channel

The *local channel* has two purposes. The main purpose is to determine the delivery options of local users. The other purpose is to deliver mail to Solaris Operating Environment mailboxes (`/var/mail` files).

### Message Store Channel

The message store channel is used to deliver messages to the Sun Message Store.

### The Pipe Channel

The pipe channel can be used to deliver messages through programs provided and configured by the site administrator (alternative message delivery programs). The pipe channel can be used to implement interfaces with other systems/components that are not provided with SIMS to invoke mail filtering programs, or mail auto-responders/auto-forwarders.

SIMS currently uses the pipe channel to implement `autoreply`-based functions such as the `vacation` utility.Refer to “Alternative Delivery Programs” on page 102 for information on how to make delivery programs available to users.

## Domain Name System

Before installing SIMS, the Domain Name System must be installed. DNS must be used as the primary hostname resolution service.

# Internet Message Transfer Agent-Directory Cache

The delivery and routing of messages by the Internet Message Transfer Agent (IMTA) is based on the user and group (distribution list) entries stored in the directory service. The IMTA needs to access the directory information for each message that it processes. Rather than querying the directory service each time it processes a message, the IMTA *caches* the directory information, or takes a snapshot of the directory information and stores it, and accesses directory information in the cache. The IMTA implements the cache for the following reasons:

- Performance – Performing a directory query for each recipient of each message is time-consuming and puts a large load on the mail server.
- Data formatting – The information stored in the directory service is not always in the format needed by the IMTA. When creating the cache, the IMTA reformats the directory information.

The directory information stored in the directory service is continuously updated. As a result, the directory information in the IMTA-directory cache must be updated periodically with the current directory information in the directory service or *synchronized*. Two types of synchronization exist:

- Full synchronization – The existing cache is replaced with a new cache, completely rebuilt with the current user and group entries from the directory service. After the synchronization occurs, the IMTA configuration file is rebuilt then the IMTA is automatically restarted.
- Incremental synchronization – The existing cache is updated with user and group entries that were created or modified since the last full or incremental synchronization. The IMTA is not restarted.

TABLE 1-1 outlines the updates to the IMTA-directory cache that are and are not performed during a full synchronization and an incremental synchronization.

**TABLE 1-1** Updates Performed During Full/Incremental Synchronizations

| IMTA-Directory Cache Update | Performed During Full Synchronization? | Performed During Incremental Synchronization? |
|---|---|---|
| New user entries added | Yes | Yes |
| Modified user entries updated | Yes | Yes |
| Deleted user entries removed | Yes | No |
| New members added to existing distribution lists | Yes | Yes |

**TABLE 1-1** Updates Performed During Full/Incremental Synchronizations *(Continued)*

| IMTA-Directory Cache Update | Performed During Full Synchronization? | Performed During Incremental Synchronization? |
|---|---|---|
| Deleted members removed from existing distribution lists | Yes | Yes |
| Modification of access control info | Yes | No |
| New distribution lists added | Yes | Yes |
| Deleted distribution lists removed | Yes | No |

For more information on user and group entry attributes, refer to “User Entries” on page 29. For more information on distribution lists, refer to “Distribution Lists” on page 18.

## Cache Synchronization Schedule Planning

By default, the Internet Message Transfer Agent (IMTA)-directory cache is fully synchronized every day at 2:00 am and incrementally synchronized every hour.

The Administration Console enables you to reconfigure the synchronization schedule. Before reconfiguring this schedule, you must consider the following:

- A full synchronization requires that the IMTA be restarted, an operation that is performed automatically. Since restarting the IMTA is a CPU-intensive operation and will temporarily affect the overall mail server performance, Sun recommends scheduling full synchronizations at times when you anticipate that the mail server load is light, for example, during lunch hour or in the middle of the night.
- An incremental synchronization does not burden the CPU to the degree that a full synchronization does; in fact, the more often incremental synchronizations are performed, the less it burdens the CPU. However, an incremental synchronization does use CPU cycles and you do not want to schedule this operation more than necessary.

Depending on the number of users (mailboxes) your mail server services, scheduling one to six full synchronizations per day and incremental synchronizations every 5 to 30 minutes is sufficient.

---

**Note –** You must schedule directory caches on each mail server to be fully or incrementally resynchronized at the same time. Not doing so could cause routing loops to occur.

---

For information on reconfiguring the synchronization schedule using the Administration Console, refer to “Alias Synchronization Schedule” on page 104.

# Rewrite Rules

When a message enters an Internet Message Transfer Agent (IMTA) channel, it must be placed in the correct channel queue and subsequently routed to the correct destination for delivery. The domain rewriting rules (from this point forward called the “*rewrite rules*”) are a tool that the IMTA uses to route messages to the correct host. Rewrite rules perform the following functions:

- Extract the host / domain specification from an address of an incoming message
- Match the host / domain specification with a rewrite rule pattern
- Rewrite the host / domain specification based on the domain template
- Decide the IMTA channel queue in which the message should be placed

The following sections walk you through the rewrite rule process. They also discuss the following elements of the rewrite rule itself:

- Pattern – A string composed of ASCII characters that the host / domain specification can potentially match
- Domain template – A template that defines how the host / domain specification is rewritten
- Routing system – The destination channel
- Controls

## Extracting the Host / Domain Specification of An Address

When a message enters a channel, all addresses on the envelope and in the message header are examined and the host / domain specification of the address is extracted. The host / domain specification is the part of the address that is to the right of the at (@) sign. For example, in the address `john@corp.acme.com`, `corp.acme.com` is the host / domain specification.

## Matching Host / Domain Specification With A Rewrite Rule Pattern

The channel scans for a match between the extracted host / domain specification and the pattern portion of the first rewrite rule in the list. (The channel scans the host / domain specification from left to right—for example, starting with corp, then acme, then com—and the rewrite rules list from top to bottom.) If a match is not found in

the first rule, the channel scans the next rule and so on. If a match is found, the host/domain specification is rewritten per the domain template of the rewrite rule. If a match is not found, the message is returned to the sender.

TABLE 1-2 outlines the types of rewrite rule patterns that your rewrite rule list can potentially contain and the order in which each type of pattern is scanned when the input address is `joe@sc.cs.cmu.edu`.

**TABLE 1-2** Rewrite Rule Pattern Types (listed in the order in which pattern is scanned).

| Example of Pattern Scanned For* | Explanation |
|---|---|
| sc.cs.cmu.edu | matches sc.cs.cmu.edu only |
| *.cs.cmu.edu | matches <any one token>.cs.cmu.edu only |
| .cs.cmu.edu | matches <multiple tokens>.cs.cmu.edu only |
| *.*.cmu.edu | matches <any one token>.<any one token>.cmu.edu only |
| .cmu.edu | matches <multiple tokens>.cmu.edu only |
| *.*.*.edu | matches <any one token>.<any one token>.<any one token>.edu only |
| .edu | matches <multiple tokens>.edu only |
| *.*.*.* | matches <any one token>.<any one token>.<any one token>.<any one token> only |
| . | matches anything |

*The example patterns shown in this column are not meant to imply that you can scan for patterns with a fixed number of address tokens.

**Note –** Your channel does not necessarily contain a rewrite rule and, subsequently, a rewrite rule pattern for each of the patterns described in TABLE 1-2. However, if your channel does contain a rewrite rule for each of the described patterns, then TABLE 1-2 outlines the default order in which each pattern is scanned for until a match is found. That is, whatever the order the rules are written, they are scanned from most specific to least specific.

The rewrite rule list for a channel can include multiple rewrite rules containing the same pattern but different domain templates. You can reconfigure the order of these types of rewrite rules so that the channel scans these rules in the reconfigured order. This features enables you to fine-tune the rewrite rule list with your preference as to how the domain/host specification in these types of rules is rewritten. You cannot reconfigure the basic order in which the rewrite rules in TABLE 1-2 are scanned.

To illustrate how the host/domain specification and rewrite rule matching process works, if the extracted host/domain specification is

zoo.cmu.edu

and the following patterns are defined in the rewrite rules:

eng.cmu.edu

*.cmu.edu

then a match would result after the second scan with *.cmu.edu.

For information on how to configure rewrite rules, refer to “Rewrite Rules” on page 132.

## Rewriting the Host / Domain Specification

If a match is made between the host / domain specification of an address and the pattern portion of a rewrite rule, the host / domain specification is rewritten according to the domain template portion of the rewrite rule.

The *domain template* defines how the host / domain specification is rewritten. The template can be composed of one or a combination of the following elements:

- A full static host / domain specification, for example, corp.acme.com, or a portion of a host / domain specification (a portion of the *address tokens* or elements set off by decimals), for example,.com.
- A single field substitution string that dynamically rewrites one address token of the host / domain specification represented by a wildcard character (*). The address token to be rewritten can be the portion of the address that did not match the rewrite rule pattern or the portion that matched the wildcard character. The rewriting of a host / domain specification is based on the contents of the specification itself. The template can include multiple field substitution strings.

---

**Note –** The IMTA allows for many more field substitution strings. Refer to the *SIMS Reference Manual* for more information.

---

The syntax of the field substitution string is $&*n*.

where *n* is an integer from 0 to infinity. *n* represents the unmatched or wildcard address token that is to be rewritten. From left to right, the leftmost address token is represented by the integer 0; the second from the left is represented by the integer 1, and so on.

For example, imagine that a host / domain specification matches the following rewrite rule pattern:

*.*.com

The corresponding domain template consists of the following:

$&0.$&1.com

In this domain template, $&0 and $&1 are the field substitution strings and com is the static portion. As mentioned earlier in this section, an address is rewritten based on the contents of the host / domain specification itself. Using the same example, imagine that the incoming address is

john@corp.acme.com

The host / domain specification is rewritten as follows:

corp.acme.com

In this domain template, $&0 is rewritten as corp, and $&1 is rewritten as acme.The address tokens corp. and acme. were taken from the host / domain specification.

Although in this example, a three-token host / domain specification is rewritten as a three-token host / domain specification, this specification can be rewritten with any number of tokens desired. Using the same example, except with the domain template

finance.$&0.$&1.com

the incoming address john@corp.acme.com can be rewritten as john@finance.corp.acme.com.

Also note that the value of *n* in the field substitution string of the domain template must correspond to the number of unmatched or wildcard address tokens in the rewrite rule pattern.In the example used above, the string $&1 implies that there are two unmatched or wildcard address tokens in the rewrite rule pattern, as is the case (*.*.com).

If the value of *n* does not correspond to the number of unmatched or wildcard address tokens, for example, a string of $&1 exists in the domain template (implies two unmatched or wildcard address tokens) but the rewrite rule pattern contains only one unmatched or wildcard address token (*.acme.com), then the rewrite rule will not work. The host / domain specification will not be rewritten.

## Mapping a Rewritten Address to a Destination Channel

The last element of a host / domain rewrite rule is the *routing system* or destination channel in whose queue a message should be placed for delivery.

## Rewrite Rule Controls

By default, a rewrite rule is scanned for all header and envelope addresses. It is possible to specify that a given rule applies to a subset of all the addresses. For example, a rule can search for some combination (AND) of the “To:” “From:” “CC:” and “Bcc:” header addresses and the envelope recipient addresses (RCPT TO). Additional filters (control sequences) are available and are documented in the *SIMS Reference Manual*.

## Default Channel Rewrite Rules

During installation of SIMS, the system generates default rewrite rules for each of the installed channels. You can modify the default rewrite rules or add new rewrite rules for each of the installed channels using the Rewrite Rules section in the IMTA property book of the Administration Console.

When you create a new channel (applies to Sun Internet Mail Server - Enterprise Edition only), the system does not generate default rewrite rules. After you create the new channel, the Administration Console prompts you to add new rewrite rules.

For complete information on installing SIMS, refer to the *SIMS Installation Guide*. For information on creating a new channel, refer to “To Create a Channel” on page 120.

## Adding and Reconfiguring Rewrite Rules

The Administration Console enables you to add a rewrite rule to an existing or newly created channel. It also enables you to delete or modify an existing rewrite rule associated with an existing channel. For more information on adding and reconfiguring the rewrite rules associated with a channel using the Administration Console, refer to “Rewrite Rules” on page 132.

# Controlling SMTP Email Access

An individual can intentionally or inadvertently overwhelm your mail server by flooding it with messages. This type of act is called a *denial of service attack.* If a denial of service attack is perpetrated against your mail server, either there may be a substantial impact to the throughput of your mail server or your mail server will become overloaded and nonfunctional.

SIMS provides features that enable you to minimize the possibility of a denial of service attack and well as help you control spam (unwanted or unsolicited email):

- Email access restrictions – Enables you to specify which incoming messages are accepted or denied based on recipient or originating domain, client IP address, server IP address, or originating email address. (See “SMTP Access and Relay Restrictions” on page 106).
- Message size limits – Enables you impose a limit at which a message is deemed too large and rejected by a channel. (See “Message Limitation” on page 125.)

## Distribution Lists

The directory service stores and manages distribution lists as group entries. Each group entry is composed of attributes, for example, distribution list name, members, and so on. For example, imagine that the widget team of the Alpha Corporation is composed of the following users with associated email addresses:

- Jane: jane@eng.alpha.com
- Bernie: bernie@eng.alpha.com
- Kevin: kevin@eng.alpha.com
- Amy: amy@eng.alpha.com
- Frank: frank@eng.alpha.com

You can configure a distribution list composed of each widget team member’s email address along with the associated email address widget@eng.alpha.com. In addition to distribution list members, you must also configure a distribution list owner. An *owner* is an individual who is responsible for the distribution list. An owner can add or delete distribution list members.

You can optionally configure a distribution list moderator. A *moderator* is an individual, usually the distribution list owner, who initially receives a message addressed to a distribution list. Upon receipt of a message, the moderator can forward the message to the distribution list, edit the message and then forward it, or not forward the message.

You can specify distribution list *Access Control, i.e.,* what domains and users can or cannot distribute mail to the group. By default, if a moderator is created, only the moderator can send mail to the group and all other submissions go to the moderator. If no moderator is specified, then anyone can send mail to all the group members.

The *Send Error Conditions To* field enables you to specify an individual to receive a message if an error condition with a distribution list arises. An example of an error condition is when a message addressed to the distribution list cannot be delivered. Upon receipt of one of these messages, it is the specified individual’s responsibility to notify the email administrator about the error. (You can specify the email administrator as the recipient of these messages.)

The *Send Request Messages To* field enables you to specify an individual to receive messages containing requests to be added as a distribution list member. Upon receipt of one of these message, it is the specified individual’s responsibility to notify

the email administrator about adding the requestor as a distribution list member. Requests are sent to `<list name>_request@domain`. (As with the *Send Error Conditions To* field, you can specify the email administrator as the recipient of these messages.)

If you don't configure the *Send Error Conditions To* and *Send Request Messages To* fields, then by default, the individual who is configured as the owner of the distribution list will receive the respective messages.

For details on group entry fields, see "Group Entries" on page 32. For information on configuring group entries, see "To Modify a Group Entry" on page 82.

## Distribution List Process

The Internet Message Transfer Agent (IMTA) retrieves the group entry attributes from the directory service (via the IMTA-directory cache) when it processes a distribution list. The distribution list process is composed of the following phases:

1. Routing if necessary
2. Address processing
3. Message expansion
4. Routing or delivery

Imagine that Alpha Corporation employee Steve, who is in the finance department, sends a message to distribution list widget@eng.alpha.com. The widget distribution list is composed of a group of fellow Alpha Corporation employees, who develop widgets in the engineering department.

The first phase in the distribution list process applies only to email systems that contain multiple mail servers and therefore IMTAs. If your email system has only one IMTA, then this phase does not apply. The IMTA that accepts Steve's message determines if the distribution list widget@eng.alpha.com is stored as a group entry on this particular mail server. If not, the IMTA checks its data base to determine which IMTA in the email system stores this particular group entry. If the IMTA finds this information in its data base, the message is routed to the appropriate IMTA. If the IMTA does not find the information in its data base, the message is routed to the IMTA configured as the router or smart host in that particular domain. Once the message is accepted by the mail server that stores the group entry for widget@eng.alpha.com, the IMTA processes each distribution list member's address (rewrite rules, access control, and so on).

The IMTA then *expands* the message or converts the message into enough copies for each distribution list member. Rather than converting the message into one copy per distribution list member, the IMTA converts the message into one copy per IMTA channel that will deliver or route the message to a distribution list member. For

example, if two distribution list members have mailboxes in `/var/mail` and two in the Sun Message Store, the IMTA will convert the message into two copies: one for the `/var/mail` channel and one for the Sun Message Store channel.

The IMTA then places a copy of the message in the queue of each IMTA channel that will deliver or route the message to a distribution list member.

## Access Control

The Sun Internet Mail Server - Enterprise Edition supports a distribution list access control feature. If you do not implement the access control feature, any email user who knows the email address can send messages to a particular distribution list.

The access control feature enables you to configure who can send messages to a particular distribution list, thereby controlling the quantity of messages as well as the quality of information that can be sent to a distribution list. You can control access to a distribution list in the following ways:

- Submitter – You can specify a list of submitters (users or groups) who are authorized to send messages to a particular distribution list and/or a list of submitters (users or groups) who are unauthorized.
- Domain – Specify a list of domains authorized to send messages to a particular distribution list and/or a list of domains that are unauthorized to send messages to a particular distribution list.

You can specify submitters who are within the email system that you are configuring or in a different email system. The four possible configured access control lists are examined in the following order:

1. Unauthorized submitter list
2. Authorized submitter list
3. Unauthorized domain list
4. Authorized domain list

FIGURE 1-3 presents the access control process visually.

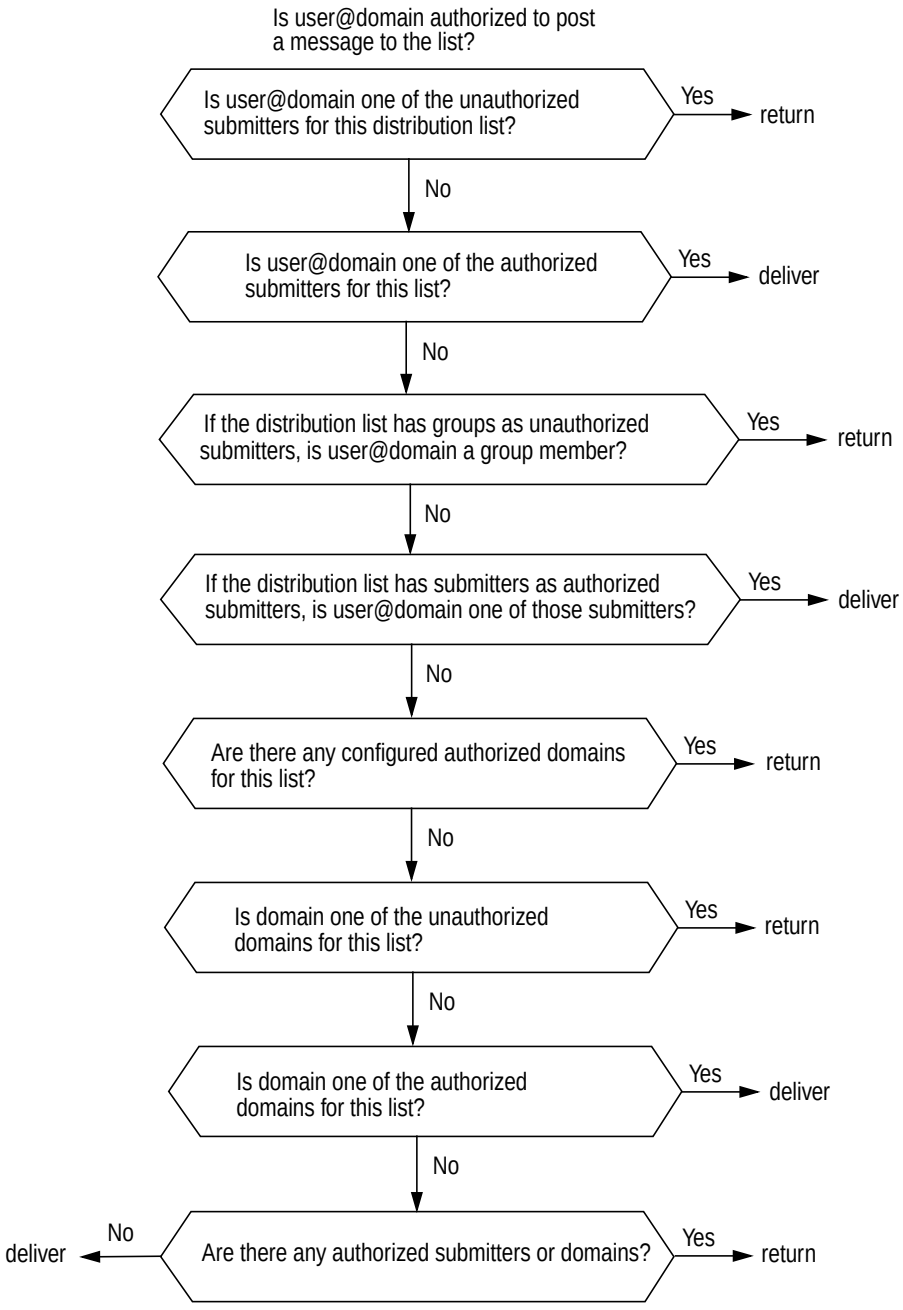



**FIGURE 1-3** Distribution List Access Control Process since SIMS 3.5

If a conflict arises between your configured unauthorized or authorized access control lists, the restriction configured in the submitter list will take precedence over the restriction configured in the domain list. For example, imagine that submitter steve@finance.alpha.com attempts to send a message to the distribution list widget@eng.alpha.com. If the domain finance.alpha.com is on the unauthorized domain list but user Steve is on the authorized submitter list, then the message will be delivered because of the precedence of the submitter lists over the domain lists.

TABLE 1-3 contains various ways that you may want to use the distribution list access control feature and the suggested way to achieve the desired results.

**TABLE 1-3** Implementing Distribution List Access Control Feature

| Restrictions/Allowances You Want to Impose | Suggested Method of Implementing* |
|---|---|
| Restrict all submitters from a particular domain from sending messages | Enter unwanted domain on unauthorized domain list. |
| Allow all submitters from a particular domain to send messages | Enter domain on authorized domain list. |
| Restrict one or more submitters from a particular domain from sending messages and allow all other submitters in the domain to send messages | Enter unwanted submitters on unauthorized submitter list. Enter the domain itself as authorized domain. |
| Allow one or more submitters from a particular domain to send messages and restrict all others from sending messages | Enter desired submitters in the authorized submitter list. Enter the domain itself in the unauthorized domain list. |
| Restrict one or more submitters regardless of domain from sending messages | Enter unwanted submitters in the unauthorized submitter list. |
| Allow one or more submitters regardless of domain to send messages | Enter desired submitters in the authorized submitter list. |
| Restrict submitters to distribution list members only. | Enter distribution list name in the authorized submitter list. |

*Although each restriction / allowance outlined in this table can be handled in multiple ways, the documented suggested methods are the most efficient and require the least impact on performance. For example, you can restrict all submitters from a particular domain from sending messages by entering the unwanted domain on the unauthorized domain list or by not including the unwanted domain on the authorized list and having other authorized entries. However, if you restrict this domain by the latter method, the mail server has to go through two steps in the distribution list process rather than one. Therefore, the latter method is not suggested.

# Sun Message Store



The Sun Message Store is composed of *users*. Each user has an Inbox, where new mail arrives, and can have one or more *folders* or *mailboxes*, where mail can be stored. A folder can contain other folders in a hierarchical tree. Folders owned by a user are *private folders*.

In addition to a user owning a folder, a *common user* or *group* can share the ownership of a folder. This type of folder is called a *shared folder*. For example, the widget design team of the Acme Corporation, collectively known by the email address widget-design@acme.com, can own a folder. A shared folder can be private or public.

A notable feature of the Sun Message Store is that it maintains only one copy of each message. If the Sun Message Store receives a message addressed to multiple users or a group (distribution list), it adds a reference to the message in each user's Inbox rather than having a copy of the message in each user's Inbox, thereby saving disk space. The individual message status (new, unread, replied to, deleted, and the like) is maintained per Mailbox.

FIGURE 1-4 shows the major elements of the Sun Message Store and the flow of messages into the Sun Message Store from the IMTA and to the mail client users.

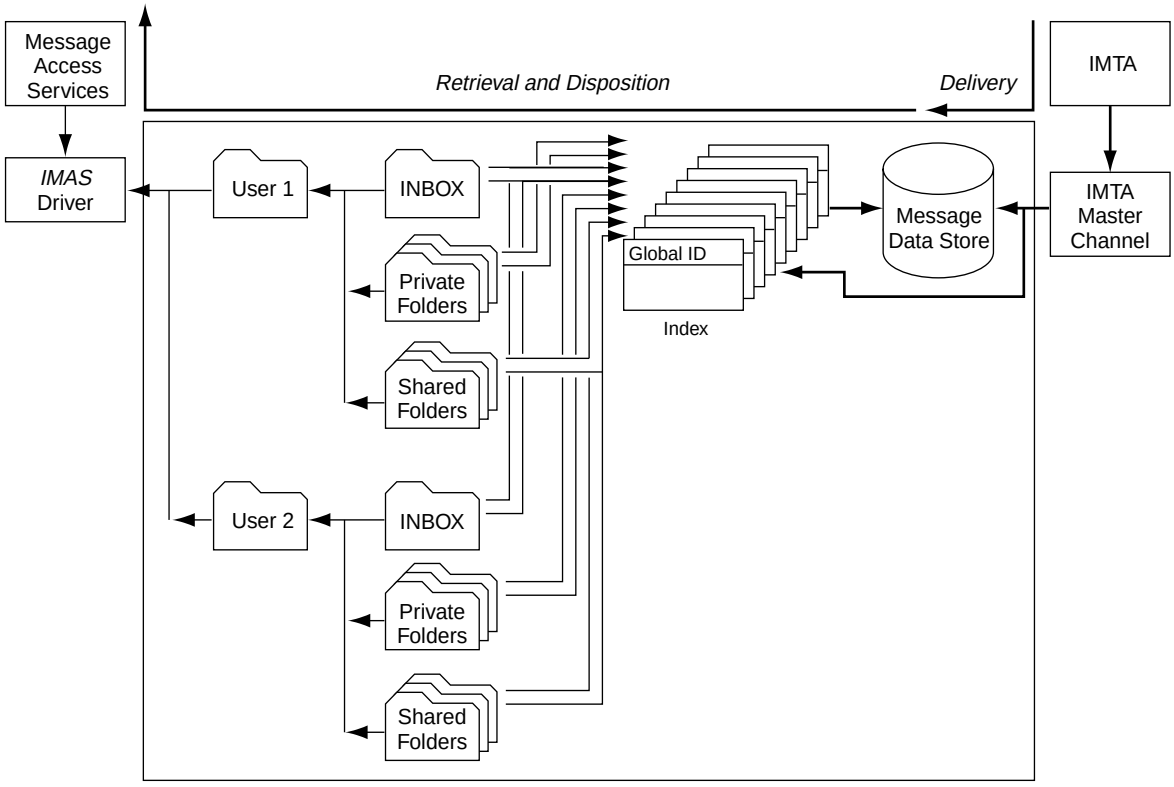



**FIGURE 1-4** Sun Message Store Message Flow

The SIMS also supports `/var/mail`. A site can implement both the Sun Message Store and `/var/mail`. Mail users can access both the Sun Message Store and `/var/mail` using the Internet Mail Access Protocol version 4 (IMAP4) or the Post Office Protocol version 3 (POP3).

Access to the Sun Message Store and `/var/mail` is multithreaded. This feature enables a single process to manage a large number of connections. Each connection is handled by a thread. Multithreaded access maximizes both performance and scaleability by minimizing the system resources required for the management of each connection.

# Sun OpenWindows Mail Tool V3 Format Conversion

Before a message can be processed by the Sun Message Store, the body of the message must be in Multipurpose Internet Mail Extensions (MIME) format. (The Sun Message Store stores messages in MIME format only.) When a message enters the Sun Message Store queue, it checks the message body format. If the message body is in MIME format, the Sun Message Store automatically processes the message. If the message body is in Sun OpenWindows™ Mail Tool V3 format (the message was generated in Mail Tool), the Sun Message Store converts the message body format to MIME.

For more information on MIME format, refer to “Internet Message Transfer Agent” on page 6.

This feature is not user configurable.

# Simultaneous Connections

SIMS allows simultaneous IMAP and POP client connections. While this is standard for most IMAP clients, for POP clients this means that if a connection is dropped, or a user wants more than one connection, then a user can immediately reconnect and have access to her messages without having to wait for the POP server to time-out.

## Simultaneous IMAP Connections

When multiple IMAP clients connect to a mailbox, all flag changes (new, seen, deleted, answered, and so on) and message states (new mail, removed message) are shared between clients by means of the IMAP protocol.

## Simultaneous IMAP and POP Connections

When IMAP and POP clients connect to a mailbox, all flag changes (new, seen, deleted, answered, and so on) and message states (new mail, remove message) are shared between IMAP clients. POP clients, however, will not see IMAP deleted messages. When a POP client connects to a mailbox, it simply grabs a snapshot of the mailbox which persists until the POP session is terminated.

### Simultaneous POP Connections

When multiple POP clients connect to a mailbox, each grabs a unique snapshot of the mailbox state at connect time. Each connection's unique snapshot persists until the POP session is terminated. Any changes to a mailbox's state are seen by clients that connect after these changes are made.

# Message Access Protocols



This section provides further explanation of the message access protocols.

## Message Access Protocol Transparency

The Sun Message Store and `/var/mail` support the Internet Mail Access Protocol version 4 (IMAP4) and the Post Office Protocol version 3 (POP3). Therefore, a mail client user can access either message store using either protocol. For example, a user's Inbox and other folders can be stored on the Sun Message Store and accessed using IMAP4 or POP3 or they can be stored in `/var/mail` and accessed using IMAP4 or POP3.

## Security Between Mail Client and Mail Server

For information on security implemented between the Internet Mail Access Protocol version 4 (IMAP4) or Post Office Protocol version 3 (POP3) email client and the SIMS, refer to Chapter 9, "Secure Sockets Layer (SSL) Support in SIMS."

# Sun Directory Service



One of the SIMS major components is the *Sun Directory Service* (also referred to as the *SunDS*, the *LDAP directory service*, or simply the *directory service*). SunDS is an implementation of a directory service based on the Lightweight Directory Access Protocol (LDAP). SunDS maintains a directory of user and distribution list information, as well as configuration data for Sun Messaging Connectivity Services channels (SMCS). The SunDS provides address information used by the IMTA to deliver messages.

## Directory Information

Directory information is stored in directory entries. A *directory entry* is a set of *attributes* and their *values*. For example, SIMS specifies an attribute called `commonname`, its value would be the user's full name. Another attribute is `mailHost`. Its value would be the host name.

Directory entries are defined by an *object class* which specifies the kind of object the entry describes and the set of attributes it contains. For example, SIMS specifies an `emailPerson` object class which has attributes such as `commonname`, `mail` (email address), `mailHost`, and `mailQuota`. SIMS also specifies an `emailGroup` object class which has attributes such as `commonname`, `authorizedSubmitter`, and `rfc822MailMember`.

The set of object classes supported by a directory service is called the *directory schema*. The schema specifies all the objects and attributes supported by the directory service, as well as which attributes are mandatory and optional for a given object class. See Appendix D, "SIMS Directory Schema and Directory Information Tree," for more information.

Directory entries are organized hierarchically in a tree-like structure called the *Directory Information Tree* or *DIT*. Each entry has a parent entry and can have child entries. The top of the hierarchy is known as the *root entry*.

An entry is identified by its *distinguished name* (DN). A distinguished name is a sequence of locale-related attributes and values. The first attribute and its value provide the entry's *relative distinguished name* (RDN). The rest of the sequence is the distinguished name of the parent entry. A distinguished name is unique throughout the whole directory service.

FIGURE 1-5 shows an example of how directory information is structured, with the DNs and RDNs of the shaded entries.

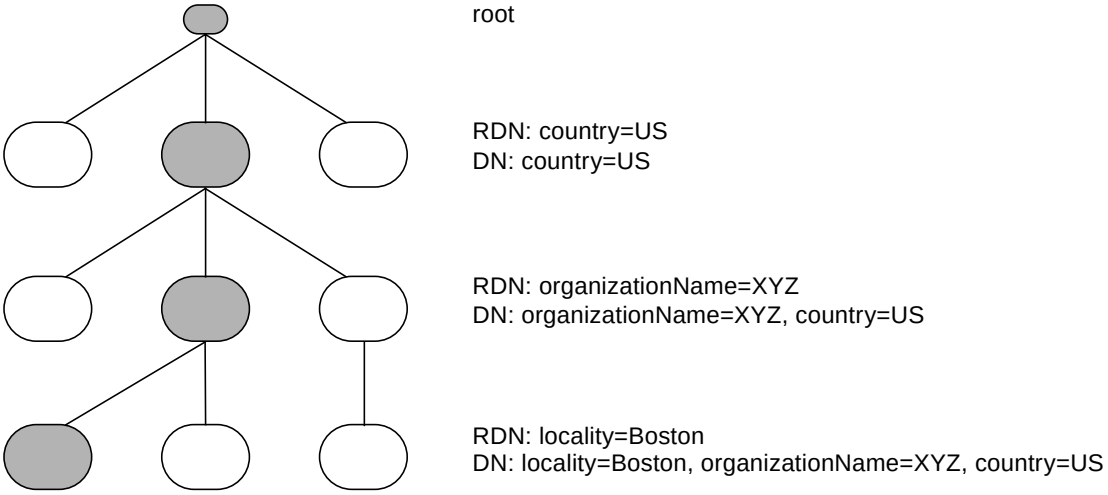



**FIGURE 1-5** Directory Information Tree

In SIMS, specific directory information is often stored in *naming context*s. A *naming context* is simply a subtree or "branch" of the DIT that is identified by its DN. An example of a naming context would be one which stores all entries for marketing employees of the XYZ Corporation at the Boston office. The name of that naming context might be `ou=mktg,ou=Boston,o=XYZ,c=US` (`ou` is organizational unit, `o` is organization, `c` is country). Naming contexts are listed under *data store* in the SIMS Admin Console.

In a general-purpose directory service, you have to decide what information you want to store and how that information will be organized. The SunDS directory service has already been designed for you, though you can modify this design, as described in "Modifying the Schema" on page 200.

## *Organizational Units*

An *organizational unit* is a layer in the directory information tree (DIT). The number of organizational units in a DIT is dictated by the size of your organization and the structure of the Domain Name Service (DNS). For more conceptual information on the DIT and organizational units, refer to the *SIMS Installation Guide*.

During installation of the SIMS software, the portion of the DIT that you plan to implement on a particular mail server, including the organizational units, if any, is automatically generated. Subsequently, you can add or delete an organizational unit. For more information, refer to “Creating Organizational Units” on page 66 and “To Delete an Organizational Unit” on page 73.

## *User Entries*

A user entry or *user profile* contains information on a user. TABLE 1-4 describes each user entry field and whether you must configure a particular field when creating a user entry. A required field is one that actually configures some aspect of the SIMS, while a field that is not required is one that provides incidental information or configures an optional feature.

**TABLE 1-4** User Entry Fields

| Field | Required/Not Required For SIMS Configuration | Description |
|---|---|---|
| **Personal Information/Name** | | |
| Distinguished name (dn) | Required | A unique path name associated with a user entry that reflects the hierarchy of the directory information tree. |
| Full name | Required | Stores the possible variations of the first name, last name, and middle initial fields combined. The middle initial is optional. Examples of full names for one particular user are Harrison Green, Harry Green, and Harry A. Green. |
| First Name | Not required | For example, in the case of Harry Green, the first name is Harry. |
| Last Name | Required | A last name is a surname, for example, in the case of Harry Green, the last name is Green. |
| Middle Initial | Not Required | The middle initial is the first letter of the middle name, for example, in the case of Harry A. Green, the initial is A. |
| Title | Not required | A business or personal title, for example, Accountant or Avid Science Fiction Fan, respectively. |
| **Personal Information/Telephone** | | |

**TABLE 1-4** User Entry Fields *(Continued)*

| Field | Required/Not Required For SIMS Configuration | Description |
|---|---|---|
| Telephone Number | Not required | Can also include extension number. |
| Fax Number | Not required | Self explanatory. |
| Pager Number | Not required | Self explanatory. |
| Mobile Phone Number | Not required | Self explanatory. |
| **Personal Information/Address** | | |
| Postal address | Not required | Self explanatory. |
| Location | Not required | Self explanatory. |
| Office Number | Not required | Self explanatory. |
| **Personal Information/Miscellaneous** | | |
| Home Page | Not required | The Uniform Resource Locator (URL) for a home page. |
| Description | Not required | Self explanatory. |
| Additional Information | Not required | Self explanatory. |
| **System Information** | | |
| Home directory | Not required | Location of user's home directory, for example, `/home/harryg`. |
| Login name | Required | Unique identification (ID) for user, for example, harryg. |
| Password | Required | Password associated with login name field; can be stored clear (unscrambled) or crypted (scrambled) |
| **Mail Information** | | |
| Mail Host | Required | Name of the user's mail server. |
| Delivery Channel Type | Required | Can be either Internet, Lotus cc:Mail, Microsoft Mail, or IBM PROFS. |
| Preferred Recipient, Lotus cc:Mail, Microsoft Mail, or IBM PROFS Mail Address | Req. if specified Delivery Channel Type is Connectivity channels | Email address that a recipient inside the email system or the local area network (LAN) proprietary system will see when a message from this user is received. Example: turquoise@mcmqa-s-7.eng.adagio.com |
| Preferred Originator, Lotus cc:Mail, Microsoft Mail, or IBM PROFS Mail Address | Required if specified Delivery Channel Type is one of Connectivity channels | Email address that a recipient outside the email system or the LAN proprietary system will see when a message from this user is received. Example: kelly.wampus@eng.adagio.com |

**TABLE 1-4** User Entry Fields *(Continued)*

| Field | Required/Not Required For SIMS Configuration | Description |
|---|---|---|
| Internet Mail, Lotus cc:Mail, Microsoft Mail, or IBM PROFS Mail Address | Required if specified Delivery Channel Type is one of Connectivity channels | All valid Internet, Lotus cc:Mail, Microsoft Mail, or IBM PROFS Mail aliases for a user. The IMTA will accept a message addressed to any one of the specified addresses. |
| Internet Mail Delivery Options | Required if you specified Internet as Delivery Channel Type | Location of user's Inbox. Can be either `/var/mail` or the Sun Message Store. If `/var/mail`, then must specify mailbox directory. Can optionally enable auto reply, program, forward, and append to file features. |
| Auto Reply Expiration Date | Req. if you enable auto reply feature in Internet Mail Delivery Options | Date that auto reply feature expires. |
| Auto Reply Mode | Req. if you enable auto reply feature in Internet Mail Delivery Options | Vacation is only mode currently supported. |
| Auto Reply Subject | Req. if you enable auto reply feature in Internet Mail Delivery Options | Subject line of auto reply message. |
| Auto Reply Text | Req. if you enable auto reply feature in Internet Mail Delivery Options | Body of auto reply message. |
| Auto Reply Text For Use Within Organization | Req. if you enable auto reply feature in Internet Mail Delivery Options | Body of auto reply message for use within the user's organization. |
| Program Delivery Info | Req. if program feature is enabled in Internet Mail Delivery Options | Specifies one or more commands with arguments to deliver to a UNIX program. |
| Forwarding Address | Req. if forward feature is enabled in Internet Mail Delivery Options | Internet address to which email should be forwarded. |
| Delivery File | Req. if append to file feature is enabled in Internet Mail Delivery Options | Pathname of file to which email should be attached to the end of. |
| **Calendar Information** | | |
| Calendar Host | Req. for HotJava/Web Access calendars | Calendar Server host name |
| Default Calendar | Req. for HotJava/Web Access calendars | Name of default Calendar. |

For complete information on the syntax required when configuring these fields, refer to either “To Create a User Entry” on page 59 or “To Modify a User Entry” on page 73.

You can add additional user entry attributes or modify existing ones. For more information, refer to “Modifying the Schema” on page 200.

## *Group Entries*

A group entry contains information for a distribution list. TABLE 1-5 describes each group entry field and whether you must configure a particular field when creating a group entry. A required field is one that actually configures some aspect of the SIMS, while a field that is not required is one that provides incidental information or configures an optional feature.

**TABLE 1-5** Group Entry Fields

| Field | Req/Not Req. For SIMS Configuration | Description |
|---|---|---|
| **General info./General** | | |
| Distinguished name (dn) | Req. | A unique pathname associated with a group entry that reflects the hierarchy of the directory information tree (DIT). |
| Full name | Req. | A *full name* is the possible variations of the group address. An example of a full name for one particular group is marketing. |
| Mail domain | Req. | The mail domain in which a group’s mail server resides, for example, sales.alpha.com. |
| Send Error Conditions To | Req. | The individual who receives a notice when an error condition related to the distribution list arises, for example, if a message addressed to the distribution list cannot be delivered. |
| Send Request Messages To | Req. | The individual who receives a notice when another individual requests being added as a distribution list member. |
| Mail Host | Req. | The hostname of the group’s mail server. |
| Password | Req. | Password associated with group and with a shared mailbox; can be stored clear (unscrambled) or crypted (scrambled). You are prompted for this password when attempting to modify group entry attributes using the command line interface or the user administration interface. |
| **General info./Telephone** | | |
| Expandable | Not req. | Make list of members for a particular group or distribution list accessible to all users. |
| Telephone Number | Not req. | Telephone number for the group. Can also include extension number. |

**TABLE 1-5** Group Entry Fields *(Continued)*

| Field | Req/Not Req. For SIMS Configuration | Description |
|---|---|---|
| Fax Number | Not req. | Fax number for the group. |
| Pager Number | Not req. | Pager number for the group. |
| Mobile Phone Number | Not req. | Mobile phone number for the group. |
| **General info./Address** | | |
| Postal address | Not req. | Postal address for the group. |
| Location | Not req. | Location for the group. |
| Building | Not req. | Building of the group. |
| Office Number | Not req. | Office number for the group. |
| Home Page | Not req. | The Uniform Resource Locator (URL) for a home page. |
| Description | Not req. | Description for the group. |
| Additional Information | Not req. | Additional information for the group. |
| Owner | | |
| Owner | Req. | An owner is an individual who is responsible for a distribution list. An owner can add or delete distribution list members. |
| **Moderator** | | |
| Moderator | Not req. | If moderator feature is enabled, a message addressed to a distribution list is initially sent to the moderator only. The moderator can take one of the following actions: forward the message to the distribution list, edit the message and then forward it, or not forward the message. |
| **Member Information** | | |
| Member | Not req. | A member is a user or group who receives a copy of an email addressed to a distribution list. |
| **Additional Delivery Options** | | |
| Shared Mailbox | Not req. | Specifies that messages are delivered to a shared mailbox in the Sun Message Store. |
| Program | Not req. | Specifies one or more commands with arguments to deliver to a UNIX program. |
| Append to File | Not req. | Path name of file to which email should be appended (attached to the end of). |
| **Access Control** | | |
| Authorized Domain | Not req. | Domain name from which users or groups are authorized to send messages to a particular distribution list. |

**TABLE 1-5** Group Entry Fields *(Continued)*

| Field | Req/Not Req. For SIMS Configuration | Description |
|---|---|---|
| Unauthorized Domain | Not req. | Domain name from which users or groups are not authorized to send messages to a particular distribution list. |
| Authorized Submitter | Not req. | Name of user or group who are authorized to send messages to a particular distribution list. If the user or group is internal to the email system, specify the distinguished name; if external to the email system, specify an email address in RFC 822 format. |
| Unauthorized Submitter | Not req. | Name of user or group who are not authorized to send messages to a particular distribution list. If the user or group is internal to the email system, specify the distinguished name; if external to the email system, specify an email address in RFC 822 format. |

For complete information on the syntax required when configuring these fields, refer to either “To Create a Group Entry” on page 62 or “To Modify a Group Entry” on page 82.

You can add additional group entry attributes or modify existing ones. For more information, refer to “Modifying the Schema” on page 200.

## Aliasing

You can define an *alias entry*. An alias entry is identified by a distinguished name (DN). It contains the name of the directory entry it represents (the aliased object name). The alias entry and the entry it represents must be in the same data store.

For bind and search operations, you can specify that the directory should translate an alias DN to the DN of the actual entry. This is known as *dereferencing* the alias. For other operations, you need to treat the alias entry as an ordinary entry and not dereference it, for example, to modify the RDN of the alias entry itself, not of the aliased object.

### *Alias Entries and Searching*

The result of a search or read operation involving an alias entry differs depending on whether you dereference the alias. There are four possible settings for the alias dereference flag:

- Never dereference alias

  All operations apply to the entry with the given DN, even though the entry is an alias entry. This is the default setting.

- Dereference alias when finding base object

  The base object identifies the top of the subtree of entries to be searched. This setting means that if you specify an alias as the base object it will be dereferenced, but no other aliases encountered during the search are dereferenced.

- Dereference alias when searching

  If the operation being carried out is a search, all alias entries specified or used in the search are dereferenced, except the base object. If the result of the search is an alias entry, the aliased object is returned to the user, not the alias entry. This can sometime lead to unexpected results from searches based on DN content, where the requested information is not present in the entries returned because the entry that contains the requested DN term is an alias entry that has been dereferenced.

- Always dereference alias

  All alias entries specified or used in the operation are dereferenced.

For example, suppose your directory contains the following pair of entries:

```
1) cn=Stan Smith, role=Personnel Administrator, ou=Personnel,
ou=Corporate, o=XYZ, c=US
```

with attributes:

```
objectclass=orgPerson
cn=Stan Smith
telephoneNumber=123 456 7890
mail=dtmail
```

```
2) cn=personnel, o=XYZ, c=US
```

with attributes:

```
objectclass=alias
aliasedObjectName=”cn=Stan Smith, role=Personnel Administrator,
ou=Personnel, ou=Corporate, o=XYZ, c=US”
```

With alias dereferencing when searching, if you search for the telephone number of cn=personnel, o=XYZ, c=US, you will see Stan Smith’s telephone number. With no alias dereferencing, you would not see any telephone number.

Defining aliases for roles is particularly useful when the person occupying a role changes frequently (the duty network manager for out-of-hours calls, for example), so that users always query the same entry. You can change the value of the aliasedObjectName with a script that runs on a schedule and calls `ldapmodify` to make the changes.

See the *SIMS Reference Manual* for details of how to specify how alias dereferencing is used in `ldapsearch`.

### *Alias Entries and Authentication*

Every interaction with the directory starts with a bind request, to authenticate the user and establish the level of access permitted. The DN supplied in a bind request can be the DN of an alias entry. With alias dereferencing, the user binds with the DN contained in the aliasedObjectName of the alias entry, and is granted the access rights defined for the entry with that DN.

Alias dereferencing in bind is a configuration choice. If aliases are not being dereferenced and the user binds with the DN of an alias entry, access is denied because the password attribute is not present.

## Infrastructure Information

Infrastructure information determines how the components of a directory service behave and how directory entry information is interpreted. It includes the directory schema, knowledge information, and component configuration information.

### *Schema*

The directory schema is the set of rules that describes the data that can be stored in the directory. It defines the types of entries permitted, their attribute structure, and the syntax of the attributes. This product contains a pre-defined schema, which you can modify, with certain restrictions. See “Modifying the Schema” on page 200 and Appendix D, “SIMS Directory Schema and Directory Information Tree for further schema details.

### *Knowledge Information and Referrals*

A directory server uses knowledge information to pass requests for information to other servers. The knowledge information held by a directory server is a reference to a directory server holding other naming contexts. When a server receives a request for information, it checks whether it can respond to the request using the information in the local data store. If it cannot, it checks the referral defined for the data store, and returns the details of an alternate directory server to the directory client. The client can then send the request to the other directory server. Some clients contact the alternate server automatically, so the referral mechanism is transparent to users. Other clients return the referral information to the user.

See “Example: The XYZ Corporation” on page 40 for an example of how referrals are used.

# Password Management

A directory entry for a SIMS user contains a userPassword attribute. The value of this attribute, which is used to authenticate the user to the directory, can be stored in encrypted or unencrypted format. See “General Properties Configuration” on page 188 for details of how to specify whether or not passwords are stored in an encrypted format. By default, passwords are encrypted.

When you supply a password for authentication, or as an attribute value in a directory operation, you specify the value in unencrypted format. You do not have to enter the password in its encrypted format.

The directory service software uses the `crypt(1)` utility to encrypt passwords.

# Access Control

Access to information in the directory is controlled by a set of rules that determine what permissions a user requires in order to access an entry or an attribute. The permission level granted to the user depends on the authentication information provided by the user. It also depends on the specific rules defined by the directory administrator for a particular entry or attribute.

## Permission Levels

There are five levels of permission for directory information:

- none
  You are not permitted to access the entry at all, and will not even see information indicating that the entry exists.
- compare
  You can compare the value of a given attribute with a value you supply, but you cannot read the attribute value. This is used when checking passwords.
- search
  You can read the distinguished name of an entry, and you can search for entries based on the existence of an attribute or attribute value. You will not necessarily be able to read the attribute value.
- read
  You can read the value of any accessible attribute within an entry.
- write
  You can write information into an entry or attribute, that is, you can modify or delete an attribute value, attribute, or whole entry.

**Note –** When you are granted permission for a given level of operation, you are implicitly granted all lower levels of permission. For example, read permission implies that search and compare permissions are granted too.

## Defining Rules for Entries and Attributes

Access control rules define which users are granted which permission for a given set of entries or attributes. For example, you can give a privileged user read permission for all attributes except password in all entries, and compare permission for password attributes.

You can define an access control rule for any set of entries that can be defined by:

- All entries
- A distinguished name based regular expression (see “Using the Distinguished Name Editor” on page 211)
- An LDAP filter (see “Specifying an LDAP Filter” on page 212)
- The presence of a particular attribute

You can define access control rules that apply to the person described by an entry (using the keyword *self*), so that, for example, only you can change your own password. You can also define access control rules that apply to any user (using the keyword *everyone* or *).

The access control rules are applied in sequence, so the order in which they are listed is important. You must state the most specific rules first, with more general rules afterward.“Configuring Access Control” on page 205 explains how to define an access control rule using the configuration tool, and how to specify the order of rules.

For example, you could define the following access control rules:

- Users have write access to their own password attribute, but only compare access to the passwords of other users.
- A user whose entry contains the attribute value locality=San Francisco has read access to all other entries that contain the attribute value locality=San Francisco, but cannot read the password attribute value.

The default access controls defined at installation are as follows:

- All users have compare access to the values of the attribute userPassword. To change the value of the userPassword attribute, you must bind with the DN of the entry containing the attribute, that is, the password can only be changed by owner of the entry.

- Everyone has read access to the following attributes: cn, dataSource, homeDirectory, messageStore, messageStoreSizeQuota, mail, mailServer, objectStatus, preferredRfc822Recipient, rfc822Mailbox, uid.
- Any user can add their DN to, or delete their DN from, the member attribute of any entry containing the attribute joinable with value TRUE.
- Anyone binding with the DN of an entry has write access to that entry. Everyone else has read access only.
- The administrator always has complete access to all attributes in all entries. You cannot change the access granted to the administrator, which ensures that there is always at least one user who has access to every entry in the directory.

These rules are applied in order, starting with the most specific followed by the more general rules.

CODE EXAMPLE 1-1 shows how the default access controls are defined in the directory server configuration file.

**CODE EXAMPLE 1-1** Default Access Controls

```
        access to attrs=userPassword
        by self write
        by * compare
access to attrs=cn, dataSource, homeDirectory, MailMessageStore, mailQuota, mail,
mailHost, objectStatus, preferredRfc822Recipient, rfc822Mailbox, uid
        by self read
        by * read
access to filter="joinable=TRUE" attrs= member, entry
        by dnattr=member self write
access to *
        by self write
        by * read
```

All directory interactions begin with a bind. The information used to establish the bind is also used to determine the permission level at which you are granted access to the directory. All further interaction with the directory for the duration of the bind is regulated by this permission level.

# Directory Structure

Information in the directory is organized in a tree structure, called the *Directory Information Tree (DIT)*. The structure of a DIT usually reflects very closely the structure of the information it contains. For example, a directory containing entries for people in a corporation could be organized by division or by location. In general, DIT structures are organizational, geographical, or include both organizational and geographical factors.

Directory information is stored in a *data store*. A data store is the physical location where a naming context is held. A data store can hold more than one naming context. A directory server can contain more than one data store.

When dividing the DIT into data stores to be held on individual servers, you need to take account of the following:

- A search cannot cross data stores. If you want to perform a search in several data stores, you have to start several simultaneous search operations.
- Alias definitions are local to a data store

## Example: The XYZ Corporation

The XYZ Corporation is a pharmaceutical company, with headquarters in Boston, Massachusetts. They have two manufacturing operations, one in San Francisco and one in Paris, and three distribution centers, in Atlanta, London, and Tokyo. There are two research groups in London and San Francisco, located with the other XYZ divisions in those cities. The Sales organization has three divisions: Europe, US&C (US and Canada), and RoW (the rest of the world).

FIGURE 1-6 shows the functional structure of XYZ Corporation.

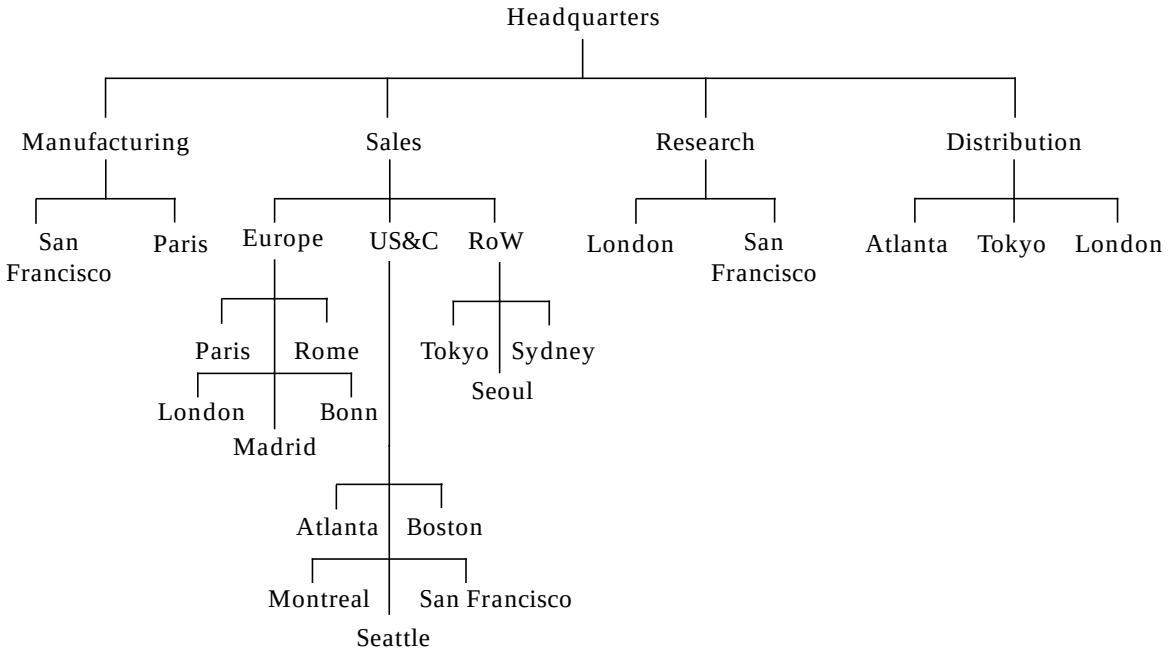



**FIGURE 1-6** Functional Structure of XYZ Corporation

FIGURE 1-7 shows the geographical structure of XYZ Corporation.

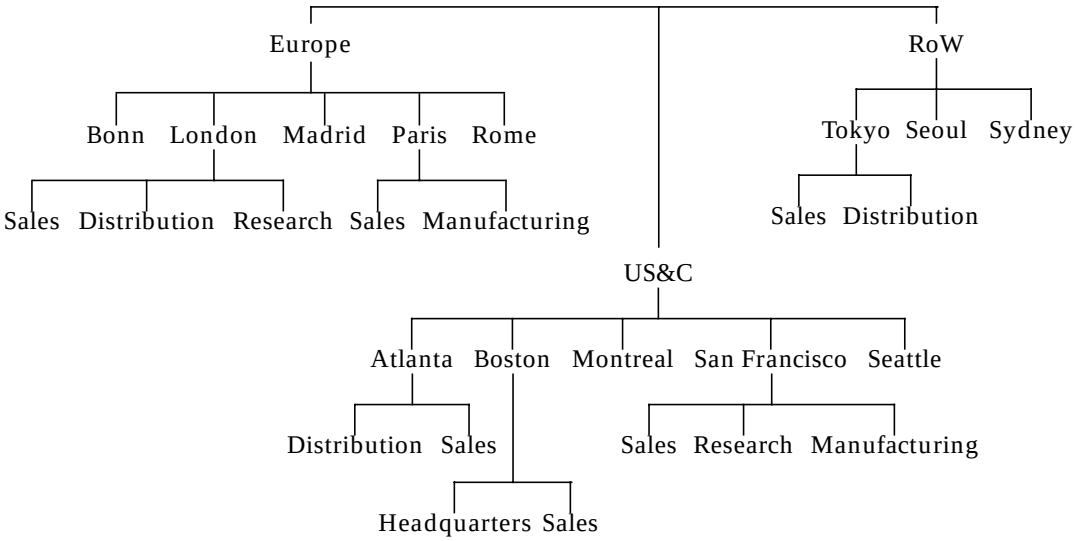



**FIGURE 1-7** Geographical Structure of XYZ Corporation

As is common with many organizations, neither an organizational DIT structure nor a functional DIT structure completely meets the directory structure needs of XYZ Corporation, so the network management team decides to combine functional and geographical factors, and to take into account the different usage patterns within the different departments. The result is the DIT structure shown in FIGURE 1-8.

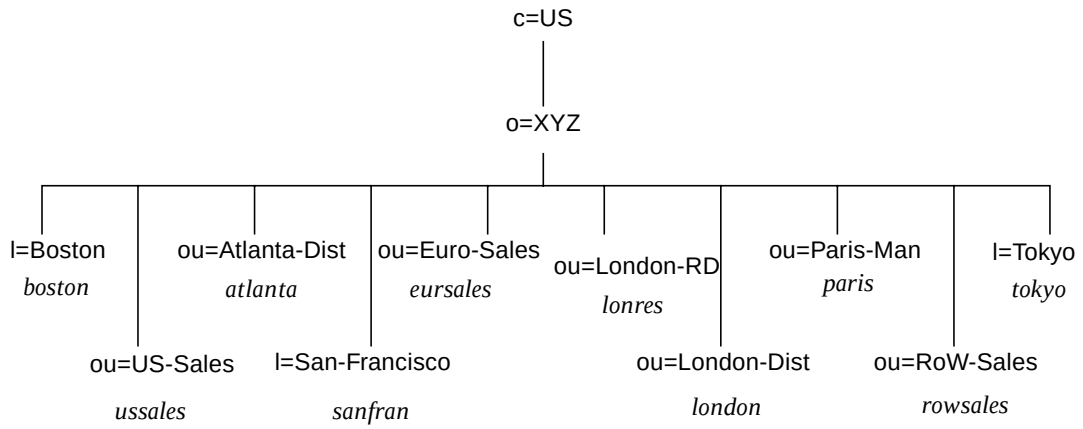



**FIGURE 1-8** DIT Structure for XYZ Corporation

In this DIT structure, the corporation is divided into seven organizational units (ou) and three localities (l) corresponding to ten naming contexts. FIGURE 1-8 shows the RDN of each naming context and the server on which it is stored. Each naming context contains entries that are related to a particular geographical or functional area. Given that much of the enquiry traffic is expected to be local to a server, this reduces the network traffic.

A referral system ensures that if an entry cannot be found locally, the directory server can pass the request to another directory server. FIGURE 1-9 shows the content of each naming context, and the referrals defined between servers.

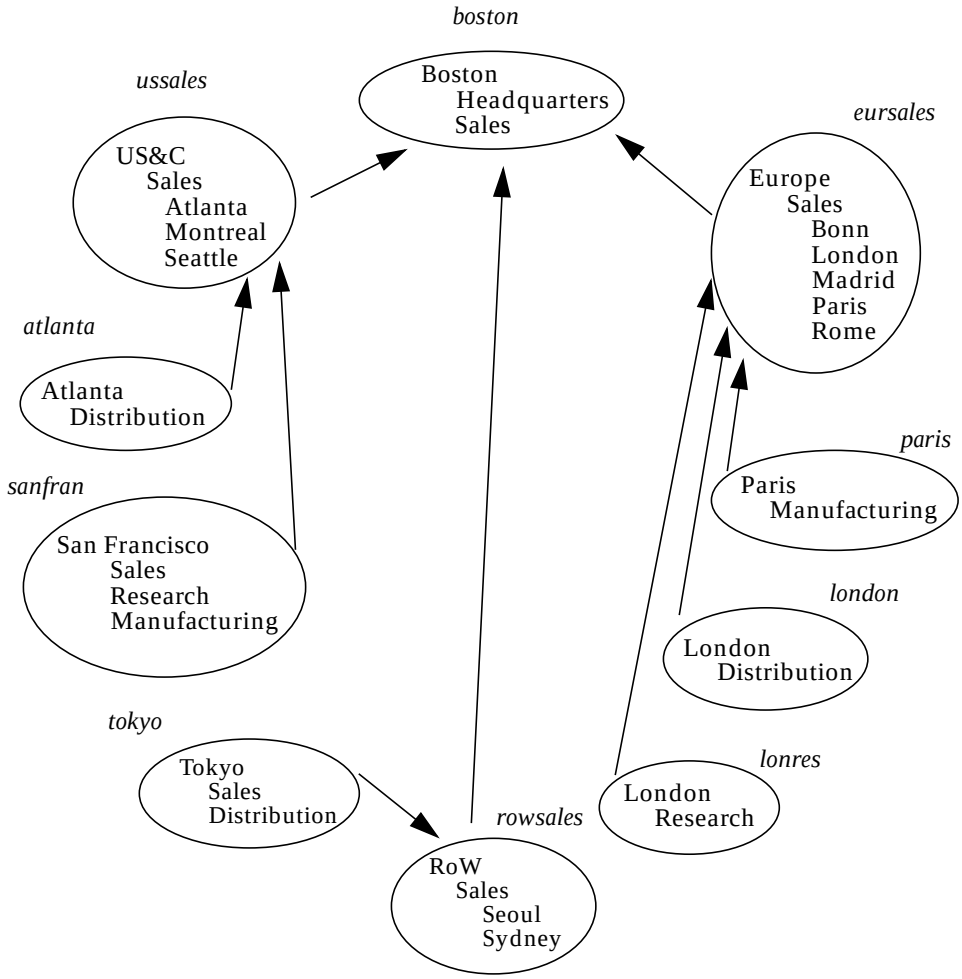



**FIGURE 1-9** XYZ Corporation Referrals

# Replication

You can share between several directory servers the load of processing requests generated by directory service clients for the same information. This is done by defining a replica, or slave server to provide an alternative access point to the directory service for clients. A master naming context can have more than one replica naming context. FIGURE 1-10 shows a master server with two replica servers. *Replication* is the process by which changes in the master data store are propagated

to all the replica naming contexts. You can replicate an entire naming context, a subtree, or a particular entry. You can replicate the entire content of an entry or you can specify a subset of attributes to be replicated.

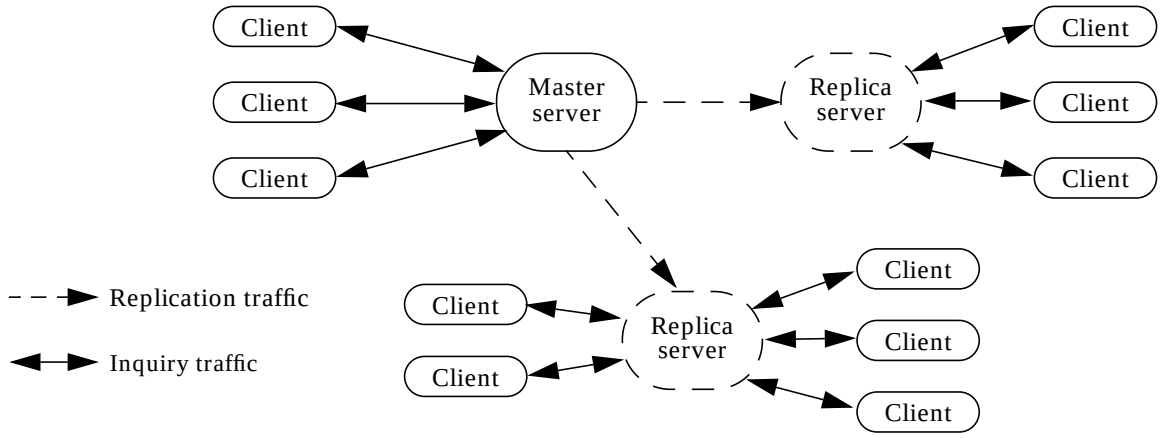



**FIGURE 1-10** Master and Replica Servers

Using replication has the following advantages:

- It reduces the load on the master server by diverting some traffic to other servers.
- You can store copies of data where it is mostly frequently used, thus reducing network traffic.
- You can store many copies of the data, but the data is maintained from a central location.
- You need only replicate the data that is required by clients of the replica server, if you know the requirements of those clients specifically enough. You may be able to tailor a replica exactly to the needs of a specific client. By reducing the number of entries replicated, you reduce the network traffic caused by replication updates.
- You could maintain a “public” replica server containing information that is not confidential, allowing greater access to this information than you usually allow for other servers. For example, you could create a server containing the email addresses for the sales and support staff who deal with current products but not the research staff working on future products, and make it available to the sales staff of a partner company.

**Note –** You could provide the same partial view of directory information with appropriate access controls. However, using a partial replica on a dedicated machine ensures that you are not providing access to your entire network. For extra security, you could connect the replica server to your network only while the replication update is in progress.

The costs of using replication are:

- Additional network traffic caused by replication of data. However, though there may be an overall increase in traffic, more of the traffic will be local, so you can avoid known network bottlenecks for inquiry traffic. Also, you can time the replication updates for when the network is least busy.
- Information retrieved from replicas may be out of date if replication has not happened since an update, so certain applications may always need to query the master data store.
- You cannot modify a replica. All updates must be performed on the master copy of an entry.

## How Replication Works

Information from a master naming context is propagated to a replica by the `slurpd` daemon. The `slurpd` daemon can run permanently, so that updates to directory information are propagated immediately, or you can define a synchronization schedule. You can override the schedule at any time and trigger an immediate synchronization. This is useful if you change a large number of entries and do not want to wait for the next scheduled synchronization.

The `slurpd` daemon uses the LDAP protocol to update a replica naming context. A master naming context for which a replica is defined maintains a replication log. Each time the master naming context is updated, the transaction is recorded in the replication log. When the `slurpd` daemon next runs, it reads the replication log and sends the change to the `slapd` server that holds the replica naming context. The `slapd` server handles update requests from `slurpd` in the same way that it handles all requests, using the information supplied in the bind request to set the access level granted to `slurpd` requests. To guarantee that all replication updates are completed, `slurpd` must bind with the DN defined when the replica naming context was configured. If a different DN is used, write access for all entries may not be granted.

A replica data store always has a referral pointing to the master data store. If a replica server receives a request to modify an entry, it returns a referral to the client, indicating the master server to be contacted. In some cases, the client software handles the referral automatically and the user need not resubmit the query. Once the modification has been made in the master naming context, the change is sent to the replica naming context the next time the `slurpd` daemon runs.

## Example: Replication in the XYZ Corporation

FIGURE 1-8 and FIGURE 1-9 show the naming contexts in the XYZ Corporation DIT and the servers that store them. In addition, the network management team decide to establish several replica naming contexts:

- All servers will contain a replica of l=Boston, o=XYZ, c=US for fast access to entries concerning the headquarters of the corporation. Only ussales, eursales, and rowsales get their replicas directly from the boston server. The other servers get a replica of the replica from the server that is closest in the network.
- A second server, eursale2, will hold a complete replica of ou=Euro-Sales, o=XYZ, c=US to share the load on the existing eursales server.
- Each of the servers at the distribution centers will hold complete or partial replicas of the other distribution center naming contexts. For example, the atlanta server will hold a complete replica of ou=London-Dist, o=XYZ, c=US and a partial replica of l=Tokyo, o=XYZ, c=US containing the information about the distribution center but not about the sales office.

TABLE 1-6 shows the replication strategy for each server in the XYZ Corporation directory service.

**TABLE 1-6** Replication Strategy for the XYZ Corporation

| Server | Naming Contexts | Replication Status |
|---|---|---|
| boston | l=Boston, o=XYZ, c=US | master, replicated to ussales, eursales, and rowsales |
| ussales | ou=US-Sales, o=XYZ, c=US | master |
| | l=Boston, o=XYZ, c=US | replica from boston, replicated to atlanta and sanfran |
| eursales | ou=Euro-Sales, o=XYZ, c=US | master, replicated to eursale2 |
| | l=Boston, o=XYZ, c=US | replica from boston, replicated to eursale2, london, and paris |
| eursale2 | ou=Euro-Sales, o=XYZ, c=US | replica from eursales |
| | l=Boston, o=XYZ, c=US | replica from eursales |
| rowsales | ou=RoW-Sales, o=XYZ, c=US | master |
| | l=Boston, o=XYZ, c=US | replica from boston |
| atlanta | ou=Atlanta-Dist, o=XYZ, c=US | master, replicated to london and tokyo |
| | l=Boston, o=XYZ, c=US | replica from ussales |
| | ou=London-Dist, o=XYZ, c=US | replica from london |
| | ou=dist, l=Tokyo, o=XYZ, c=US | partial replica from tokyo |

| Server | Naming Contexts | Replication Status |
|---|---|---|
| sanfran | l=San-Francisco, o=XYZ, c=US | master |
| | l=Boston, o=XYZ, c=US | replica from ussales |
| london | ou=London-Dist, o=XYZ, c=US | master |
| | l=Boston, o=XYZ, c=US | replica from eursales |
| | ou=Atlanta-Dist, o=XYZ, c=US | replica from atlanta |
| | ou=dist, l=Tokyo, o=XYZ, c=US | partial replica from tokyo |
| lonres | ou=London-RD, o=XYZ, c=US | master |
| | l=Boston, o=XYZ, c=US | replica from london |
| paris | ou=Paris-Man, o=XYZ, c=US | master |
| | l=Boston, o=XYZ, c=US | replica from eursales |
| tokyo | l=Tokyo, o=XYZ, c=US | master, partially replicated to atlanta and london |
| | l=Boston, o=XYZ, c=US | replica from rowsales |
| | ou=Atlanta-Dist, o=XYZ, c=US | replica from Atlanta |
| | ou=London-Dist, o=XYZ, c=US | replica from london |

For details of how to implement this replication strategy for the XYZ Corporation, see “Configuring Replication for XYZ Corporation” on page 200.

# Connectivity Services

For conceptual information on the Connectivity services, refer to the *Sun Messaging Connectivity Channel Guides* which describe how to connect SIMS with cc:Mail, Microsoft Mail or PROFS mail system.

CHAPTER 2

# The Administration Console Road Map

After installing, you may wish to add users, configure and monitor SIMS components, or perform other tasks. The *Admin Console* provides a GUI interface to these common tasks. To start the Admin Console, use the HotJava browser provided with SIMS and go to `http://<machine-name>/sims/`. Enter the login (default: `admin`) and password (default: `secret`). Note that only one administrator can be logged on to the Admin Console at a time.

**Tasks:**

1. Logout, Stop SIMS
2. Create/Modify User, Group or Org
3. View Event Log
4. Configure mail transport agent/ Monitor Queued Messages
5. Configure/Monitor Message Storage attributes
6. View all client connections to SIMS
7. Configure/Monitor Sun Directory (LDAP) Services
8. Help
9. Home
10. View SIMS Component Status

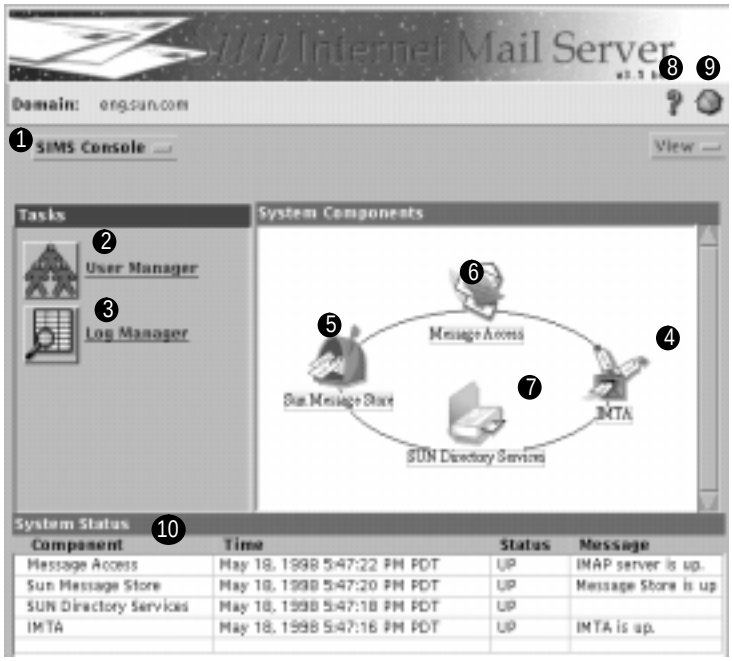



**FIGURE 2-1** Admin Console Home Page

**Note –** Also, the following line must be in the file `properties` in the `.hotjava` directory: `hotjava.default.security = low` in order for the Admin Console to run correctly.

# SIMS Components and Tasks

**TABLE 2-1** SIMS Components and Tasks (see FIGURE 2-1 on page 49)

| Components/Task | Description | Page |
|---|---|---|
| 1) Stop SIMS, Logging Out and Version Information | Self Explanatory | 54 |
| 2) User Management | Provides instructions for adding, deleting, or modifying user, group or organizational units entries in the directory. | 57 |
| 3) Event Log Manager | The Log Manager enables you to trace event messages that may help you determine why the component is in an alert or down state. | 249 |
| 4) Internet Message Transport Agent (IMTA) Administration | The IMTA is the SIMS component that receives, routes, and sends incoming messages to their destination. SIMS message transport can be customized by modifying the IMTA attributes. This chapter provides instructions for viewing and changing the message transport characteristics of SIMS, including configuring/monitoring channels, directory cache update, anti-spam features, IMTA location relative to firewall, and routability scope. | 95 |
| 5) Message Store Administration | The Sun Message Store contains the messages and attachments of the SIMS email system. You can monitor and customize the SIMS message storage and access characteristics by modifying the parameters of this component. This chapter provides instructions for monitoring message store paths, space usage and user space quotas; configuring user quotas, mail server client type, directory context, maximum connections, disk space threshold, /var/mail support, message store size increase, message purge schedule. | 141 |
| 6) Message Access Protocol Connections | View and monitor all user connections to SIMS, as well as start and stop client access to the message store. | 161 |

**TABLE 2-1** SIMS Components and Tasks (see FIGURE 2-1 on page 49)

| Components/Task | Description | Page |
|---|---|---|
| 7) Sun Directory Services Administration | The Sun Directory Service (SDS) or LDAP Server contains the addressing information for SIMS. When an email is received, the IMTA looks up the address in the directory to determine its forwarding destination. This chapter provides instructions for viewing and modifying the SDS, including modifying general properties, data store configuration, logging, directory replication, modifying the schema, configuring access control, populating the directory, and web access to the directory. | 163 |
| 8) Help | On-line help for SIMS including this book. | |
| 9) Home | Return to Home Page. | |
| 10) SIMS Component Status | Display the current state of each component. | 52 |
| **Other Important Tasks and Chapters:** | | |
| Maintenance | Describes various maintenance tasks including SIMS licensing, messages store backup and restore, folder checks, importing /var/mail users, and Sun Directory Service maintenance. | 225 |
| Troubleshooting | Various troubleshooting procedures. | 247 |
| Secure Sockets Layer (SSL) Support in SIMS | Adding SSL support in SIMS. | 269 |
| User Administration | Changing mail passwords, start/stop vacation notice, mail forwarding, alternative delivery programs. | 279 |
| Configuring SIMS as a Proxy Message Access Server | Describes how to use the SIMS message access proxy capabilities. | 285 |
| SIMS Directory Schema and Directory Information Tree | Describes the SIMS directory information tree (DIT) requirements and the type and format of objects and attributes for SIMS directory entries. | 329 |
| Error Messages | Lists SIMS error messages and appropriate responses. | 357 |

**Tip –** When you set a configuration, you must press the Apply button to save your settings. If you do not do this, you may lose your settings. You are not always warned about losing the settings when you exit a page.

# Admin Console Buttons

TABLE 2-2 describe the most common buttons that appear in the Admin Console or dialog boxes.

**TABLE 2-2** Buttons and Associated Actions

| Button | Explanation |
|---|---|
| Apply | Submits the changes made to the configuration file associated with the current page. If the changes made require a restart of the associated component, a dialog box prompts you to restart the component. The current page remains displayed. |
| Cancel | Discards the changes made to the configuration file associated with the current page. The previous page displays. |
| Help icon or button | Displays documentation specific to the Admin Console page or dialog that you are attempting to configure. |
| OK | Submits the changes made to the configuration file associated with the current page. If the changes made require a restart of the associated component, a dialog box prompts you to restart the component. The previous page displays. |
| Reset | Discards the changes made to the configuration file associated with the current page and sets the values of the changed attributes to their last saved value. The current page remains displayed. |

**Tip –** When you set a configuration, you must press the Apply button to save your settings. If you do not do this, you may lose your settings. You are not always warned about losing the settings when you exit a page.

# SIMS Component Status

SIMS polls each component periodically to determine its current state. The System Components on the Administration Console home page (FIGURE 2-2) displays the current state of each component. TABLE 2-3 outlines the possible SIMS component states.

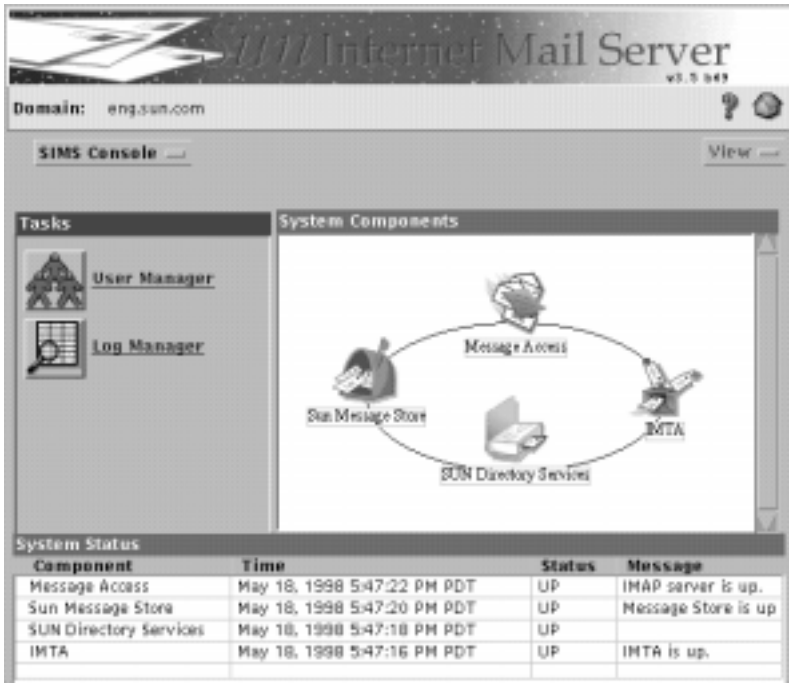



**FIGURE 2-2** Monitoring Current Component State

**TABLE 2-3** SIMS Component States

| Component State | Icon Representation | Explanation |
|---|---|---|
| Up | Component icon. | Component is functioning normally. |
| Alert | Component icon with exclamation point overlaid | Component has a non-fatal problem and is still functioning. For more information, refer to the System Status section of the Admin Console home page and “Troubleshooting” on page 247. |
| Down | Component icon with cross-bars overlaid | Component has a fatal problem and is not functioning. For more information, refer to the System Status section of the Admin Console home page and “Troubleshooting” on page 247. |

If the System Components indicates that a component is in either an alert or down state, you can access more information from the following sources:

- The System Status section on the Admin Console home page. Each entry indicates the time at which the component was polled, the component status, and more detailed information about if the component is in either an alert or down state.

  If the Internet Message Transfer Agent (IMTA) is in an alert or down state, you should also check the status of each IMTA channel. For more information, refer to "Monitoring Channel Status" on page 101.
- By using the Log Manager. For more information, refer to "Event Log Manager" on page 249.

---

**Note –** The *Stop all* function on the home page does not stop the SIMS LDAP Directory Service, `slapd`, so the SUN Directory Services icon will not have red X over it if you execute this procedure (see "To Stop SIMS Components" on page 54).

---

# Stop SIMS, Logging Out and Version Information

This section describes tasks that affect the Admin Console. They are:

- "To Stop SIMS Components" on page 54
- "To Log Out of the Administration Console" on page 55
- "To Access SIMS Version Information" on page 55

## ▼ To Stop SIMS Components

To stop the SIMS components (IMTA, Sun Message Store, message access protocolsand the Connectivity Services, if installed), go to the Admin Console Home Page, click on the SIMS Console menu and select *Stop all*.

## ▼ To Start SIMS Components

To Start all the SIMS components (IMTA, Sun Message Store, message access protocolsand the Connectivity Services, if installed) together, if they are not already started, go to the Admin Console Home Page, click on the SIMS Console pulldown and select *Start all*. This is only available if a component is stopped.

## ▼ To Log Out of the Administration Console

After you are finished with the Admin Console you may log out by going to Admin Console Home Page, clicking on the SIMS Console menu, and select *Logout*.

---

**Note –** For security reasons, we recommend logging out of the SIMS Admin Console after a task is complete. Also, since only one administrator at a time can be logged on, remaining logged on locks other administrators out of the system.

---

## ▼ To Access SIMS Version Information

Access the SIMS Version of all SIMS components by going to the Admin Console Home Page, clicking on the SIMS Console menu and selecting About SIMS.

CHAPTER 3

# User Management

This chapter describes how to add, delete, or modify user, group or organizational unit *entries* in the Sun Directory Service. To start, click on the User Manager icon on the Admin Console home page. Note that some user management functions can also be done via the command line interface (see the SIMS Reference Manual), or in a web browser (see "Web Access to the Directory" on page 215).

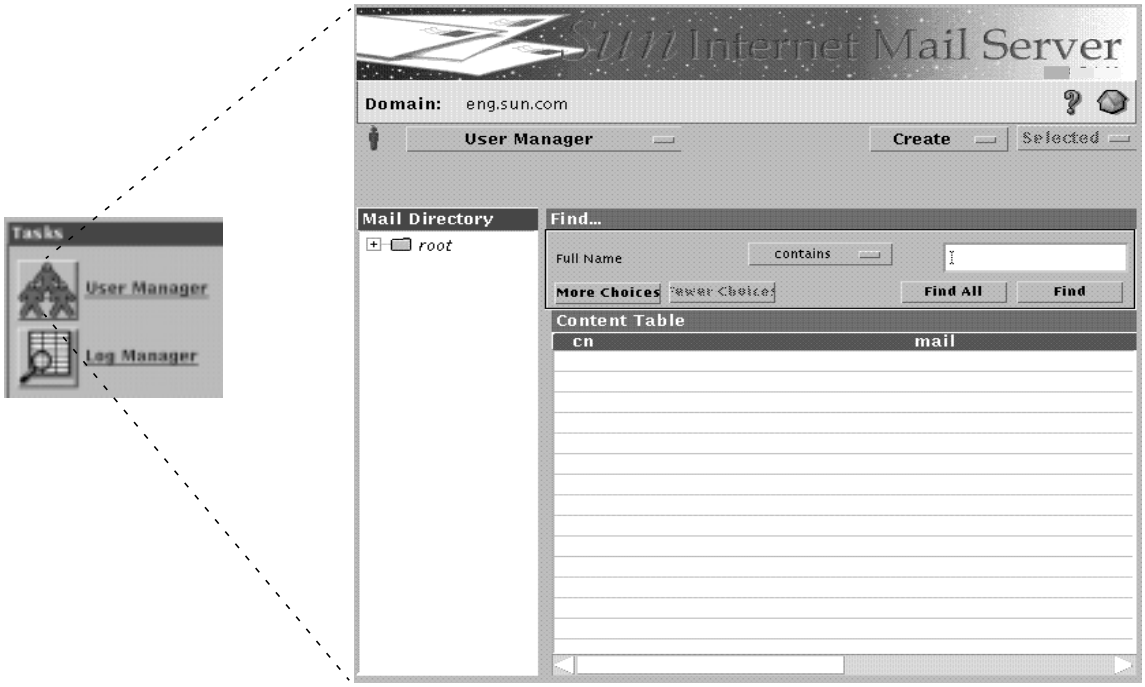



**FIGURE 3-1** User Manager Page

# User Management Topics and Tasks

**TABLE 3-1** Message Transport Topics and Tasks

| Topic/Task | Description | Page |
|---|---|---|
| Populating the Directory | Describes how to populate the directory with entries for mail users, user aliases, and distribution list from other naming or directory services (NIS, NIS+, or /etc/passwd and /etc/mail/aliases)<br>See also Appendix C, “Populating the Directory Examples” on **page 319** | 167 |
| To Create a User Entry | How to add a new mail/calendar user to the directory. | 59 |
| To Create a Group Entry | How to add a group entry to the directory. | 62 |
| Creating Organizational Units | How to add an organizational unit to the directory. | 66 |
| To Find and View an Existing User/Group Entry | Searching for an entry and displaying its property sheet. | 69 |
| To Delete a User or Group Entry from the Directory | How to remove a user or group from the directory | 72 |
| To Delete an Organizational Unit | Deleting an organizational unit from the directory. | 73 |
| To Modify a User Entry | Changing User entries | 73 |
| To Modify a Group Entry | Changing a group entry. | 82 |
| Command Line User Management | - Adding a root entry<br>- Adding Entries<br>- Modifying Entries<br>- Deleting Entries | 93 |
| Web Access to the Directory | How to search for and read entries, and to modify some directory information from a web browser. | 215 |
| User Management Error Messages | Error messages that occur when doing User Management. | 357 |

# Admin Console User Management

This section describes user management using the SIMS Admin Console.

## ▼ To Create a User Entry

Add new users by creating a user entry. There are three kinds of users: an e-mail and e-calendar user, an e-mail-only user, or a calendar-only user. Use the Admin Console for adding a small number of user entries after initially populating the directory. To initially populate the directory or to add a large number of user entries at one time, refer to “Populating the Directory” on page 167.

To see a newly created user in the Mail Directory, you need to select the Display All or Find button. If you create a new Group or Organizational Unit, you need to reload the page to see the new Group/Organizational Unit.

---

**Note –** User names must be in lower**-case** letters. Upper-case letters are converted to lower-case. Also, Newly created users will not receive mail until after an incremental or full `dirsync` occurs. See “Alias Synchronization Schedule” on page 104.

---

AdminConsole>User Manager>Create pulldown>User

1. **In the Admin Console home page, click the User Manager icon.**
2. **Choose User from the Create menu.**

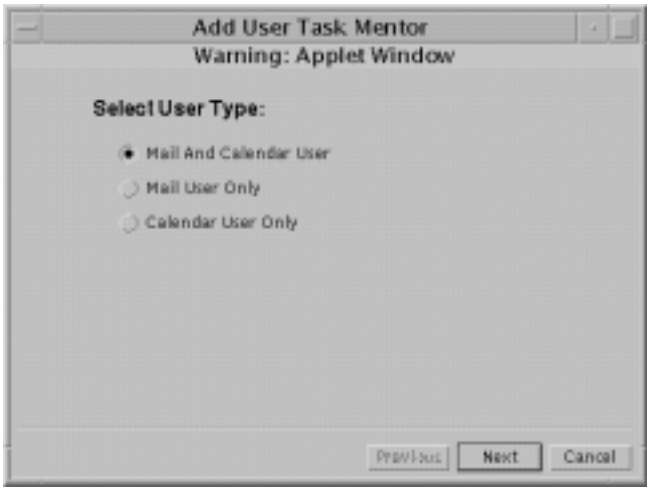



**FIGURE 3-2** Add User Task Mentor Dialog

3. **Enter the user's login name and press the Next button:**


**FIGURE 3-3** Add User Task Mentor Dialog

4. **Enter the first name, last name, middle initial, and password—password is case-sensitive—for the user and click Next.**

   If you have chosen to create a mail and calendar user, the following Add User Task Mentor dialog displays:

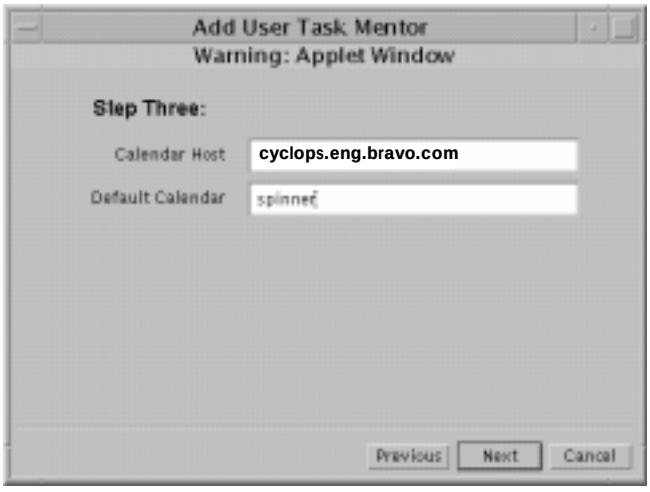



**FIGURE 3-4** Add User Task Mentor Dialog for Calendar Options

If you have chosen to create a mail user only, the following dialog displays:


**FIGURE 3-5** Add User Task Mentor Dialog for Mail Options

5. **Enter the calendar host, calendar, mail host and mail domain for the user as necessary.**

   If possible, the mail server detects a default mail host and mail domain and provides information in the appropriate fields. Verify that the default information is correct. Click the Next button. The Add User Task Mentor dialog appears.

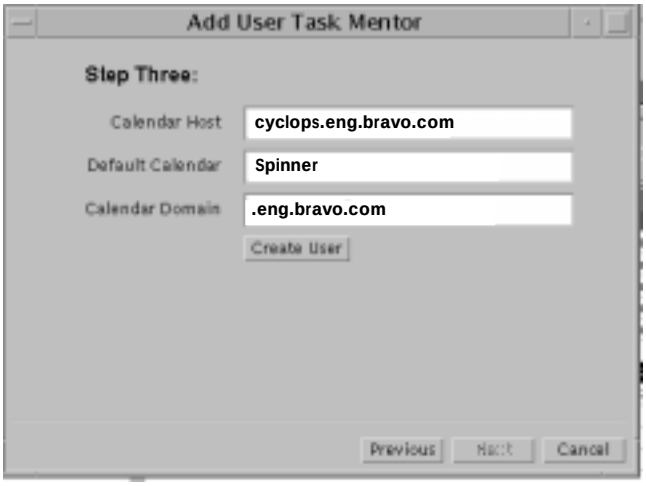



**FIGURE 3-6** Add User Task Mentor Dialog for Addresses

6. **If necessary, enter the preferred originator and preferred recipient addresses for the user.**

   A preferred originator address is a mail address that a recipient outside the email system will see when a message from this user is received. A preferred recipient address is a mail address that a recipient inside the email system will see when a message from this user is received. If possible, the mail server detects a default preferred originator address and preferred recipient address and provides information in the appropriate fields. Review the default information for correctness. Both fields must be fully qualified and in RFC 822 format.

7. **Click the Create User button.**

   The following dialog appears.


**FIGURE 3-7** Add User Task Mentor Dialog for Addresses

8. **If finished, click the Done button.**

   If you need to create another user entry, reopen the Add User Task Mentor dialog.

## ▼ To Create a Group Entry

A group entry is also known as a *distribution list*. When a message is sent to the group address, SIMS sends the message to all members in the group. You can also create a *shared mailbox* to which all the messages sent to the group are saved. To do this, first create a group entry by following the directions in this section. Then modify the group entry by following the instructions in To Modify a Group Entry," specifically page 90, for creating the shared mailbox.

**Note –** The Admin Console is practical for adding small numbers of group entries after initially populating the directory. To initially populate the directory or to add large numbers of group entries at one time, see “Populating the Directory” on page 167.

AdminConsole>User Manager>Create pulldown>Group

1. **In the Admin Console home page, click User Manager.**

   The User Manager page displays.

2. **Choose Group from the Create menu.**

   The Add Group Task Mentor dialog displays.

3. **Enter the group name, mail domain, and password. Press the Enter or Return key after entering each field.**

   The login name is case-insensitive. The mail domain must be fully qualified. For example, you could input the following for the distribution list widget_team:

   ```
   widget_team
   eng.bravo.com
   secret
   ```

4. **Click the Next button.**

   The next Add Group Task Mentor dialog appears.

5. **Identify the owner of the distribution list.**

   An owner is an individual who is responsible for a distribution list. An owner can add or delete distribution list members.

   a. **Click the Yes radio button if the owner is a mail user within your organization. Click the No radio button if the owner is not a mail user within your organization.**

   b. **Specify the owner's email address.**

      The email address must be fully qualified. For example, to specify Jane Campbell as the owner of the distribution list, enter the following email address:

      ```
      jane.campbell@eng.bravo.com
      ```

   c. **Click the Next button.**

      The next Add Group Task Mentor dialog appears.

6. **Do you want the distribution list to be moderated?**

   A moderator is an individual, usually the owner of the distribution list, who initially receives a message addressed to a distribution list. Upon receipt of a message, the moderator can forward the message to the distribution list, edit the message then forward it to the distribution list, or not forward the message to the distribution list.

a. **Click the Yes radio button if you want the distribution list to be moderated. Click the No radio button if you do not want the distribution list to be moderated.**

b. **If you decided to have the distribution list moderated, specify the moderator's email address.**

   The email address must be fully qualified. For example, to specify Bernie Miller as the moderator of the distribution list, enter the following email address:

   ```
   bernie.miller@eng.bravo.com
   ```

c. **Click the Next button.**

   The next Add Group Task Mentor dialog appears.

7. **Do you want the group members viewable by all users in the email system?**

   a. **Click the Yes radio button if you want the distribution list members to be viewable by all users in the email system. Click the No radio button if not.**

   b. **If you clicked the Yes button, you must set up a mail host through which the distribution list members can be viewed.**

      Enter a fully qualified mail host name. For example, to designate mailhost1 in the eng.bravo.com domain as the mail host through which the members can be viewed, enter the following:

      ```
      mailhost1.eng.bravo.com
      ```

      Users in the email system can view the list of members by establishing a telnet session with the specified mail host, specifying port 25, and using the following syntax:

      ```
      expn <distribution-list-name>
      ```

      For example, to view the distribution list of `widget_team`, enter the following command:

      ```
      expn widget_team
      ```

   c. **Click the Next button.**

      The next Add Group Task Mentor dialog appears.

**8. Add or delete distribution list members.**


**FIGURE 3-8** Add Group Task Mentor dialog

**a. To delete an existing member, click on the member entry in the Member screen to highlight it, then click the Delete button.**

**b. To add members, click on the Add button.**

The Add Member dialog appears.

**c. If a desired member is a mail user in your organization, click the Yes radio button. If the desired member is not a user in your organization, click No.**

Two versions of the Add Member dialog exist. The version that appears depends of if you selected the Yes or No button.

**d. If you are specifying a member who is a configured user within the organization, perform the following steps:**

**i. Search for the desired member's user entry by specifying a portion of the user's full name.**

A full name is any of the possible variations of a user's first name, last name, and middle initial. For example, if you want to specify Bernie Miller as a member, you can specify one of the following full names that appear in Bernie's user entry:

```
Bernard Miller
Bernie Miller
Bernard A. Miller
Bernie A. Miller
```

Click the Find button. Once the search is completed, the preferred recipient address(es) of the user entry(ies) that match the search parameters appears in the table. Click on the preferred recipient address of the desired member. If the search did not yield desired results, perform another search.

**ii. Click the Add button.**

**iii. Repeat Step b for each internal member you want to add.**

**iv. When you are finished adding internal members, click the Done button.**

**e. If you are specifying a member who is not a configured user within your organization, perform the following steps:**

**i. Specify the Internet address of the desired member.**

Enter an address in ASCII characters. You can enter the characters in either upper- or lowercase. For example, the following is a valid entry for desired member Bernie Miller: `bernie.miller@contractor.com`

**ii. Click the Add button.**

**iii. Repeat Step b for each external member you want to add.**

**iv. When you are finished adding external members, click the Done button.**

**f. To delete a member, click on the member entry in the display to highlight it, then click the Delete button.**

**9. Click the Create Group button.**

**10. If finished, click the Done button. If you need to create another group entry, click the Previous button until you are at the first Add Group Task Mentor dialog.**

## Creating Organizational Units

You can add an organizational unit in the directory information tree (DIT) if a corresponding mail domain in your Domain Name Service (DNS) exists. For example, if you add an organizational unit named Marketing to the DIT of the Bravo Corporation, then the mail domain `mkting.eng.bravo.com` (or something to this effect) must exist in Bravo's DNS. If the corresponding mail domain does not exist, the Administration Console will not allow you to add the organizational unit. For conceptual information on organizational units and the DIT, refer to the *Sun Internet Mail Server Installation Guide*.

**Note –** If you create a user not under the default domain, will not have message access unless the follow the instructions under “To Create an Organizational Unit That is Not Under the Default Domain” on page 67.

## ▼ To Create an Organizational Unit Under the Default Domain

In this example, the default domain (the domain specified at install) is `eng.bravo.com`. We will create an organizational unit called, `mkting.eng.bravo.com`.

AdminConsole>User Manager>>Create pulldown>Org Unit

1. **From the Admin Console home page, click on the User Manager icon**
2. **Click on the Create pulldown menu and select Org Unit.**

   The Add Organizational Unit dialog displays.
3. **Enter a name for the organizational unit.**

   For this example, name the organizational unit Marketing by entering:

   `Marketing`
4. **Enter the corresponding mail domain.**

   For this example enter: `mkting.eng.bravo.com`
5. **Click the Add button.**

   You can now create user entries in this mail domain. You may have to do a browser reload to display the new organizational unit.

## ▼ To Create an Organizational Unit That is Not Under the Default Domain

In this example, the default domain (the domain specified at install) is `eng.bravo.com`. We will create an organizational unit called `mkting.bravo.com`.

AdminConsole>User Manager>Create pulldown>Org Unit

1. **From the Admin Console home page, click on the User Manager icon.**

2. **Click on the Create pull-down menu and select Org Unit.**

   The Add Organizational Unit dialog displays.

3. **Enter a name for the organizational unit.**

   For this example, name the organizational unit Marketing by entering: `Marketing`

4. **Enter the corresponding mail domain.**

   For this example enter: `mkting.bravo.com`

5. **Click the Add button.**

   You may have to do a browser reload to display the new organizational unit. You will get an warning message says:

   ```
   Warning: The new OU is outside of the default domain, mail may not
   work for users under this OU.
   ```

   If users are created under this organizational unit, they will not get message access authentication, nor mail access. There are situations where you may want to create a second domain which is used only to route mail, and where users under this domain are not intended to have message access. However, if you wish to provide users under this domain with message access, then the following steps are required.

6. **To provide message access for users under this domain do the following**:

   a. **Edit `/etc/opt/SUNWmail/ims/ims.cnf`**

      Change the line `ims-basedn:ou=eng,o=sun,c=US`
      to `ims-basedn:o=sun,c=US`

   b. **Restart the Indexed Message Store.**

      ```
      # /opt/SUNWmail/ims/sbin/mt.scheduler start
      # /opt/SUNWmail/ims/sbin/mt.scheduler stop
      ```

   c. **In Admin Console go into IMTA property page Mail Server Domains section.**

      Add the newly added mail domain `mkgt.bravo.com`. See “To Configure Mail Server Domains” on page 116.

   d. **Apply the change and restart the im.server.**

      In the Admin Console Home Page select **Stop All**, and then selected **Start All,** or at the command line enter:

      ```
      # /opt/SUNWmail/admin/sbin/im.server stop
      # /opt/SUNWmail/admin/sbin/im.server start
      ```

   e. **Add users under the new organization.**

      When adding users through the Admin Console, “Add User Task Mentor Step Three” asks for a Mail Host and Mail Domain. Make sure the Mail Host is of the format `<server>.mkgt.bravo.com`, where `<server>` is your mail server host name, and the Mail Domain is of the format `mkgt.bravo.com`.

**f. Synchronize the IMTA directory cache.**

```
# imta dirsync -l <server>.eng.bravo.com,server.mkgt.sun.com
```

Newly created users under new organizational unit should be able to send and receive messages.

## ▼ To Find and View an Existing User/Group Entry

A user or group entry is viewed from its s Property Book. The User/Group Property Book contains common personal, system, mail server, and calendar server configuration information for a particular user or group stored in the LDAP directory.

| AdminConsole>User Manager> |
|---|

1. **From the Admin Console home page, click the User Manager icon.**

   The User Manager page displays as shown in FIGURE 3-1 on page 57.

2. **Selected the part of the DIT that contains entry you wish to view.**

   Click the root folder and any subsequent folders to view the portion of the DIT containing the desired entry. FIGURE 3-9 shows an example. Here we want to view a user entry in `dn:ou=people,ou=eng,o=sun,c=us`. If you had wanted to view a group entry, you would click on Group instead of People.

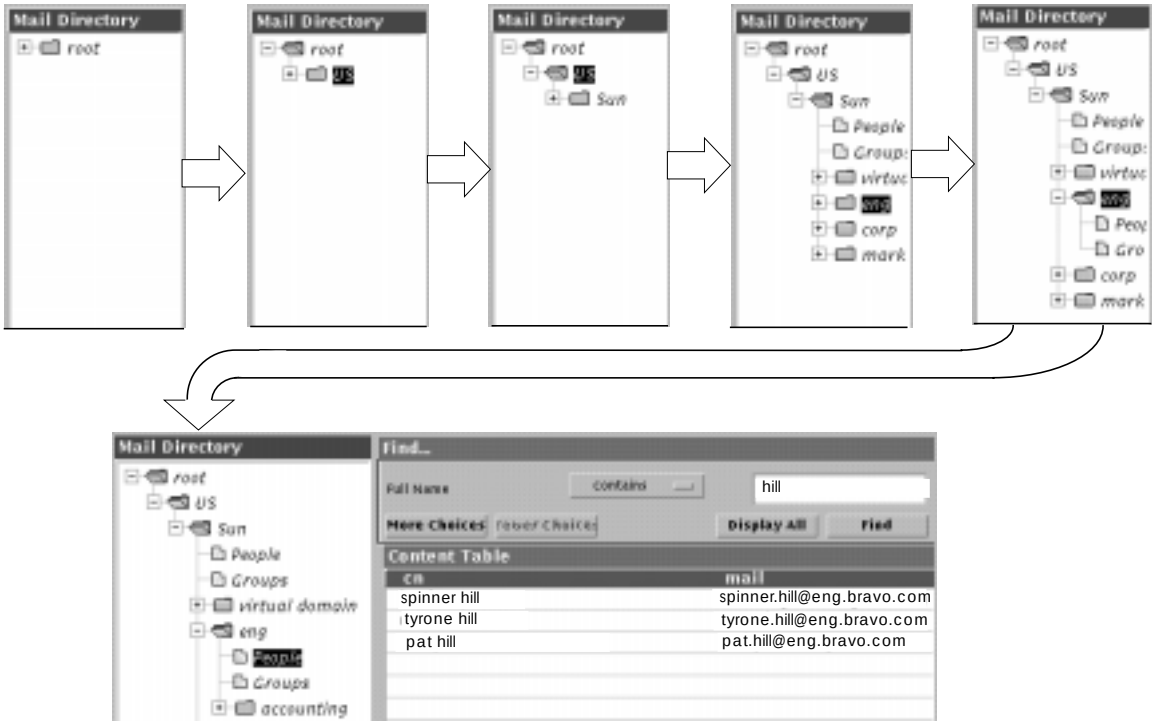



**FIGURE 3-9** Browsing the DIT

---

**Note –** You must select either People or Groups in the Mail Directory tree before doing a Find operation.

---

3. **Type the name or part of the entry you want to view and press Find, or press Display All to display entries without regard to find parameters.**

   In the example above we entered *hill*. You can also search by other parameters. Press More Choices to view these parameters (FIGURE 3-10).

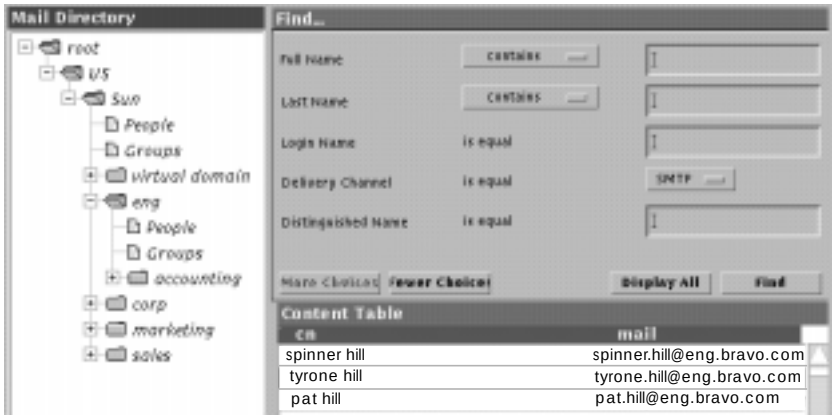



**FIGURE 3-10** Full Find Menu.

Pressing Find or Display All loads the first fifty entries. Load additional entries by scrolling down. The number of entries loaded will be equals Maximum Hit. (Maximum Hit can be configured by selecting Configure Maximum Hits from the User Manager pull down menu. The default is 2000.)

---

**Note –** If your browser does not scroll down, and you know you have more than 50 entries, you need to set the HotJava browser security property to low. See "Preventing the "Warning Applet" Banner" on page 252

---

4. **Once you find the entry you are searching for, double-click the entry.**

   The property book for that particular user or group appears (FIGURE 3-11). This property book is divided into sections that contain information for that particular user or group. For a description of each of the user or group entry fields, refer to TABLE 1-4 on page 29.

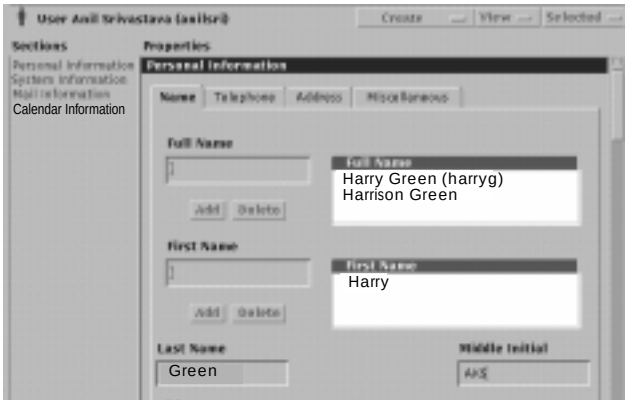



**FIGURE 3-11** User Property Book

## ▼ To Delete a User or Group Entry from the Directory

To remove a user from SIMS you must delete the user/group's entry from the SIMS directory, synchronize the cached directory with modified directory, and remove the user's folders and mailboxes from the SIMS mail store.

---

**Note –** Deleting a user entry does not remove the entry from distribution lists. See "To Modify a Group Entry" on page 82 to remove users from distribution lists.

---

| AdminConsole>User Manager>Display entry |
|---|

1. **Display the entry to delete in the Content Table of the User Manager Property Book.**

   See "To Find and View an Existing User/Group Entry" on page 69.

2. **Highlight the entry and choose Delete from the Selected menu.**

   A dialog box prompts you to confirm the deletion of the entry. Click OK. The entry is now removed from the SIMS LDAP directory.

3. **Synchronize the cached directory.**

   Even though the entry is removed from the SIMS directory, it still remains in the IMTA directory cache until the cache is synchronized with the SIMS directory. You must run a full directory synchronization using `imta dirsync -F` (see the *SIMS Reference Manual*).

4. **Remove the user's folders and mailboxes from the mailstore.**

   Wait 2 minutes after running `imta dirsync -F`, then execute the `imdeluser` command (see the *SIMS Reference Manual*). The short wait insures that the message queue is cleared before removing the folders and mailboxes.

## ▼ To Delete an Organizational Unit

You can delete an organizational unit from the directory information tree (DIT). Performing this operation causes the deletion of all folder and entries contained in that organizational unit. For example, if you delete the organizational unit named Marketing from the DIT of the Bravo Corporation, then all user entries in the People folder and all group entries in the Group folder contained in the Marketing organizational unit will be deleted.

| AdminConsole>User Manager> |
|---|

1. **From the Administration Console home page, click on the User Manager icon in the Tasks portion of the page.**

   The User Manager page displays.

2. **In the directory tree highlight the organizational unit label (e.g., Marketing), then click on the Selected menu and choose Delete.**

   A dialog prompts you to confirm deletion of the organizational unit.

3. **Click OK.**

## ▼ To Modify a User Entry

User entries are modified by displaying and configuring the user's property book.

| AdminConsole>User Manager> |
|---|

1. **Display the user's Property Book.**

See “To Find and View an Existing User/Group Entry” on page 69. The user’s property book contains a number of configurable property fields (see TABLE 1-4 on page 29 for a complete list of fields). The following fields are mandatory:

- Full name
- Last name
- Login name
- Password
- Mail host
- Delivery channel type
- Preferred recipient address
- Preferred originator address
- Mail aliases

If you specify Internet as the delivery channel type in the Mail Information section, you must also configure Internet mail delivery options. The configuration of all other fields is not required.

2. **Configure the fields in the Name section (**FIGURE 3-12**).**

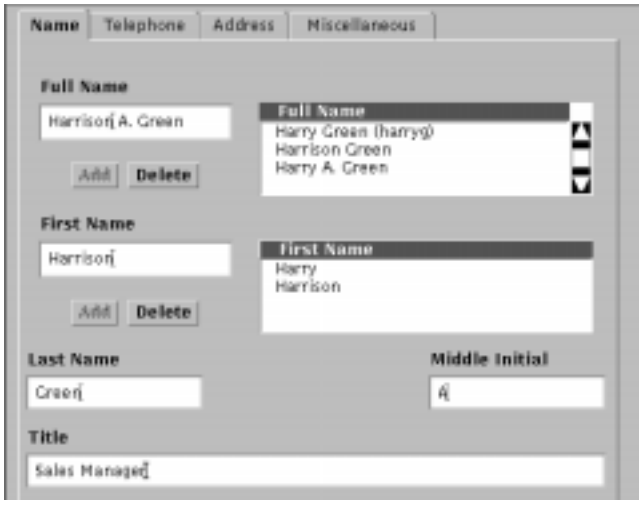



**FIGURE 3-12** Name Section

You must configure the full name and last name fields. All other fields in this section is not required.

**a. Enter full name(s).**

You can also enter variations of the full name. Click the Add button under the Full Name field for each full name you enter.

**b. Enter last name.**

Enter the same last name specified in the full name field.

c. **Optional: Enter the First Name, Middle Initial, and Title Fields if desired.**

For the first name field, you can enter first name variations. For each given name you enter, click the Add button under the First Name field.

3. **Optional: Enter the fields in the Telephone section (see** FIGURE 3-13**).**

Click on the Telephone tab. Enter the telephone numbers in any desired syntax. For each entry, click the Add button under the appropriate field.


**FIGURE 3-13** Telephone Section

4. **Optional: Enter the fields in the Address section.**

Click on the Address tab (FIGURE 3-14). Configure the desired fields.


**FIGURE 3-14** Address Section

5. **Optional: Enter the fields in the Miscellaneous section.**

   Click on the Miscellaneous tab (FIGURE 3-15).


**FIGURE 3-15** Miscellaneous Section

6. **Configure the fields in the System Information section.**

   Click on System Information in the Sections list (FIGURE 3-16). You must configure the login name and password fields. The home directory field is optional.


**FIGURE 3-16** System Information Section

   a. **Configure the login name field.**

      Enter a unique identification for the user in ASCII characters. Enter the characters in lowercase.

   b. **Configure the password field.**

      Enter a default password for the user in ASCII characters. You can enter the characters in either upper- or lowercase. For example, a valid entry is as follows:

      `Abra_CaDabra`

      For security reasons, the mail server by default stores the password in an en*crypted* or scrambled state. Later, the user can change the default password. For more information, refer to “To Change the Mail Password” on page 280.

      If the user has an existing encrypted password, you can use the following syntax to load the encrypted password into the mail server:

      {crypt}<*password*>

**c. Configure the home directory field if desired.**

**7. Configure the fields in the Mail Information section.**

Click on Mail Information in the Sections list (FIGURE 3-17).

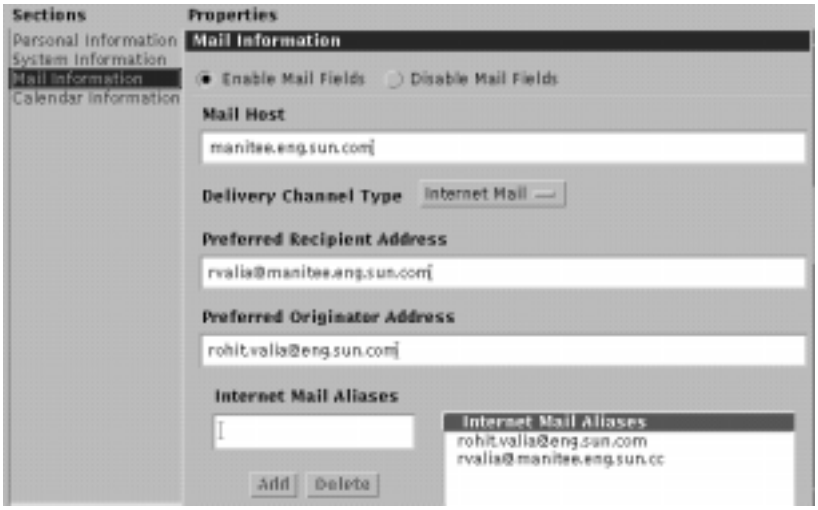



**FIGURE 3-17** Mail Information Section

For a mail user entry you must configure the mail host, delivery channel type, preferred recipient address, preferred originator address, and mail aliases fields in this section. The configuration of all other fields in this section is not required.

---

**Note –** There are two radio buttons labeled Disable Mail and Enable Mail in the mail information section. If an entry is defined as a calendar-only user, then the Mail Information section will be disabled. Later, if you wish to change the entry to support mail, you can click the Enable Mail button and enter mail information in this section.

---

**a. Configure the mail host field.**

Enter the hostname, including the full domain name, of the user's mail server in ASCII characters. Enter the characters in lowercase. For example, if the hostname for user Harry Green's mail server is `mailserver1` and this mail server exists in the `sales.bravo.com` domain, then the following is a valid entry:

```
mailserver1.sales.bravo.com
```

**b. Configure the delivery channel type field.**

Use the pull-down menu to select the user's delivery channel type. You can select the Internet mail channel or one of the Connectivity services channels.

**c. Configure the preferred recipient address field.**

Enter the email address that a recipient within the email system will see when a message from the user is received. Enter the address in upper- or lowercase ASCII characters. The format of the address must be in RFC 822 format:

```
harryg@mailserver1.sales.bravo.com
```

**d. Configure the preferred originator address field.**

Enter the email address that a recipient outside the email system will see when a message from the user is received. Enter the address in upper- or lowercase ASCII characters. The format of the address must be in RFC 822 format:

```
harry.green@sales.bravo.com
```

**e. Configure the mail aliases field.**

Enter the email addresses that you specified for the preferred recipient address and preferred originator address fields and any other email address that any recipient will see when a message from the user is received. Enter the address in upper- or lowercase ASCII characters. The format of the address must be in RFC 822 format:

```
harry.green@sales.bravo.com
```

Click the Add button under the mail aliases field for each address that you enter.

8. **If you specified one of the Connectivity services channels in Substep b on page 77, then go on to the next step. If you specified the Internet Mail channel, then you can also configure the Internet Delivery Options.**

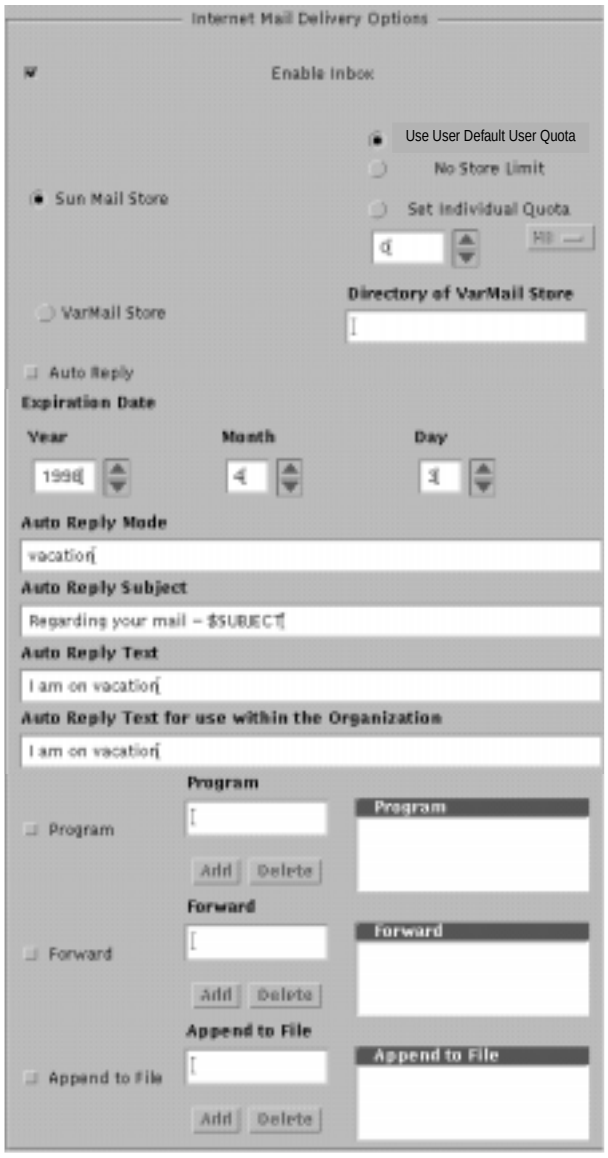



**FIGURE 3-18** Internet Mail Deliver Options

**a. Check Enable Inbox if you wish to read mail.**

**b. Press which message store the user's Inbox will reside in.**

Click the radio button for either the Sun Message Store (Sun Mail Store) or `/var/mail` (VarMail Store). We highly recommend the Sun Message Store as it is more secure, more space efficient, more centralized, and much more easy to back up than `/var/mail`.

**i. If you specified the Sun Message Store, set the maximum amount of hard disk space or *quota* that the user's mailboxes can occupy.**

This message store quota only takes effect if the User Quota Enforcement option in the Message Store Property Book is set to ON. (See "User Quota Enforcement" on page 152 and "To Configure Advanced Options" on page 155 for details.). The following size limit options are offered:

Use Default User Quota - Default user quota is set in the Advanced Options section of the Message Store Property Book. It is factory set to 20Mbytes.

No Store Limit - User has unlimited message store space.

Set Individual Quota - Select a number and the unit of measure (KB or MB). This quota will not take effect until an incremental or full directory synchronization occurs (see "Alias Synchronization Schedule" on page 104 or see the `dirsync`, `iminitquota`, and `imquotacheck` command in the *SIMS Reference Manual* for more information).

**ii. If you specified that the user's Inbox will reside in `/var/mail`, then a user directory will automatically be created in `/var/mail/<userID>`.**

If you want it to be under some other directory, you need to create it. Any mail sent to the user before the directory is created will be lost.

**c. Optional: enable the auto reply feature for the user by clicking the Auto Reply check box.**

The auto reply feature automatically generates a form reply to be returned to each sender who sends an email message to a user during a specified timeframe. This feature is typically used when a user is on vacation. The auto reply feature contains defaults for the following fields but you can reconfigure them:

- Expiration date - auto reply feature is disabled at midnight on specified date.
- Auto reply mode - currently the only mode available is "*vacation*".
- Auto reply subject - the subject line of the auto reply message. If you specify $SUBJECT, this token is replaced with the subject line of the incoming message.
- Auto reply text - the body of the auto reply message.
- Auto reply text for use with the Organization - the body of the auto reply message for use within an organization.

Enter the auto reply fields in ASCII characters. You can enter the characters in either upper- or lowercase.

Note that the user has access to this feature. See “To Start and Stop the Vacation Notice” on page 280.

**d. Optional: you can enable the delivery of email to UNIX programs by clicking the Program check box.**

Enter a pre-configured method name defined by the `imta program -a -m <method name> -p <program name>` command (See the *SIMS Reference Manual*, “To Make Delivery Programs Available to Users” on page 102 and “To Use Alternative Delivery Programs” on page 281.)

**e. Optional: you can enable the forwarding of email to specified addresses by clicking the Forward check box.**

When specifying a forwarding address, use the following syntax:

*<user>@<domain>*

For example, to forward a message to Harry Green, enter the following:

`harry.green@sales.bravo.com`

Enter the forwarding address in ASCII characters. You can enter the characters in either upper- or lowercase. You can provide multiple forwarding addresses. For each address, click the Add button under the Forward field. (See also “To Forward Mail” on page 282.)

**f. Optional: you can enable the appending of email to specified files by clicking the Append to File check box.**

Specify the full pathname of the file. For example, you can specify the following:

`/home/harryg/widget/component.txt`

The email will be attached to the end of the `component.txt` file. Enter the file name in ASCII characters. You can enter the characters in either upper- or lowercase. You can provide multiple file names. For each file name, click the Add button under the Append to File field.

9. **Configure the Calendar Information.**

   You can add Calendar information to the user entry. This will allow the user to maintain a calendar using the Web Access user application. Click on Calendar Information, then click the Enable Calendar radio button and enter the Calendar Host (mandatory) and the Default Calendar (optional). `rpc.cmsd` must be installed the calendar host.

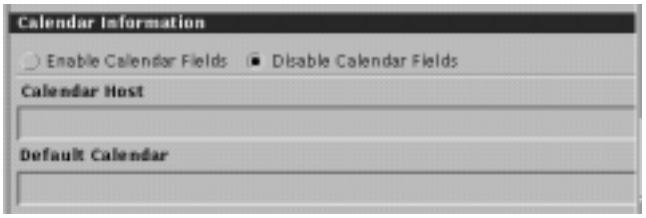



**FIGURE 3-19** Calendar Information

   If the entry is a *calendar-only* entry, the Internet Mail Delivery Options are disabled. That is, the Disable Mail radio button in the Internet Mail Delivery Options section will be pressed. If you press the Enable Mail radio button, then you must fill in the mandatory mail configuration fields: mail host, delivery channel type, preferred recipient address, preferred originator address, and mail aliases.

10. **When you have input required and optional fields for a user, click on the Apply button at the bottom of the Add User page.**

    If you entered a field incorrectly, an error message will identify the field. Refer to the documentation for the correct syntax and reenter the field. Click either the OK or Apply button.

## ▼ To Modify a Group Entry

This section describes the procedure for modifying a field in a group or *distribution list* entry. See “Distribution Lists” on page 18 for conceptual information.

| AdminConsole>User Manager>Group entry in DIT |
|---|

1. **Display the group entry property book.**

See “To Find and View an Existing User/Group Entry” on page 69. The group’s attributes are displayed as fields. You can modify fields or enter fields not previously entered. For a description of each of the group entry fields, refer to TABLE 1-5 on page 32.

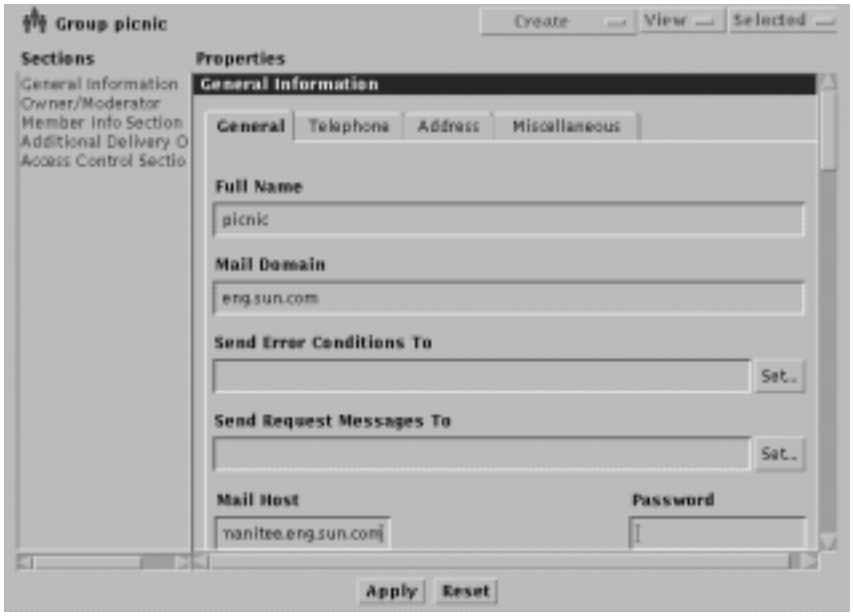



**FIGURE 3-20** Group Entry Property Book

2. **Modify the fields in the General section (**FIGURE 3-20**).**

   a. **Full Name and Mail Domain cannot be modified.**

   b. **Enter the Send Error Conditions To and the Send Request Messages To fields.**

      Send Error Conditions To specifies an address to send a message if a distribution list error condition arises. Send Request Messages To field specifies an address to send messages containing requests to be added to the distribution list. Click the Set button next to the desired field. The Address Lookup dialog appears. If the individual to which you want error condition or request notices sent is a mail

user within your organization, click the Internal radio button at the top of the dialog (FIGURE 3-21) or if the individual is not within your organization, click the External radio button (FIGURE 3-22).


**FIGURE 3-21** Internal Address Lookup Dialog


**FIGURE 3-22** External Address Lookup Dialog

To specify someone in your organization, search for their mail user entry by specifying their full name or a portion of it then clicking the Find button to display a list of matches. If the search did not yield desired results, perform another search. Click on the address of the desired user and click Add.

To specify someone outside your organization, enter their Internet address in

either upper- or lowercase ASCII characters. Click the Add.

**c. Configure the mail host field.**

Enter the hostname, including the full domain name, of the group's mail server in lowercase ASCII characters. If the hostname for the widget team's mail server is mailserver1 and this mail server exists in the sales.bravo.com domain, then the following is a valid entry:

```
mailserver1.sales.bravo.com
```

**d. Configure a password.**

Enter a default password for the group and the shared mailbox, if applicable, in ASCII characters. Enter the characters in either upper- or lowercase. For example:

```
Abra_CaDabra
```

This password is required when attempting to modify the group entry fields using the command line or the email administrator's configuration interface. For security reasons, the mail server by default stores the password in an *encrypted* or scrambled state.

Later, the group can change the default password using the email user's configuration interface. (Refer to "To Change the Mail Password" on page 280.)

If the group has an existing encrypted password, you can use the following syntax to load the encrypted password into the mail server:

{crypt}<*password*>

**e. Make the member list accessible to all users if desired.**

Click the check box labeled Expandable to make the distribution list members accessible to all users. Users can use the SMTP EXPN command to expand (get the membership of) distribution lists. If not checked, SMTP will have an Access to List Denied message.

**3. Optional: Enter the fields in the Telephone section.**

Click the Telephone tab to display the Telephone section (FIGURE 3-23). Enter the desired fields. You can provide multiple entries for each field in this section. For each entry, click the Add button under the appropriate field.

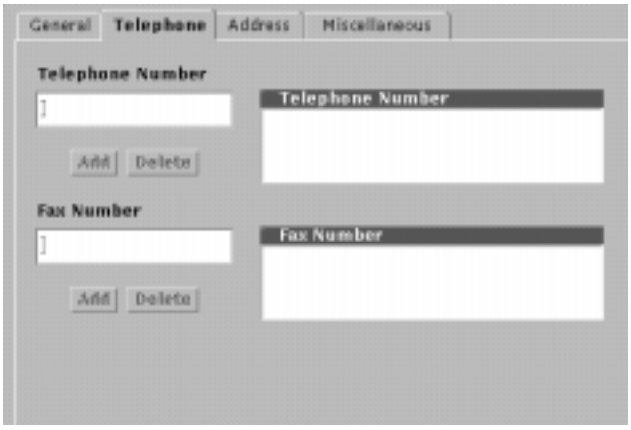



**FIGURE 3-23** Telephone Section

4. **Optional: configure the fields in the Address section.**

   Click the Address tab to display the Address section and fill in the address as desired.

5. **Optional: Complete the fields in the Miscellaneous section if desired.**

   Click the Miscellaneous tab in the General Information (FIGURE 3-24.)


**FIGURE 3-24** Miscellaneous Section

**6. Configure the fields in the Owner/Moderator section.**

Click on Owner / Moderator FIGURE 3-25).

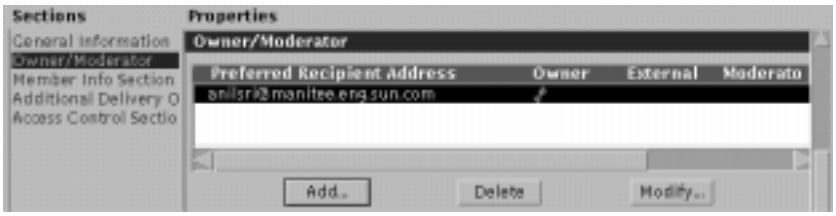



**FIGURE 3-25** Owner / Moderator Section

An *owner* is an individual who is responsible for a distribution list. An owner can add or delete distribution list members. A *moderator* is an individual, usually the owner of the distribution list, who initially receives a message addressed to a distribution list. Upon receipt of a message, the moderator can forward the message to the distribution list, edit the message then forward it to the distribution list, or not forward the message to the distribution list. External indicates that the address is not local to the mail system.

Although a distribution list is created with an owner, you can also configure a group as moderator only. Both owner and moderator fields are not required.

**a. To modify an existing owner, click the Modify button.**

Click on the check boxes labeled Owner and Moderator to modify the role(s) of the existing owner as appropriate. Click the Add button.

**b. To delete an existing owner, click on the owner entry in the Owner/Moderator screen to highlight it, then click the Delete button.**

**c. To configure an owner for the group, click on the Add button.**

**i. If the group owner is a user in the email system, click the radio button labeled Internal.** FIGURE 3-26 **is displayed. If the group owner is not configured as a user in the email system, click the External button.** FIGURE 3-27 **is displayed.**


**FIGURE 3-26** Internal Add Owner Dialog

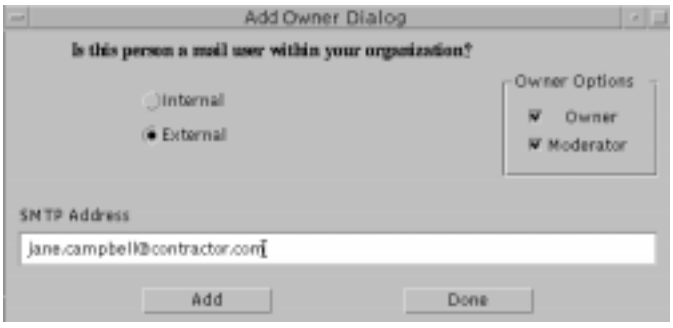



**FIGURE 3-27** External Add Owner Dialog

**ii. If the owner is a local user, perform a search for her entry by entering her name of a portion of it and clicking the Find button. Click on the preferred recipient address of the desired owner. If the search did not yield desired results, perform another search.**

Click the check box labeled owner. If desired, click the check box labeled moderator.

Click the Add button.

**iii. If the owner is not in the local email system, specify her Internet address.**

Enter address and click the check box labeled owner. If desired, click the check box labeled moderator. Click the Add button.

**7. Add or delete members to the group.**

Click on Member Information (FIGURE 3-28).

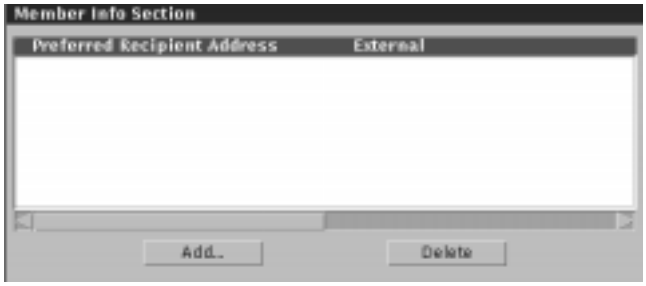



**FIGURE 3-28** Member Information Section

**d. To delete an existing member, click on the member entry in the Member screen to highlight it, then click the Delete button.**

**e. To add group members, click on the Add.**

**i. If the desired member is a user in the local email system, click the radio button labeled Internal.**

An internal Add Member Dialog is displayed (see FIGURE 3-21 on page 84). If the desired member is not configured as a user in the email system, click the External button to display external Add member dialog (see FIGURE 3-22 on page 84).

**ii. If the desired member is a local user, perform a search for her entry by entering her name of a portion of it and clicking the Find button.**

Click on the address of the new member.Click the Add button and repeat this step for each member you want to add to a group. If the search did not yield desired results, perform another search.

**iii. If the owner is not part of the local email system, enter her Internet address and click Add.**

Repeat this step for each member you want to add to the group.

**8. Optional: Configure the fields in the Additional Delivery Options section.**

Click on Additional Delivery Options (FIGURE 3-29) to send mail to a shared mailbox, to a UNIX program, or to append mail to a file.

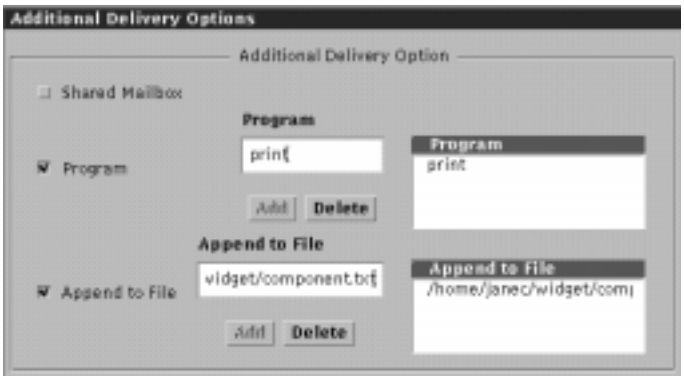



**FIGURE 3-29** Additional Delivery Options Section

**a. If the messages will be delivered to a shared mailbox in the Sun Message Store, click the check box labeled Shared Mailbox.**

Members can only access the shared mailbox from an IMAP server, and by entering the mailbox name as follows: `#shared/<distribution list name>`. Only the owner of the group entry can expunge a message from the shared mailbox. Members can expunge their *view* of the message from their own mailbox, but the message still remains in the shared folder until expunged by the owner.

**b. To enable the email delivery to UNIX programs, click the Program checkbox.**

Enter a pre-configured method name defined by the `imta program -a -m <method name> -p <program name>` command (See the *SIMS Reference Manual*).

**c. To append of email to specified files, by clicking the Append to File check box.**

Specify the full pathname of the file. For example, you can specify the following:

```
/home/janec/widget/component.txt
```

The email will be attached to the end of the `component.txt` file. You can provide multiple file names. For each file name, click the Add button under the Append to File field.

9. **Optional: Configure the fields in the Access Control section.**

   These fields block specified domains and users. If nothing is specified, anyone can send messages to the list. If a moderator is created, the message first goes to the moderator. Without a moderator, the message goes to all group members. The exact distribution list delivery algorithms are described in “Access Control” on page 20.

   To configure access control, click on Access Control to set these attributes.

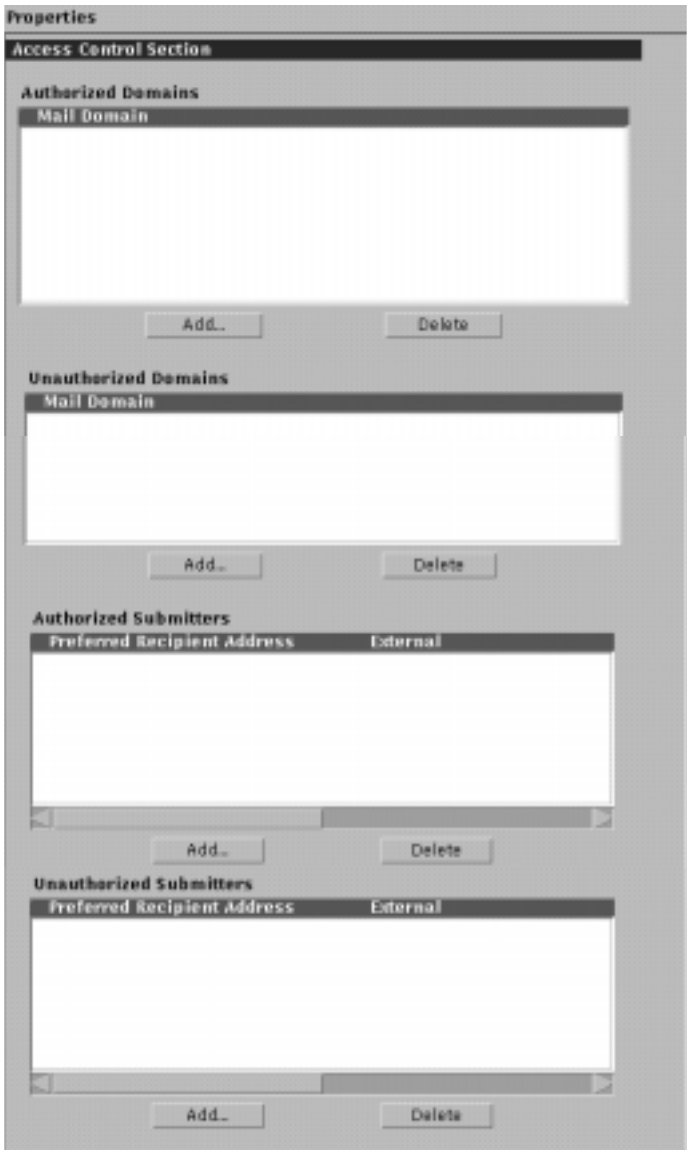



**FIGURE 3-30** Group Entry Access Control Section

**a. To delete an existing domain or submitter, highlight the entry click Delete.**

**b. To add an authorized or unauthorized domain, click on the Add button below either the Authorized or Unauthorized Domain screen.**

The Add Domain dialog appears as shown in FIGURE 3-31. Enter the unauthorized domain and click Add. Note that you can use the wildcard character (*) as part of the specified domain.

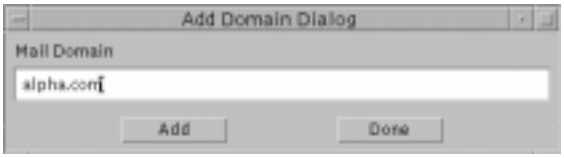



**FIGURE 3-31** Add Domain Dialog

**c. To add an authorized or unauthorized submitter, perform the following steps:**

**i. If the submitter is a user in the local email system, click the radio button labeled Internal.**

An internal Add Submitter Dialog is displayed (see FIGURE 3-21 on page 84). If the desired member is not configured as a user in the email system, click the External button to display external Submitter Dialog (see FIGURE 3-22 on page 84).

**ii. If the desired member is a local user, perform a search for her entry by entering her name of a portion of it and clicking the Find button.**

Click on the address of the new member. Click the Add button and repeat this step for each submitter to add to the list. If the search did not yield desired results, perform another search.

If you want to specify all members of the distribution list, you can specify the full name of the entry.

**iii. If the owner is not part of the local email system, enter her Internet address and click Add.**

Repeat this step for each member to add to the list. If you are specifying a submitter who is not a configured user or group in the email system, specify the Internet address of the desired submitter.

10. **When you have input required and not required fields for a group, click on either the OK or Apply button at the bottom of the Group dialog.**

    If you need to create a group entry for another group, click Apply for the Group dialog to remain open. If you have completed your task of modifying a group entry, click OK to close the Group dialog and return to the User Manager page.

    If you entered a field incorrectly, an error message will identify the field. Refer to the documentation for the correct syntax and reenter the field. Click either the OK or Apply button.

# Command Line User Management

"Populating the Directory" on page 167 explained how to populate the directory with the information required to support SIMS. Once the directory service is running, you need to maintain that directory information by adding, modifying, or deleting entries. This section summarizes the tools you can use to maintain directory information from the command line.

## ▼ To Create the Root Entry for XYZ Corporation

1. **Create an LDIF file called** `root-file` **that contains:**

```
dn: o=XYZ, c=US
objectClass: organization
```

   The *SIMS Reference Manual* describes the LDIF file format.

2. **Add this file using** `ldapadd`**:**

```
prompt% ldapadd -D "cn=admin, o= XYZ, c=US" -w secret -f root-file
```

   where the option `-D` introduces the distinguished name of the data store administrator, `-w` introduces the administrator password, and `-f` introduces the file holding the information to add to the database.

   The *SIMS Reference Manual* describes the `ldapadd` command.

   The root entry now exists. You can create as many new entries as you like.

## Adding Entries

You can add an entry to the directory using `ldapadd(1)`. You can specify a single entry on the command line, or you can specify one or more entries in a file. See the *SIMS Reference Manual* or the `ldapadd(1)` manpage for details of how to use `ldapadd`.

You can use `ldapsync` to read information from the `/etc/passwd` file or `/etc/aliases` file and create a file suitable for use with `ldapadd.`

## Modifying Entries

You can modify an entry in the directory using any of the following tools:

- Use `ldapmodify(1)`
  - to modify the attributes in a single entry, by specifying the modification on the command line
  - to modify multiple entries, by specifying a file containing entry modification information

  See the *SIMS Reference Manual* or the `ldapmodify(1)` manpage for details of how to use `ldapmodify`. `imldifsync` reads information from the `8/etc/passwd` file or `/etc/aliases` file and creates a file suitable for use with `ldapmodify`.
- Use `ldapmodrdn(1)` to modify the naming attribute of an entry. Changing the naming attribute changes the distinguished name of the entry. See the *SIMS Reference Manual* or the `ldapmodrdn(1)` manpage for details of how to use `ldapmodrdn`.
- Use the email administrator's configuration interface to modify certain attributes of an entry. See "Web Access to the Directory" on page 215" for more information.

## Deleting Entries

You can delete an entry in the directory using `ldapdelete(1)`. See the *SIMS Reference Manual* or the `ldapdelete(1)` manpage for details.

CHAPTER 4

# Internet Message Transport Agent (IMTA) Administration

This chapter provides step-by-step instructions for changing the message transport characteristics of SIMS. To start, bring up the IMTA property book pages.

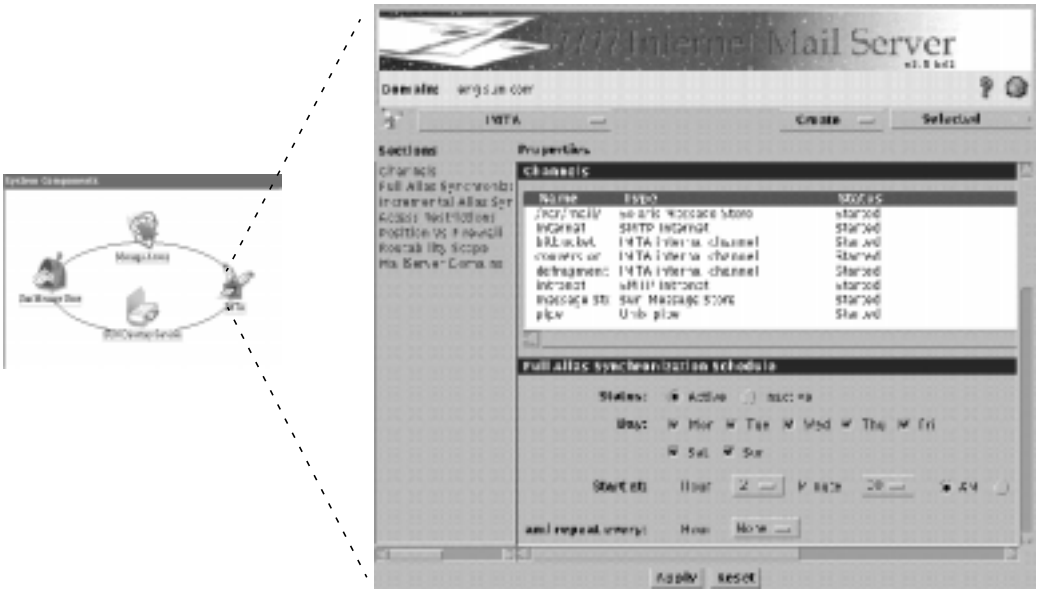



**FIGURE 4-1** (Internet Mail Transport Agent (IMTA) Property Book

Each *Section* displays configurable attributes of a particular IMTA property. The *IMTA* pulldown menu allows you to start/stop the IMTA as well as save or restore IMTA configuration. The *Create* pulldown menu allows you to create new channels. The *Selected* pulldown menu allows you to view a channel's properties, start, stop and delete channels, and monitor the message queue.

# IMTA Topics and Tasks

**TABLE 4-1** Message Transport Topics and Tasks

| Topic/Task | Description | Page |
|---|---|---|
| IMTA Maintenance Tasks | Stopping, starting, and restarting the IMTA. Stopping, starting, and restarting Connectivity Service Channels. Also Backup and restore of IMTA configuration. | 97 |
| Notary Message Locale | Changing the language of notary messages. | 100 |
| Monitoring Channel Status | View the operating status of the SIMS channels | 101 |
| Alternative Delivery Programs | Providing user access to alternative delivery programs. | 102 |
| Alias Synchronization Schedule | Schedule when the IMTA directory cache is incrementally or fully up-dated with the latest directory information in the directory service. | 104 |
| SMTP Access and Relay Restrictions | Restrict/limit access or delivery of messages from specified email and IP addresses. Spam control, limiting email access, etc. | 106 |
| IMTA Location Relative to Public Internet | Specify location of the IMTA relative to a firewall. | 113 |
| Routability Scope | Specify to whom IMTA can route messages—users, domains etc. | 114 |
| **Channels** | **This section and its subsections below describe channel configuration.** | **117** |
| Configuring Channels | Creating channels, channel descriptions, configuring SMTP router hosts, setting character set labels. | 118 |
| Message Limitation | Set message size limitations. | 125 |
| Delivery Status Notification | Specify how delivery failure message is handled. | 126 |
| Diagnostics Output | Writing master/slave diagnostic output to a log file | 128 |
| Performance Tuning | Specify the maximum number of recipients for a single message at which message processing is deferred on the SMTP channel. | 129 |
| Message Logging | Configure channel so that it logs as each message enters and is removed from the queue | 130 |
| Reassembling MIME Messages | Enable the Sun Message Store channel, `/var/mail` channel, and pipe channel to reassemble MIME fragmented messages. | 131 |
| Rewrite Rules | Add, delete, or modify channel rewrite rules. | 132 |
| Monitoring Channel Queues | Monitor accumulated message traffic statistics for each IMTA channel queue. | 135 |

**TABLE 4-1** Message Transport Topics and Tasks

| Topic/Task | Description | Page |
|---|---|---|
| Viewing Enqueued Messages | View a list of the messages currently stored in the channel queue. | 138 |
| Connectivity Channels | See the Sun Messaging Connectivity Services Channel Guides | 139 |
| **IMTA Error Messages** | Appendix E, “Error Messages | 359 |

# IMTA Maintenance Tasks

## Stopping, Starting, and Restarting a Channel or the IMTA

You may need to stop and start, or restart a channel under the following circumstances:

- If you experience a problem with an IMTA channel, you may need to stop the channel, resolve the problem, then start the channel again.
- If you reconfigure an attribute of a channel, you will need to restart the channel so that the reconfiguration takes effect. (Typically, when you reconfigure a channel attribute using the Admin Console, you are prompted to restart the channel.)

You can stop, start, and restart the connectivity services channels individually. If you need to stop, start, or restart any of the IMTA channels, (for example, the firewall SMTP channel), you will actually need to stop, start, or restart the entire IMTA.

To minimize the time it takes to stop / start or restart the IMTA, try to plan these operations during a part of the day when mail traffic is light.

### ▼ To Stop And Start the Internet Message Transfer Agent

AdminConsole>IMTA>IMTA pulldown>Start IMTA

1. **In the IMTA property book, choose Stop IMTA from the IMTA menu.**

   The IMTA closes its connections and shuts down. Depending on the amount of email traffic present, shutdown should take a few minutes.

2. **Resolve whatever problem exists with the channel or IMTA.**

3. **Click on the IMTA pull-down menu and select Start IMTA.**

   The IMTA reestablishes its connections and starts up. Startup should take a few minutes.

## ▼ To Restart the Internet Message Transfer Agent

AdminConsole>IMTA>IMTA pulldown>Restart IMTA

1. **From the IMTA property book, click on the IMTA menu.**
2. **Choose Restart IMTA.**

   The IMTA and all channels except the Connectivity services channels restart. This operation takes a few minutes.

## ▼ To Stop then Start the Connectivity Services Channels

AdminConsole>IMTA>Channels

1. **From the IMTA property book, click on Channels from the Section list.**

   The Channels section appears. This section displays a list of installed channels.
2. **Click on the Connectivity services channel that you want to stop.**

   The channel you want to stop is highlighted.
3. **Click on the Selected pull-down menu and select Stop Channel.**

   The channel closes its connections and shuts down. Depending on the amount of email traffic present, shutdown should take a few minutes.
4. **Resolve whatever problem exists with the channel.**

   For information on specific channel-related problems and how to resolve them, refer to the *Sun Messaging Connectivity Services Channel Guides*.
5. **Click on the Selected pull-down menu and select Start Channel.**

   The channel reestablishes its connections and starts up. Startup should take a few minutes.

## ▼ To Restart the Connectivity Services Channels

AdminConsole>IMTA>Channels

1. **From the IMTA property book, click on Channels from the Section list.**

   The Channels section appears. This section displays a list of installed channels.

2. **Click on the Connectivity services channel that you want to restart.**

   The channel you want to stop is highlighted.

3. **Click on the Selected pull-down menu and select Restart Channel.**

   The restart operation takes a few minutes.

## Backing Up and Restoring the IMTA Configuration

After the SIMS is installed, the server saves, or *backs up,* the IMTA, the Sun Message Store, and the directory service configuration. This configuration version is known as the *default configuration*. Subsequently, you can backup the latest IMTA configuration (also called the *current configuration*) at any time. The saved configuration is known as the *backup configuration*. Doing a backup overwrites the existing backup.

If for some reason you wish to use a previous configuration, you can restore one of the following configuration versions:

- Default configuration.
- Backup configuration (provided that this exists)

### ▼ To BackUp And Restore the IMTA Configuration

| AdminConsole>IMTA>IMTA pulldown>Save Current Config |
|---|

1. **In the IMTA property book, click the IMTA pulldown menu and choose Save Current Config.**

   The current configuration is backed up.

2. **To restore a previous configuration, click the IMTA pulldown menu and choose Restore Default Config or Restore Backup Config depending on which you desire.**

   The backup configuration is restored.

---

**Note –** Backing up the IMTA configuration does not back up the Sun Connectivity Services Configuration.

---

# Notary Message Locale

To change the default locale (C) so that *notary messages* (text messages sent by the IMTA to an email sender indicating delivery or non-delivery status of a sent message) appear in a different character set, you will need to create a separate locale directory under the IMTA configuration directory and edit `imta_tailor` file such that the `IMTA_LANG` points to the new locale directory.

## ▼ To Change the Notary Message Locale

For example, if you want the notary messages to appear in Japanese, do the following:

1. **Create a directory for the Japanese locale in /etc/opt/SUNWmail/imta/locale:**

   ```
   % mkdir /etc/opt/SUNWmail/imta/locale/ja
   ```

2. **Create a directory under the ja directory to hold the messages:**

   ```
   % mkdir /etc/opt/SUNWmail/imta/locale/ja/LC_MESSAGES
   ```

3. **Copy the nine message files from the /etc/opt/SUNWmail/imta/locale/C/LC_MESSSAGES (default) directory into the /etc/opt/SUNWmail/imta/locale/ja/LC_MESSAGES directory.**

   The files are:

   ```
   return_bounced.txt     return_delivered.txt   return_prefix.txt
   return_deferred.txt    return_failed.txt      return_suffix.txt
   return_delayed.txt     return_forwarded.txt   return_timedout.txt
   ```

4. **Translate the message text in the files into the Japanese character set.**

   You may also wish to provide the message text in English as well as any other local languages for senders who do not speak Japanese.

5. **Edit the tailor file (/etc/opt/SUNWmail/imta/imta_tailor).**

   Change the line:

   ```
   IMTA_LANG=/etc/opt/SUNWmail/imta/locale/C/LC_MESSAGES
   ```

   to

   ```
   IMTA_LANG=/etc/opt/SUNWmail/imta/locale/ja/LC_MESSAGES
   ```

6. **Restart the IMTA.**

# Monitoring Channel Status



All channels can be monitored by SIMS. This feature can be helpful in diagnosing various problems.

## ▼ To Monitor Channel Status

AdminConsole>IMTA>Channels

1. **Bring up the IMTA property book.**
2. **Click on Channels From the Sections list.**

   The Channels section appears, as shown in FIGURE 4-2.

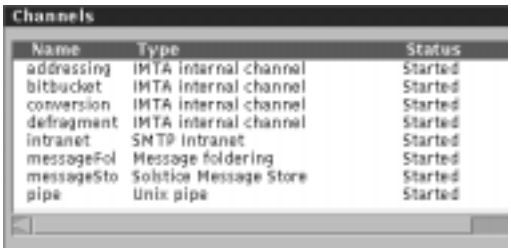



**FIGURE 4-2** Channels Section

The section displays a list of installed channels and channels that you created (channel creation applies to Sun Internet Mail Server - Enterprise Edition only). Check the status of each channel to determine if it is up (Started) or Stopped. If the channel is Stopped, use the Log Manager to help diagnose the problem (refer to “Event Log Manager” on page 249).

# Alternative Delivery Programs

Users may want incoming mail passed to a program instead of to their mailbox. For example, users may want their incoming mail sent to a mail sorting program like Personal Postmaster and procmail, or to a auto-reply agent like Vacation Notice. Alternative delivery programs can be added to the user interface using the `imta program` command (see "To Make Delivery Programs Available to Users" on page 102 and "To Use Alternative Delivery Programs" on page 281). Alternative delivery programs must, however, conform to the following exit code and command line argument restrictions:

*Exit codes*: If the subprocess exits with exit code of 0 (`EX_OK`), the message is presumed to have been delivered successfully and is removed from IMTA's queues. If it exits with an exit code of 71, 74, 75, or 79 (`EX_OSERR`, `EX_IOERR`, `EX_TEMPFAIL`, or `EX_DB`), a temporary error is presumed to have occurred and delivery of the message is deferred. If any other exit code is returned, then the message will be returned to its originator as undeliverable. These exit codes are defined in the system header file `sysexits.h`.

*Command Line Arguments*: Delivery programs can have any number of fixed arguments as well as the variable argument, `%s`, representing user name for programs executed by the user or *username+programlabel* (*programlabel* is a string that refers to the particular program) for programs executed by the postmaster, "inetmail." For example, the following command line delivers a recipient's mail using the program `procmail`:

```
/usr/lib/procmail -d %s
```

## ▼ To Make Delivery Programs Available to Users

These procedures add a delivery program to the User Profile described in "To Use Alternative Delivery Programs" on page 281.

1. **Obtain delivery program executable.**

   Program must conform to the format specified in "Alternative Delivery Programs" on page 102.

2. **Create a symbolic link from the actual executable to `/opt/SUNWmail/imta/programs`**

   Make sure the actual executable has execute permissions for "`others`."

3. **Use the `imta program -a` command to add a new delivery program option.**

   Run `imta program` as root or inetmail (see *SIMS Reference Manual* for details). The format is as follows:

   ```
   imta program -a -m method_name -p program_name [ -g
   argument_list ] [ -e execute_permission ]
   ```

   a. **Examples:**

      Add a delivery program called `procmail1`, which executes the program `procmail` in the programs directory. Use the argument `-d username`, and make this program execute as the user. Use the `-e user` argument so that this option is available only to mail users with UNIX accounts:

      ```
      % imta program -a -m procmail1 -p procmail -g "-d %s" -e user
      ```

      Add a delivery program `print_hickory`, which executes the program `lp` with the arguments `-d hickory`. Make this program option available to all mail users.

      ```
      % imta program -a -m print_hickory -p lp -g "-d hickory"
      ```

4. **Optional: Add a description of the delivery program.**

   Write a description of the delivery program—what it does, what it's used for, how to use it, and so forth. Put the information in a help file called `method_name.html` in `/opt/SUNWmail/imta/programs/locale/{en}` (change locale as necessary). The file need only be a "partial" html file (example below). The contents will be added to the Delivery Programs Options page. In addition to a text description of the program, you can also add a hypertext link to another help file.

   a. **Example 1:**

      This file, `procmail1.html`, provides a description of procmail and hypertext link to the procmail man page.

```
< p ALIGN="right" >
< a href="http://tools/man/procmail"> help < /a >
< p ALIGN="left" >
procmail1 is an email sorting program which is available for UNIX
users only.
< /p >
```

   b. **Example 2:**

      This file, `print_hickory.html`, provides a description of print_hickory.

```
< p >
print_hickory sends your mail to the printer, hickory
< /P >
```

5. **Do a full directory synchronization to add the newly created delivery programs to the database and make them available to users.**

   ```
   # imta dirsync -F
   ```

   Refer to Chapter 10, “User Administration” for details on how users can choose these delivery programs.

# Alias Synchronization Schedule



Rather than performing a directory query for each message that it processes, the IMTA caches directory information that is needed for processing a message. The directory information stored in the directory service is continuously updated. As a result, the directory information in the IMTA-directory cache must be synchronized periodically with the directory information in the directory service. (Note that the cache persists and is updated through an incremental or full `dirsync`.)

The IMTA supports full and incremental synchronizations.

- Full synchronization – The existing cache is replaced with a new cache completely rebuilt from information in the directory. After the synchronization occurs, the IMTA configuration file is rebuilt and then the IMTA is automatically restarted.
- Incremental synchronization – The existing cache is updated with user and group entries that were created or modified since the last full or incremental synchronization. The IMTA is not restarted.

Specifically, during an incremental synchronization, the directory information in the cache is updated with:

- User entries – New user entries and modifications to existing user entries. The cache is not updated with deleted user entries.
- Group entries – New and deleted members of existing distribution lists and new distribution lists. The cache is not updated with deleted distribution lists or new rules for existing distribution lists. New distribution list is also updated.

---

**Note – Important!** You must schedule each IMTA-directory cache to be fully or incrementally resynchronized at the same time. Not doing so could cause routing loops to occur.

---

By default, the IMTA-directory cache is fully synchronized every day at 2:00 am and incrementally synchronized every hour. For more information on the IMTA-directory cache, including synchronization schedule planning, refer to “Internet Message Transfer Agent-Directory Cache” on page 11.

## ▼ To Reconfigure the Alias Synchronization Schedule

AdminConsole>IMTA>Full Alias Synchronization (also Incremental Alias Synchronization)

1. **Click the IMTA icon on the Admin Console home page to bring up the IMTA property book.**
2. **From the Sections list, click Full Alias Synchronization Schedule.**

   The Full Alias Synchronization Schedule section appears followed by the Incremental Alias Synchronization Schedule section as shown in FIGURE 4-3.

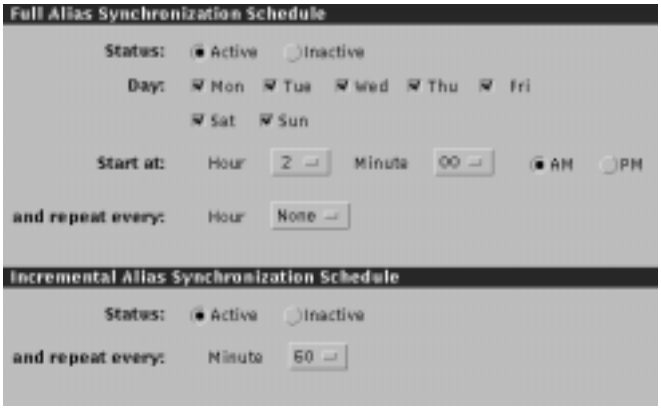



**FIGURE 4-3** Schedule for Synchronizing Aliases Section

3. **Configure the full synchronization schedule.**

   a. **Click the Active button in the Status field to enable full synchronization.**

   b. **Click next to the days on which you want full synchronization to occur.**

   c. **Specify the time at which you want the first full synchronization to occur.**

**d. If you want multiple full synchronizations to occur per day, use the menu to specify how often the synchronizations should occur.**

For example, if you specify that full synchronizations should occur every four hours, then they will occur six times per day.

**4. Configure the Incremental Alias Synchronization Schedule.**

**a. Enable incremental synchronization by clicking the Active button.**

**b. Specify how often the incremental synchronization should occur.**

### ▼ To Disable Full and Incremental Synchronization

AdminConsole>IMTA>Full Alias Synchronization (or Incremental Alias Synchronization)>Inactive

1. **Access the IMTA property book by clicking the IMTA icon on the home page.**
2. **Click the Inactive radio button in the full or incremental synchronization section.**
3. **Click the Apply button.**

## SMTP Access and Relay Restrictions



The Message Access and Relay Restriction feature allows you to restrict messages from passing through SIMS based on source/destination email address, IP address, and domain. This feature provides several types of functionality:

- Limits spam by blocking unwanted mail
- Limits spam by not relaying (sending mail from one domain to another) unwanted mail
- Restricts email usage to internal users

Email can be blocked by the following specified elements:

- By source or destination domain
- By source client IP address
- By destination server IP address

- By source or destination email address

## ▼ To Configure Access and Relay Restrictions

AdminConsole>IMTA>Access Restrictions

1. **In the Sections list of the IMTA property book, click Access Restrictions.**

   The Access Restrictions section appears.

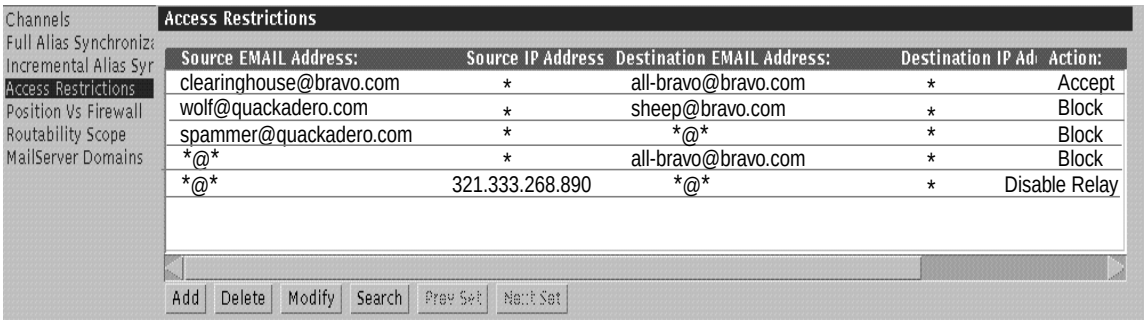



**FIGURE 4-4** Access Restrictions Section

A list of the first fifty *access restriction rules* appears in the display (press Next Set or Prev Set to display the next or previous fifty rules). An access restriction rule is a rule applied to a message which determines whether SIMS will block or accept the message. The order of rules displayed is the order in which each rule is applied to an incoming message before forwarding the message. The first rule that applies to the message will have its action applied to the message. Each rule consists of the following parameters:

*Source EMAIL Address* is the address of the sender on which to take action.

*Source IP Address* is the client IP address from which a message has been sent.

*Destination EMAIL Address* is the address of the recipient of the mail.

*Destination IP Address* is the server IP address at which the mail has been received.

*Action* is the action to take for a message specified by the preceding parameters. It can be *Accept* (accept message or forward to next stop), *Block* (refuse message delivery and return to sender), *Disable Relay* (refuse message from an external domain directed to another external domain and return mail to sender). Rules based

on IP addresses alone are applied before all other rules. If the `/etc/opt/SUNWmail/imta/mappings` file is edited by hand to include any other action, it will be shown as * in the console.

Rules are automatically sorted from most to least specific with *Source EMAIL Address* being used as the primary sort key, and *Destination EMAIL Address* being used as the secondary key. For example, a rule with a source email address as `*@stork.env.sunny.com` would be higher on the list compared to a rule with source email address `*@*.env.sunny.com`.

In FIGURE 4-4 the first rule allows `clearinghouse@bravo.com` to send messages to `all-bravo@bravo.com`. The second rule blocks mail from `wolf@quackadero.com` to `sheep@bravo.com`. The third rule blocks mail from `spammer@quackadero.com` from being delivered. The fourth rule blocks delivery of mail to `all-bravo@bravo.com`. The fifth rule stops inter-domain messages from IP address 321.333.268.890 from being delivered to the next domain.

You could also configure users within a particular company or group to be restricted to only sending mail to each other. The access rules for this are shown below.

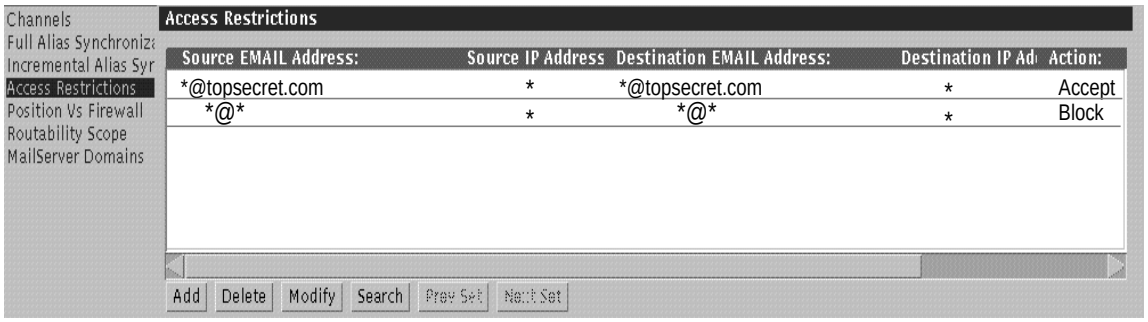



**FIGURE 4-5** Restricting Access to Users Within a Company

2. **To add an Access Restriction rule click Add. The Access Restriction Dialog appears:**


**FIGURE 4-6** Access Restriction Dialog

**a. Specify the desired Access Restriction parameters and click Done**

Enter email addresses for the Source and Destination EMAIL Address fields. Asterisks may be used as a wildcard character. If no IP address is entered, the default *.*.*.* is entered.

Addresses must contain a local-part and a domain part, separated by an @ sign. The wild card character * may be used, with the following restrictions:

i. Wild cards in addresses cannot be used on the right of non-wildcarded areas. That is: username@*, username*@*, or username@japan.* are illegal, *@*.com, *@xyz.com are legal.

ii. Wild cards cannot be used within an address token. A wild-card may only replace one or more entire domain part token. So for instance, *@*sun.com is illegal.

iii. Wild-cards may be used in local parts as long as they replace the entire username. As such, username*@sun.com is illegal.

**b. Resolve conflicting rules.**

If you try creating a rule that conflicts with previously written rule, the Admin Console will bring up a dialog box that shows all the rules with which the current rule conflicts. You must then resolve the conflict by modifying the fields of the rule.

An example of a conflicting rules is shown below:

| | **Rule A** | **Rule B** |
|---|---|---|
| Source address: | `*@*.quacky.com` | `*@*.quacky.com` |
| Destination address: | `*@eng.bravo.com` | `*@eng.bravo.com` |
| Action: | `Block` | `Accept` |

A conflict occurs if a message from `usr1@eng.quacky.com` is addressed to `usr2@eng.bravo.com`. Rule A says to block this message, and rule B says to accept. This conflict must be resolved before the new rules will be accepted. More complex rules conflict may occur, see "Conflicting Access Restriction Rules" on page 110 for a more indepth discussion.)

c. **After a rule is set press OK to add the rule and keep dialog up, or press Done to add the rule and close the dialog.**

   To save the rules press Apply. You are prompted to restart the IMTA, which will incorporate the new or modified rule.

3. **To modify an Access Restriction rule, select the rule and click Modify.**

   Modify parameters as desired and press Apply to save.

4. **To delete an Access Restriction rule, select the rule and click Delete and Apply.**

5. **To search for an Access Restriction rule, click Search and add the search parameters in the Access Restriction Dialog.**

   You can use the wildcard (*) character. Search is used for finding a particular set of access restriction rules that you wish to modify, verify, or delete.

## Conflicting Access Restriction Rules

The previous section described a very obvious case of rules conflict. A less obvious case of conflict arises when Rule 1 is less specific than Rule 2 according to one parameter, and more specific than rule 2 according to another parameter. For example, suppose we have defined the following two rules:

| | Source address: | Destination address: | Action: |
|---|---|---|---|
| **Rule 1** | `*@hosta.a.com` | `*@*.b.com` | `Block` |
| **Rule 2** | `*@*.a.com` | `*@hostb.b.com` | `Accept` |

These rules would not resolve the scenario where mail from `*@hosta.a.com` is to be delivered to `*@hostb.b.com`. To create a set of rules that would address the various delivery scenarios involved with these two addresses, we need to determine whether to block or deliver in these two specific scenarios:

| | Source address: | Destination address: | Action: |
|---|---|---|---|
| **Scenario 1** | `*@hosta.a.com` | `*@hostb.b.com` | ? |
| **Scenario 2** | `*@*.a.com` | `*@*.b.com` | ? |

The outcomes of these scenarios would be described by Rules 3 and 4 described below in the Outcome sections. These rules may or may not already exist.

The following matrix shows Scenarios 1 and 2, their specified actions (Block or Accept), and the rules needed to create the desired outcomes.

| Scenario 1 / Scenario 2 | Block | Accept |
|---|---|---|
| Block | Outcome A:<br>Scenario 1-Block<br>Scenario 2-Block | Outcome B<br>Scenario 1 - Accept<br>Scenario 2 - Block |
| Accept | Outcome C<br>Scenario 1-Block<br>Scenario 2-Accept | Outcome D<br>Scenario 1 - Accept<br>Scenario 2 - Accept |

## *Outcome A*

Here are rules resolving Outcome A. (Rule 1 not necessary, it is displayed for edification.) Rules are displayed in the sorted order, most to least specific with *Source* being the primary key, and *Destination* being the secondary key, and the order in which each rule would be applied to an incoming message before forwarding.

| | Source address: | Destination address: | Action: |
|---|---|---|---|
| Rule 3 | `*@hosta.a.com` | `*@hostb.b.com` | `Block` |
| * Rule 1 | `*@hosta.a.com` | `*@*.b.com` | `Block` |
| Rule 2 | `*@*.a.com` | `*@hostb.b.com` | `Accept` |
| Rule 4 | `*@*.a.com` | `*@*.b.com` | `Block` |

- Rule 1 is not needed because in the following delivery scenario the delivery is not matched by Rule 2 and covered by Rule 4:

  `*@hosta.a.com  —>  *@<any host but hostb>.b.com`

- Rule 3 is needed because Rule 2 applies to the following scenario. The conflict can be resolved by using the rules 2, 3, and 4. Rule 4 may not be needed if it is coverd by a more generic rule.

  `*@hosta.a.com  —>  *@hostb.b.com`

## *Outcome B*

Here are the rules needed to resolve Outcome B:

| | Source address: | Destination address: | Action: |
|---|---|---|---|
| *Rule 3 | `*@hosta.a.com` | `*@hostb.b.com` | `Accept` |
| *Rule 1 | `*@hosta.a.com` | `*@*.b.com` | `Block` |
| Rule 2 | `*@*.a.com` | `*@hostb.b.com` | `Accept` |
| Rule 4 | `*@*.a.com` | `*@*.b.com` | `Block` |

- Rule 1 is not needed because in the following delivery scenario

  `*@hosta.a.com  —>  *@<any host but hostb>.b.com`

  the delivery is not matched by Rule 2 and covered by Rule 4.
- Rule 3 is not needed because Rule 2 applies to the scenario

  `*@hosta.a.com  —>  *@hostb.b.com`

Hence the conflict can be resolved by using the rules 2, and 4. Rule 4 may not be needed if it is coverd by a more generic rule.

### *Outcome C*

Here are the rules needed to resolve Outcome C:

| | Source address: | Destination address: | Action: |
|---|---|---|---|
| ***Rule 3** | `*@hosta.a.com` | `*@hostb.b.com` | `Block` |
| **Rule 1** | `*@hosta.a.com` | `*@*.b.com` | `Block` |
| ***Rule 2** | `*@*.a.com` | `*@hostb.b.com` | `Accept` |
| **Rule 4** | `*@*.a.com` | `*@*.b.com` | `Accept` |

- Rule 2 is not needed because in the following delivery scenario

  `*@<any host but hosta>.a.com  —>  *@hostb.b.com`

  the delivery is covered by Rule 4 and doesn't match Rule 1.
- Rule 3 is not needed because Rule 1 applies to the scenario

  `*@hosta.a.com  —>  *@hostb.b.com`

Hence the conflict can be resolved by using the rules 1, and 4. Rule 4 may not be needed if it is coverd by a more generic rule.

### *Outcome D*

Here are the rules needed to resolve Outcome D:

| | Source address: | Destination address: | Action: |
|---|---|---|---|
| ***Rule 3** | `*@hosta.a.com` | `*@hostb.b.com` | `Accept` |
| **Rule 1** | `*@hosta.a.com` | `*@*.b.com` | `Block` |
| ***Rule 2** | `*@*.a.com` | `*@hostb.b.com` | `Accept` |
| **Rule 4** | `*@*.a.com` | `*@*.b.com` | `Accept` |

- Rule 2 is not needed because in the following delivery scenario

```
*@<any host but hosta>.a.com  —>  *@hostb.b.com
```

the delivery is not matched by Rule 1 and is covered by Rule 4.

- Rule 3 is needed because Rule 1 applies to the scenario

```
*@hosta.a.com  —>  *@hostb.b.com
```

Hence the conflict can be resolved by using the rules 1, 3, and 4. Rule 4 may not be needed if it is coverd by a more generic rule.

# IMTA Location Relative to Public Internet



If the IMTA is directly connected to the public internet (such as on a firewall system), it delivers outbound mail by using the domain part (right-hand side) of the envelope recipient in the DNS and routes accordingly. Conversely, if the IMTA is not connected to the public internet, outbound mail for external addresses has to be forwarded to a smart host—an SMTP host which can resolve addresses that the current IMTA cannot resolve.

This section describes how to specify the position of the IMTA relative to the public internet, and how to specify a fully qualified smart host name if the IMTA is not directly connected to the internet. The routing configuration will differ depending on whether the IMTA is or is not connected to the internet. Depending on the position you select, the Admin Server will modify the IMTA rewrite rules to reflect that position.

---

**Note –** Use this option only when the IMTA location relative to the public internet changes. If the IMTA is connected to the Internet, but you want it to forward outbound mail to a dedicated outbound system, create an additional SMTP router channel to forward mail to this machine, then edit the configuration file `imta.cnf` to make the "." rule point to the newly created channel. See *SIMS Reference Guide* for more details.

---

## ▼ To Configure IMTA Position Relative to the Internet

| AdminConsole>IMTA>Position Vs. Firewall |
|---|

1. **From the Sections list of the IMTA property book, click on the Position Vs. Firewall.**

   The Position Vs Firewall section appears as shown in FIGURE 4-7.

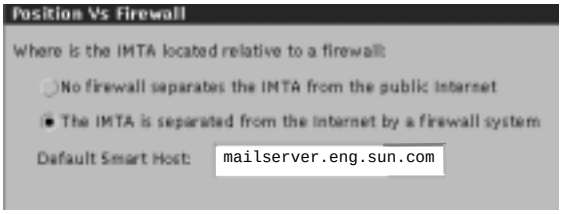



**FIGURE 4-7** Position Versus Firewall Section

2. **Determine whether the IMTA is connected directly to the public internet.**

   If the IMTA is connected directly to the public internet, select *No firewall separates the MTA from the public internet*. If the IMTA is not directly connected to the public internet, click *The IMTA is separated from the Internet by a firewall system*.

3. **If you indicated that a firewall separates the IMTA from the Internet, then specify the smart host name.**

   It must be a fully-qualified name. The syntax is `mailhost.domain`. Enter the name using ASCII characters. The characters are case-sensitive.

4. **Click the Apply button.**

# Routability Scope



By default, the IMTA is expected to be able to resolve an address of the form `user@xzy.com`, where `xzy.com` is the mail domain name. To resolve addresses, the IMTA constructs an alias cache containing *all* users in the domain `xzy.com` via alias synchronization.

It may be useful to change the routability scope of the alias cache in the following cases:

- When the directory is not populated with the entire domain.
- To limit the size of the alias cache
- To enforce routing policies. For example, if all mail going outside of the domain must be forwarded by a specific set of IMTAs.

The routability scope is the group of addresses to which the IMTA can route directly (send directly to the user's delivery mail store) or to which it can deliver locally.

This section explains how to set the routability scope to one of following:

- Nobody – Indicates that the mail server does not support a user community. This setup is typical if your mail server is a backbone IMTA that routes messages between domains. It does not know of each mail user, but uses the host or domain specifications to forward the message to the appropriate mail server for delivery. For example, if a message is sent to *user*@eng.alpha.com, the IMTA knows to forward this message to mailhost.eng.alpha.com. Similarly, it can forward a message addressed to *user*@qa.eng.alpha.com to mailhost.qa.eng.alpha.com.
- Local system users only – The IMTA can deliver messages to local users only. The IMTA cannot deliver to non-local users. If a message arrives that is not addressed to a local user and the To: envelope address is not canonical and fully qualified (that is, it does not specify the address's information as *user@host.domain*), the IMTA forwards it to a specified smart host. The smart host is more likely to be able to forward the message to the recipient's mail server.
- Mail server domains – The IMTA can route messages within its internet domain or to a specified set of domains. The mail server can forward a message to the recipient's mail server if the recipient belongs to one of the specified domains. The advantage of specifying this routability scope is that it reduces network hops by sending messages directly to the appropriate mailserver, rather than sending them to a smart host. A disadvantage may be the requirement for the directory to contain one entry for each valid mail user in the domains.
- Entire mail network – The IMTA can route messages to local users and to users that it knows about through replicated directory information that exists on this particular mail server.

Modifying the routability scope modifies the way the IMTA persistent alias cache is created and modifies the IMTA rewrite rules. If the routability scope is set to `domains`, you must specify which domain(s) in "To Configure Mail Server Domains" on page 116. Generally, it is recommended that you list all domains shared by users within the routability scope.

## ▼ To Configure Routability Scope

AdminConsole>IMTA>Routability Scope

1. **From the Sections list of the IMTA property book, click Routability Scope.**

   The Routability Scope section appears as shown in FIGURE 4-8.

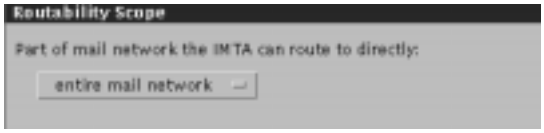



**FIGURE 4-8** Routability Scope Section

2. **Select the portion of the mail network to which the IMTA can route using the pull-down menu.**
3. **If you selected the Mail Server Domains option, then make certain that mail server domains are configured, add domains to the Mail Server Domain list, or modify the existing list.**

   To check on and configure the mail server domain list, refer to "To Configure Mail Server Domains" on page 116.

4. **Click the Apply button.**

## ▼ To Configure Mail Server Domains

If you specified the Mail Server Domain option in the Routability Scope section of the IMTA property book as described in "Routability Scope" on page 114, you must configure a list of mail server domains to which your IMTA can route messages. Your list can include as many Internet domains as desired.

AdminConsole>IMTA>Mail Server Domains

1. **From the Sections list of the IMTA property book, click Mail Server Domains.**

   The Mail Server Domains section appears in FIGURE 4-9.


**FIGURE 4-9** Mail Server Domain Section

2. **To add a mail server domain to the list, enter the name in the Mail Server Domains field.**

   For example, if you want your IMTA to route messages to the `sales.bravo.com` domain, then enter the following:

   `sales.bravo.com`

3. **Click the add button.**

   The mail server domain name appears in the list.

4. **To delete a mail server domain from the list, you can either click on the domain name in the list to highlight it then click the delete button, or enter the domain in the Mail Server Domains field and click the delete button.**

5. **Click the Apply button.**

# Channels

The Channels section enables you to view the status of the IMTA channels. For more information, refer to “Monitoring Channel Status” on page 101. In addition, you can modify the properties of specific channels by double-clicking on the desired channel and bringing up the property book associated with that channels.

# Configuring Channels



This section describes how to configure channel attributes. The IMTA supplied with the SIMS Enterprise Edition includes the following configurable channels:

- Internet SMTP channel
- Intranet SMTP channel
- Router SMTP channel
- Sun Message Store channel
- `/var/mail` channel
- Pipe channel

Note that the number of SMTP channels will depend on the mail server's position versus the internet. With the Sun Messaging Connectivity Package installed, you also get these channels:

- cc:Mail channel
- PROFS channel
- Microsoft Mail channel

This section only describes configuring non-connectivity channels. See the *Sun Messaging Connectivity Services Channel Guides* for connectivity channel information.

**Note –** SIMS does not support the configuration of the UNIX to UNIX Copy Program (UUCP) channel using the Admin Console. You can configure the UUCP channel by editing the `imta.cnf` file. For more information on `imta.cnf`, refer to the *SIMS Reference Manual*.

TABLE 4-2 summarizes the configurable attributes of these channels, specifically which channels each attribute applies to.

**TABLE 4-2** Configuring Non-Connectivity Channels

| Configurable Aspect | Channel Applies To | Description |
|---|---|---|
| Channel description | Applies to all channels | You can generate a description of a channel for administrative purposes only. |
| Router | Applies only to SMTP router channels | In the event that an IMTA cannot resolve a particular message address, you must configure a host to which the IMTA can route the message. |
| Character set labels | All channels | Determines the label for 7-bit character sets and for 8-bit character sets to be used in plain text messages. |
| Message Limitation | Some attributes apply to SMTP channels only | Determines how the channels handle large messages and messages with many recipients. |
| Delivery Status Notification | Applies to all channels | Determines how a channel handles the messages that warn of or return undelivered mail. |
| Report Problems to Postmaster | Applies to all channels | Determines how a channel handles the sending of warning messages to the postmaster. |
| Diagnostics Output | Applies to all channels | Determines whether a channel produces diagnostic output for its master program, slave program, or both. |
| Performance Tuning | Applies to all channels | Determines how the IMTA delivers messages and defers the processing of messages, thereby tuning its performance. |
| Logging | Applies to all channels | When selected, provides logging information to the global log. |
| Multiple Internet Mail Extensions (MIME) Fragmentation | Applies to Sun Internet Mail Server - Enterprise Edition only. Further applies to Sun Message Store, `/var/mail`, and pipe channels only. | You can enable a channel to reassemble message fragments into one message upon receipt. |
| Rewrite rules | Applies to all channels | You can add a new rewrite rule to an existing or newly created channel, or delete or modify an existing rewrite rule. |

## ▼ To Create a Channel

If you have the Sun Internet Mail Server - Enterprise Edition, you can create additional channels. You can create channels of the type *SMTP explicit router* only through the Admin Console. No limitations exist for the overall number of channels that you can create and the number of a specific type of channel that you can create.

| AdminConsole>IMTA>Create pulldown>Channel |
|---|

1. **In the Admin Console home page, click IMTA.**
2. **In the IMTA property book, choose Channel from the Create menu.**

   The Create Channel dialog box appears.
3. **Enter a name for your new channel.**

   Valid entries include ASCII characters. A maximum of 40 characters is accepted.
4. **Click the Ok button.**

   The New Channel Property Book appears. Note that this is similar to the Existing Channel Property Book described in “To Access Channels Property Book” on page 121.

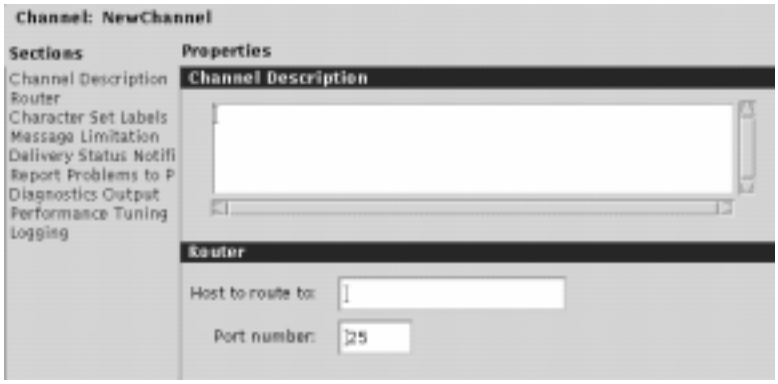



**FIGURE 4-10** New Channel Property Book

5. **Fill in the various sections and press OK**

   Only two sections are mandatory at this time: Router and Message Limitation. *Enter the Host to route to:* field and change the *Max. no. of recipients per msg* field from 0 to whatever maximum number of recipients you wish to allow a single message to be sent by the IMTA. If you don’t change this value, the channel will not work. The remaining fields can be entered at a later time. These fields are described in the remaining sections of this chapter.

6. **After the channel is created, you must configure rewrite rules for the new channel to process messages properly.**

   Refer to “Configuring Channels” on page 118 and “Rewrite Rules” on page 132 for more information on rewrite rules and configuring the channel.

## ▼ To Delete A Channel

If you have the Sun Internet Mail Server - Enterprise Edition, you can delete the router SMTP channel (explicit route) that you created.

| AdminConsole>IMTA>Channels>Selected pulldown>Delete Channel |
|---|

1. **In the IMTA property book, click Channels in the Sections list.**

   The Channels section appears. This section displays a list of installed channels.

2. **Click the channel that you want to delete.**

   The channel you want to delete is highlighted.

3. **Choose Delete Channel from the Selected menu.**

   Confirm that you want to delete the channel.

4. **Click the Yes button.**

   The deleted channel name, type, and status are removed from the channel list.

## ▼ To Access Channels Property Book

| AdminConsole>IMTA>Channels |
|---|

1. **In the Admin Console home page, click the IMTA icon.**

2. **In the Sections list, click Channels.**

   The Channels section appears. This section displays a list of channels.

3. **Either click the desired channel in the list and then select Properties from the Selected menu, or double-click the desired channel in the list.**

   The channel property book appears, as shown in FIGURE 4-11.


**FIGURE 4-11** Sample Channel Property Book

## ▼ To Configure A Channel Description

By default, the IMTA generates a channel description. You can update this description with any desired notes or details. This description is for administrative purposes only and does not affect the configuration of the channel.

| AdminConsole>IMTA>Channels>desired channel description>Selected Menu>Properties |
|---|

1. **Click Channel Description from the Sections list of the channel property book.**

   See “To Access Channels Property Book” on page 121 for more information. The Channel Description section appears, as shown in FIGURE 4-12.

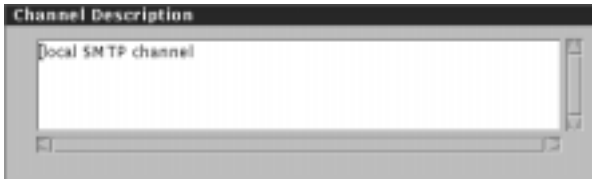



**FIGURE 4-12** Channel Description Section

2. **Update the text description of the channel with up to 256 characters.**
3. **Click the Apply button.**

## ▼ To Configure a Router Host

---

**Note –** This section applies to SMTP router channels only. See “SMTP Channels” on page 10.

---

In order to route messages to a particular domain, the IMTA may rely on the DNS and deliver mail through the SMTP intranet / SMTP internet channels. An alternative is to specify the domain host to which to route mail. The messages are then delivered through an SMTP router channel

Note that if the IMTA is not connected to the internet, mail to all external domains is forwarded to a smart host by the default SMTP router channel. The smart host is the host to route to for the default SMTP router channel.

| AdminConsole>IMTA>Channels>selected channel>Selected Menu>Properties>Router |
|---|

1. **From the Sections list of the Channel property book, click Router.**

   To find out how to access the channel property book, see “To Access Channels Property Book” on page 121. The Router section appears as shown in FIGURE 4-13.

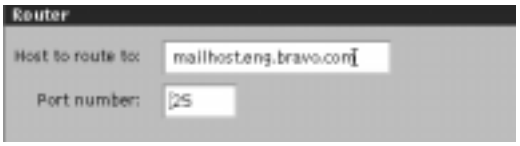



**FIGURE 4-13** Router Section

2. **Type the host name of the IMTA that functions as a router using the following syntax:**

   *hostname.domain*

   A sample host name is mailhost.eng.bravo.com.

3. **Type the port number through which the routed messages should enter.**

4. **Click the Apply button.**

   You are prompted to restart the IMTA.

5. **Click the Yes button.**

## ▼ To Configure Character Set Labels

The MIME standard provides a means of labeling or naming the character set used in a plain text message. The character set labels are inserted in the Content-type header of a message, indicating what type of characters are used in the message.

For more information on MIME, refer to “Message Content / MIME” on page 8.

| AdminConsole>IMTA>Channels>selected channel>Selected Menu>Properties |
|---|

1. **In the Sections list of the Channel property book, click Character Set Labels.**

   To find out how to access the channel property book, see “To Access Channels Property Book” on page 121.

   The Character Set Labels section appears as shown in FIGURE 4-14.


**FIGURE 4-14** Select Character Set Section

2. **Use the menus to specify a label for the 7-bit character set and for the 8-bit character set.**

3. **Click the Apply button.**

   A dialog box appears prompting you to restart the IMTA. click Yes.

# Message Limitation



Some email systems and IMTAs encounter problems when handling large messages. You can control how the channels handle large messages (measured in bytes and number of lines) for the SMTP intranet/internet channels, SMTP explicit route channels, and local or message store channels (for other channels, this feature has no effect). You can specify that messages be limited by size, number of recipients, or that they be fragmented if they exceed a certain size.

## ▼ To Configure Message Limitation

AdminConsole>IMTA>Channels>selected channel>Selected Menu>Properties>Message Limitation

1. **In the Sections list of the Channel property book (see "To Access Channels Property Book" on page 121), click Message Limitation.**

   The Message Limitation section appears as shown in FIGURE 4-15.

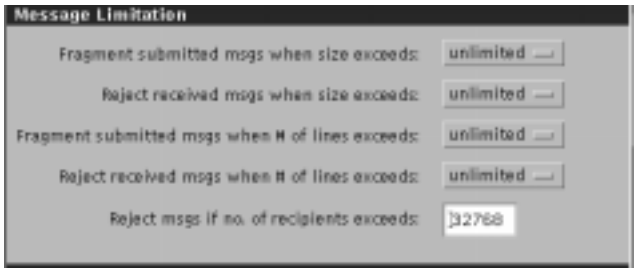



**FIGURE 4-15** Message Limitation Section

   To set local user to local user mail message limitations, configure the *local* (Solaris Message Store) or *message Store* (Sun Message Store) channel depending on the recipient's channel. Note that the *local* and *message Store* channels do not support either of the *Fragmented submitted msgs* parameters or the *Max. no. of recipients per msg* parameter.

2. **(Optional) Set limit on message size.**
3. **(Optional) Set limit on the number of lines in incoming messages.**

You can specify a limit at which a message should be fragmented into smaller messages and a limit at which a message should be deemed too large and rejected. The default is unlimited.

4. **Set limits on the maximum number of recipients per message by typing a value.**

   Valid entries include 0 to 32,768.By default, a channel handles up to 32,268 recipients per message. If desired you can specify a limit at which a message is deemed to have too many recipients and is rejected.

5. **Click the Apply button**

   You are prompted to restart the IMTA. Click Yes.

## Delivery Status Notification



Occasionally, a channel may not be able to process a message (for example, the channel may go down, which is considered a transient failure, or a user is unknown, which is a permanent failure). For a permanent failure, the message is bounced and a notification is sent to the postmaster.

For a transient failure, by default, the channel will send up to three warning messages to the originator of the message at the following intervals: 1 day, 2 days, 4 days, and 7 days after the original message was sent. Each warning message will inform the originator that the original message is undeliverable, why the message is undeliverable, and how long delivery attempts will continue. By default, 12 days after the original message was sent, the original message will be returned to the originator. If the failure is corrected before the 12 days, the messages will be delivered. You can reconfigure the interval at which the warning messages are sent and the original message is returned.

### ▼ To Configure Delivery Status Notification

AdminConsole>IMTA>Channels>selected channel>Selected Menu>Properties>Delivery Status Notification

1. **In the Sections list of the Channel property book, click Delivery Status Notification.**

The Delivery Status Notification section appears as shown in FIGURE 4-16.

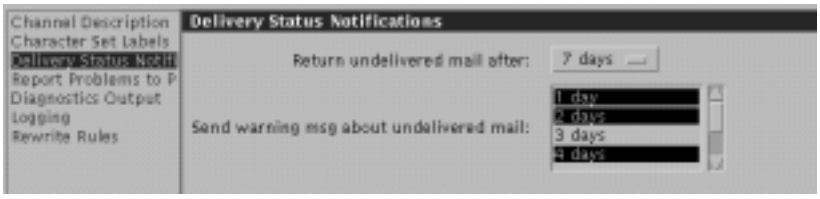



**FIGURE 4-16** Deliver Status Notification Section

2. **Configure the number of days after which undelivered mail should be returned.**
3. **Configure the interval and the days after which a message is sent that a warning message is sent.**

   You can specify up to four selections by clicking each desired number of days to highlight it.
4. **Click the Apply button.**

   You are prompted to restart the IMTA.Click Yes.

## ▼ To Configure Failed Delivery Reports to the Postmaster

By default, the local postmaster receives a copy of all warning messages for transient and permanent failures except those undelivered messages that do not have an originator address. You can reconfigure the channel to send a copy of all or no warning messages to the local postmaster. Although receiving a copy of each of the warning messages may help you monitor the state of the channel queue, you will have to weigh this benefit against the increase in traffic that the channel will need to handle. To configure what part of the message is sent to the postmaster, you can add additional channel keywords in `imta.cnf` (see the *SIMS Reference Manual*).

| AdminConsole>IMTA>Channels>selected channel>Selected Menu>Properties>Report Problems to Postmaster |
|---|

1. **In the Sections list of the Channel property book, click Report Problems to Postmaster.**

   To find out how to access the channel property book, see “To Access Channels Property Book” on page 121.

The Report Problems to Postmaster section appears as shown in FIGURE 4-17.

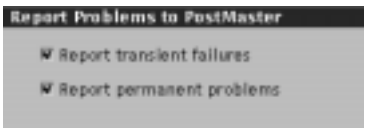



**FIGURE 4-17** Report Problems to Postmaster Section

2. **To send transient failure warning messages to the local postmaster, click Report transient failures.**
3. **To send warning messages of permanent failures to the local postmaster, click Report permanent problems.**
4. **Click the Apply button.**

   You are prompted to restart the IMTA. Click Yes.

## Diagnostics Output



By default, a channel does not produce diagnostics output for its master and slave programs. You can reconfigure the channel so that it produces diagnostics output for either its master program, its slave program, or both.

When enabled, diagnostic output is written to the log file associated with the channel program. For more information on the channel's master and slave programs, refer to "Channels" on page 9. For more information on diagnosis using log files refer to the *SIMS Reference Manual*.

### ▼ To Configure Diagnostics Output

AdminConsole>IMTA>Channels>selected channel>Selected Menu>Properties>Diagnostics Output

1. **In the Sections list of the Channel property book (see "To Access Channels Property Book" on page 121), click Diagnostics Output.**

The Diagnostics Output section appears as shown in FIGURE 4-18.

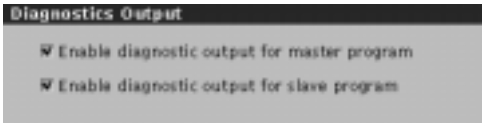



**FIGURE 4-18** Diagnostics Output Section

2. **Determine whether diagnostic output is generated.**
   - To generate diagnostic output for the channels master program, click the first check box.
   - To generate diagnostic output for the channel's slave program, click the second check box.
   - To enable diagnostic output for both master and slave programs, click the check boxes associated with each of these choices.
3. **Click the Apply button.**

   You are prompted to restart the IMTA. Click Yes. The debug output file is at `/var/opt/SUNWmail/imta/log/`

   The format of the channel debug file name is as follows:

   `channel_master.log_xxxxxx` - master debug file (xxxxx is a random string).
   `channel_slave.log_xxxxxx` - slave debug file (xxxxx is a random string).

## ▼ Performance Tuning

By default, the SMTP channel allows an unlimited number of recipients on a message without deferring processing. Too many recipient addresses can result in a delay of message processing. If the delay is long enough, network timeouts can occur and this in turn can lead to repeated message submission attempts and other problems. You can specify a limit of recipients for a single message at which message processing is deferred. When the specified number is exceeded, the message is enqueued and the remaining recipients are not verified on-line. Non-delivery receipts are generated for recipients later found undeliverable.

| AdminConsole>IMTA>Channels>double-click channel>Properties>Performance Tuning |
|---|

1. **In the Sections list of the Channel property book, click Performance Tuning.**

   To find out how to access the Channel property book, see "To Access Channels Property Book" on page 121.

   The Performance Tuning section appears as shown in FIGURE 4-19.


**FIGURE 4-19** Performance Tuning Section

2. **Specify a limit of recipients for a single message using the menu.**
3. **Click the Apply button.**

## Message Logging



By default, a channel logs in each message in `/var/opt/SUNWmail/imta/log/mail.log_current`. You can reconfigure the channel so that it logs in each message as it enters and is removed from the queue.

A log entry consists of the following fields:

- Date and time that entry was made
- Name of the source channel (channel that message originated from)
- Name of the destination channel (channel that message needs to be delivered to)
- Type of entry:
  - E = message was entered into the channel queue
  - D = message was removed from the channel queue
  - Q = an unsuccessful attempt was made to remove the message from the channel queue
- Size of the message in kilobytes
- Address in the From: header
- Address in the To: header

For more information refer to “Tracking Messages” on page 259 and the *SIMS Reference Manual*.

### ▼ To Configure Message Logging

AdminConsole>IMTA>Channels>double-click channel>Properties>Logging

1. **From the Sections list of the channel property book, click Logging.**

   To find out how to access the Channel property book, see “To Access Channels Property Book” on page 121.

   The Logging section appears as shown in FIGURE 4-20.


**FIGURE 4-20** Logging Section

2. **To enable the logging of each message as it enters and is removed from the channel queue, click the check box.**

3. **Click the Apply button.**

   You are prompted to restart the IMTA. Click Yes.

# Reassembling MIME Messages



---

**Note –** This section applies to the Sun Internet Mail Server - Enterprise Edition only. It further applies to the UNIX to UNIX Copy Program (UUCP) channel, Sun Message Store channel, `/var/mail` channel, and pipe channel only.

---

Occasionally a large message must traverse email systems that impose message size limitations. MIME allows the breaking up of a large message into smaller messages, a process known as *fragmentation*. A Message/Partial Content-Type header field that appears in each of the smaller messages or *fragments* contains information that helps reassemble the fragments into one message, a process known as *defragmentation*.

By default, the Sun Message Store channel, `/var/mail` channel, and pipe channel do not defragment a message. You can configure each of these channels to reassemble a fragmented message upon receipt.

For complete information on message fragmentation and defragmentation, and the Message/Partial Content-Type header field, refer to RFC 1521.

## ▼ To Enable Reassembly of Message Fragments

AdminConsole>IMTA>Channels>double-click channel>Properties>MIME Fragmentation

1. **From the Sections list of the Channel property book, click MIME Fragmentation.**

   The MIME Fragmentation section appears as shown in FIGURE 4-21.

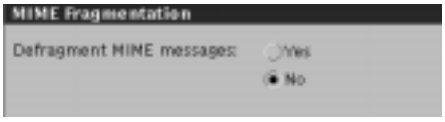



**FIGURE 4-21** MIME Fragmentation Section

2. **Determine whether message fragments are reassembled.**
   - To enable the reassembly of message fragments, click the Yes.
   - To disable the reassembly of messages, click No.
3. **Click the Apply button.**

   You are prompted to restart the IMTA. Click Yes.

# Rewrite Rules



The Admin Console enables you to add a rewrite rule to an existing or newly created channel. It also enables you to delete or modify an existing rewrite rule associated with an existing channel.

Before adding or modifying a rewrite rule for a particular channel, read the associated conceptual information. For complete conceptual information, refer to “Rewrite Rules” on page 13.

When adding or modifying a rewrite rule, you may need to configure the following elements:

- Pattern
- Domain template
- Element to which rule applies
- Address direction

- Order of rewrite rules

A pattern is a string composed of ASCII text, which may include a wildcard characters that can potentially match the host / domain specification of an incoming email address. The host / domain specification is the portion that is to the right of the at (@) sign; for example, in the address john.smith@eng.acme.com, the host / domain specification of the address is eng.acme.com. The wildcard character that you can specify is an asterisk (*). An example of a pattern entry is

*.acme.com

A domain template defines how the host / domain specification of the address is rewritten. A template can be composed of one or both of the following elements:

- A full static host / domain specification, for example, corp.acme.com, or a portion of a host / domain specification (a portion of the address tokens), for example, .com.
- A single field substitution string that dynamically rewrites one address token of the host / domain specification represented by a wildcard character (*). The address token to be rewritten can be the portion of the address that did not match the rewrite rule pattern or the portion that matched the wildcard character. The rewriting of a host / domain specification is based on the contents of the specification itself. The template can include multiple field substitution strings.

The syntax of the field substitution string is as follows:

$&*n*

where *n* is an integer from 0 to infinity. *n* represents the unmatched or wildcard address token that is to be rewritten. From left to right, the leftmost address token is represented by the integer 0; the second from the left is represented by the integer 1, and so on. An example of a template entry is

$&0.$&1.com

For complete information, including a full example of the domain template, refer to “Rewriting the Host / Domain Specification” on page 15.

Element to which rule applies determines whether the rewrite rule applies to the address that appears on the envelope only, the message header only, or both.

Address direction determines whether the rewrite rule applies to a forward address (To:, CC:, and BCC: headers) or a backward address (From: or Reply-to: headers).

You can reconfigure the order of multiple rewrite rules containing the same pattern but different domain templates by moving one of the rules up or down in the Admin Console’s rewrite rule display. Changing the order of these types of rewrite rules causes the channel to scan these rules in the reconfigured order when an incoming message appears. For more information on the order of rewrite rules, refer to “Matching Host / Domain Specification With A Rewrite Rule Pattern” on page 13.

## ▼ To Add, Delete, or Modify A Rewrite Rule

AdminConsole>IMTA>Channels>double-click channel>Properties>Rewrite Rules

1. **In the Sections list of the Channel property book (see “To Access Channels Property Book” on page 121, click Rewrite Rules.**

   FIGURE 4-22 shows the Rewrite Rules section.

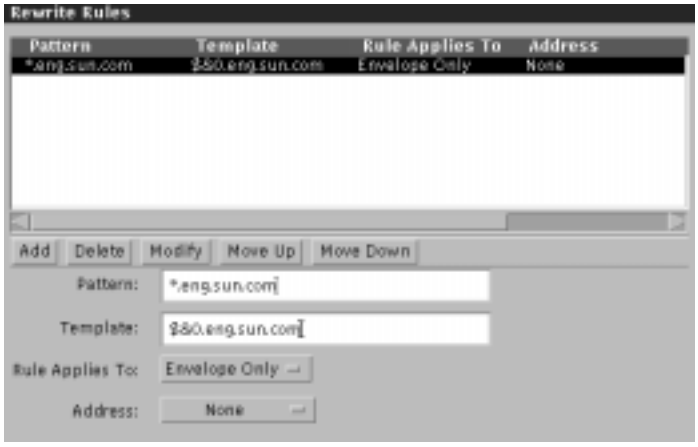



**FIGURE 4-22** Rewrite Rules Section

   The Rewrite Rules section is divided into two sections: a display of already existing rewrite rules in the top part of the display and various configurable fields in the bottom part of the display.

2. **If you want to delete, modify, move up, or move down an existing rule in the display, click on the rule in the top part of the display to highlight it.**

   The bottom part of the display shows the existing entries for pattern and template as well as the settings for protocol and direction.

3. **Take one of the following actions with the existing rewrite rule:**

   a. **Delete the rule.**

      Click on the Delete button.

   b. **Modify the rule.**

      Delete and reenter the Pattern and Template entries and use the pull-down menus to reselect the value for Rules Applies to and Address. Click the Modify button.

   c. **Move the rule up or down in the list of rules.**

      Click either the Move Up or Move Down button.

4. **If desired, add a new rewrite rule using the following steps:**

   a. **Specify a pattern.**

      Enter a pattern composed of ASCII text and the wildcard character of an asterisk (*). An example of a pattern entry is

      *.acme.com

   b. **Specify a template.**

      Enter either a full host/domain address, for example, corp.acme.com, or a partial host/domain address set off by decimals, for example, .com.

      If you enter a partial host/domain address, you also need to specify the appropriate number of field substitution strings. The syntax of the field substitution string is as follows:

      $&*n*

      where *n* is an integer from 0 to infinity. The integer represents the address token that is to be rewritten. From left to right, the leftmost address token is represented by the integer 0; the second from the left is represented by the integer 1, and so on. For example, if an incoming address matches the following rewrite rule pattern:

      *.acme.com

      then the domain template will consist of the following:

      $&0.acme.com

   c. **Specify whether you want the rewrite rule to apply to envelope only, message header only, or both using the pull-down menu.**

   d. **Specify whether you want the rewrite rule to apply to a forward address only, a backward address only using the pull-down menu.**

   e. **Click the Add button.**

5. **Click the Apply button.**

   You are prompted to restart the IMTA. Click Yes.

# Monitoring Channel Queues



You can monitor accumulated message traffic statistics for each IMTA channel queue (includes user-created channels). At the same time, you can compare statistics accumulated for one channel queue to statistics accumulated for the IMTA.

The queue monitor accumulates the following statistics:

- Received messages – Number of messages that have entered the channel queue
- Submitted messages – Number of messages that have exited from one channel queue and entered another channel queue
- Delivered messages – Number of messages that have exited the channel queue
- Stored messages – Number of messages currently stored in the channel queue
- Received volume – Volume of messages that have entered the channel queue measured in kilobytes
- Submitted volume – Volume of messages that have exited from the channel queue and entered another channel queue measured in kilobytes
- Delivered volume – Volume of messages that have exited from the channel queue measured in kilobytes
- Stored volume – Volume of messages currently stored in the channel queue measured in kilobytes

## ▼ To Monitor the IMTA Channel Queues

AdminConsole>IMTA>Channels>select channel>Selected pulldown>Monitor Queue

1. **From the Admin Console home page, click on the IMTA icon.**
2. **Click on Channels from the Sections list.**
3. **Click the channel whose queue you want to monitor.**

   The channel name, type, and status are highlighted.
4. **Choose Monitor Queue from the Selected pull-down menu.**

   The Monitoring Queue page for the selected channel appears (FIGURE 4-23).

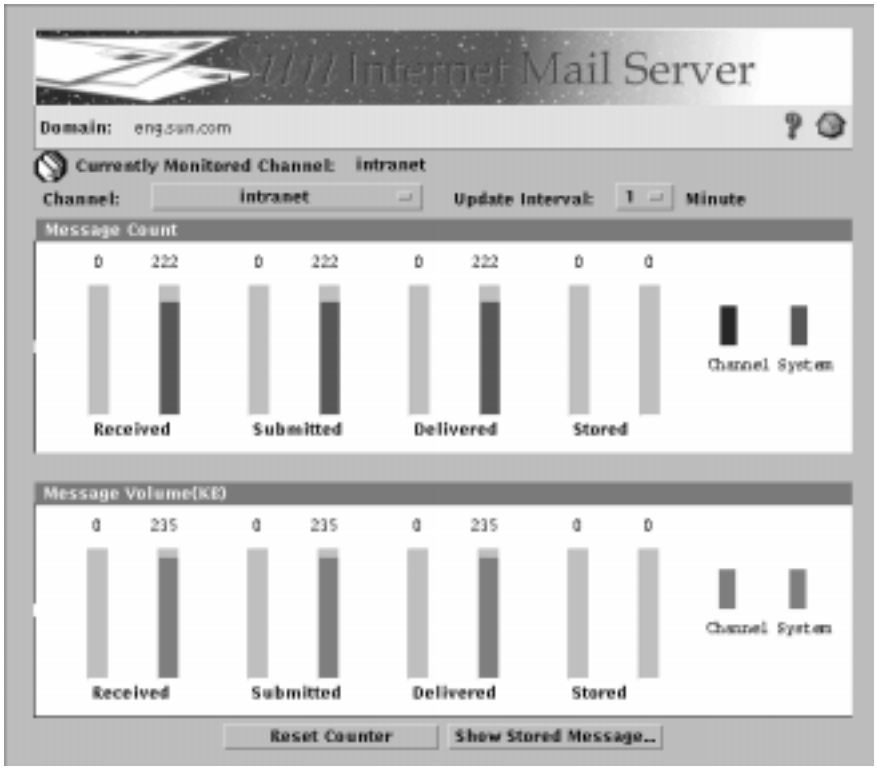



**FIGURE 4-23** Internet Message Transfer Agent Channel Queue Monitor

The queue monitor incorporates the following color codes:

- Red – Message count per channel
- Blue – Message count per system
- Red – Message volume per channel
- Blue – Message volume per system

5. **(Optional) Change the time interval in which data is polled.**

   Click on the Update Interval option pull-down menu and select the desired option.

6. **(Optional) Reset the counter by clicking the Reset Counter button.**

   Note that after resetting the counter, messages in transit will not be counted; therefore, the statistics provided in the Message Count portion of the page will not be accurate. In this situation, rely on the statistics presented in Message Volume portion of the page.

7. **To view statistics for other channel queues, click on the Channel option pull-down menu and select the desired channel.**

   This list of channels includes all installed and user-created channels.

## Viewing Enqueued Messages



You can also view a listing of the messages currently in the channel queue. The messages are stored for various reasons; for example, the mail server may be currently unavailable and the message delivery will be retried later. The type of information you can view about the stored messages includes the following:

- Message ID
- Message originator
- Date/time that message was originated
- Message size
- Contents of message itself

Once you find the specific message, you can view the contents of the message, save it, or delete it from the queue.

### ▼ To View Messages Stored In the IMTA Channel Queues

AdminConsole>IMTA>Channels>selected channel>Selected Menu>Monitor Queue>Show Stored Message

1. **From the Admin Console home page, click on the IMTA icon.**

   The IMTA property book appears.

2. **From the Sections list, click on Channels.**

   The Channels section appears.

3. **Click on the channel whose queue you want to monitor.**

   The channel name, type, and status is highlighted.

4. **Click on the Selected pull-down menu and select Monitor Queue.**

   The Monitoring Queue page with statistics for the selected channel appears.

5. **From the Monitoring Queue page, click on the Show Stored Message button.**

   The Stored Messages dialog appears as shown in FIGURE 4-24.


**FIGURE 4-24** Stored Queue Monitoring

   A listing of stored messages appears in the top half of the dialog.

6. **(Optional) View the contents of the message by double-clicking on the message in the list.**

   The contents of the message displays in the bottom half of the page.

7. **(Optional) Save or delete a message from the channel queue.**

   Click the message and then click the Save or Delete button as appropriate.

8. **Close the Stored Messages dialog by clicking the Close button.**

# Connectivity Channels

For complete information on configuring the Lotus cc:mail, Microsoft Mail, and IBM PROFS channels, refer to the *Sun Messaging Connectivity Services Channel Guides*.

CHAPTER 5

# Message Store Administration

This chapter describes step-by-step instructions for changing the Sun Message Store characteristics of SIMS. To start, bring up the Sun Message Store property book pages.

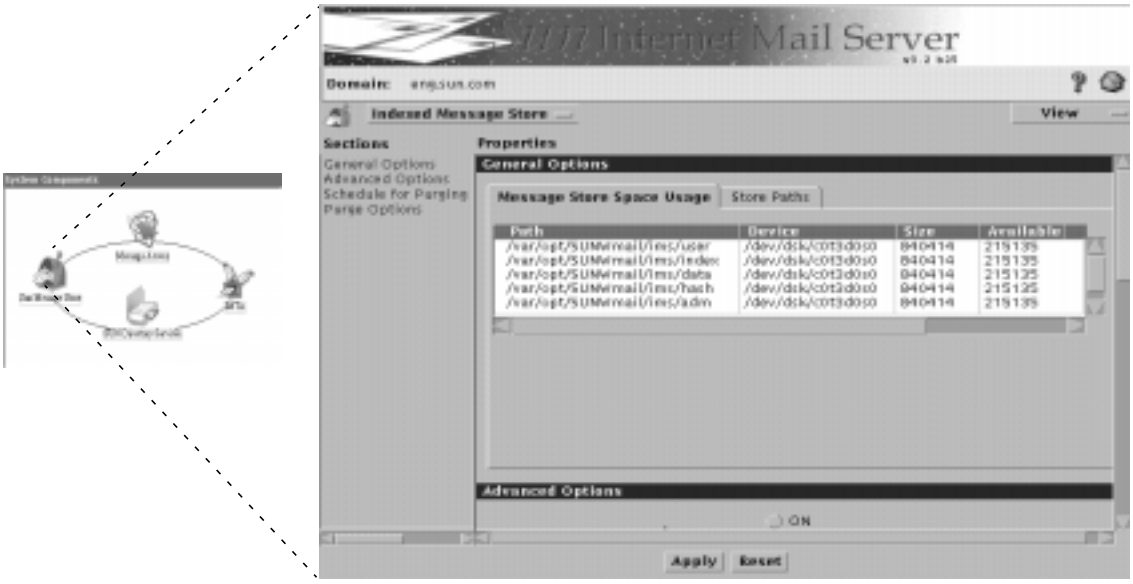



**FIGURE 5-1** Sun Message Store Property Book

# Sun Message Store Topics and Tasks

**TABLE 5-1** Message Store Topics and Tasks

| Topic/Task | Description | Page |
|---|---|---|
| Sun Message Store Configuration Back Up and Restore | Backup and restore | 143 |
| Monitoring the Sun Message Store | Describes how to view the SIMS message store path, space usage, and user space quotas. | 144 |
| Message Store Quota Enforcement | Describes the SIMS user message store quota system. | 148 |
| Configuring Advanced Options | Describes the following:<br>- User quota enforcement / default message quota<br>- Mail server client type<br>- Directory context for users<br>- Maximum connections permitted<br>- Percentage space left warning threshold<br>- `/var/mail` support<br>- Size by which to increase Sun Message Store<br>- Message purge options and scheduling | 152 |
| Message Purge | Discusses how to configure message purge options and purge schedules. | 158 |
| Message Access Protocol Connections | How to view and monitor all user connections to SIMS, as well as start and stop client access to the message store | 161 |
| Sun Message Store Maintenance | This section is in Chapter 7, “Maintenance.” It describes the following Sun Message Store maintenance procedures:<br>- Recommended Maintenance Schedule<br>- Message Purge<br>- Message Store Backup and Restore<br>- Folder Check<br>- Importing /var/mail Users<br>- Deleting Old Messages<br>- Delete User<br>- Sun Message Store Configuration Back Up and Restore | 233 |
| Message Access Protocols Error Messages | Error messages and proper responses. Appendix E, “Error Messages.” | 363 |

# Sun Message Store Configuration Back Up and Restore

After SIMS is installed and you have responded to prompts for various information during the initial setup of the server, the server saves or *backs up* the Sun Message Store configuration. This configuration version is known as the *default configuration*.

Subsequently, the Admin Console enables you to back up your Sun Message Store configuration at any time. The Admin Console enables you to save up to two versions of the Sun Message Store configuration. The latest working configuration is known as the *current configuration*. The previously saved working configuration is known as the *backup configuration*.

For example, imagine that you reconfigure certain aspects of the Sun Message Store using the Admin Console. You decide to back up this particular configuration on May 1. Because this configuration is the latest working configuration, it is considered the current configuration. On June 1, you reconfigure more aspects of the Sun Message Store and perform another backup. The May 1 configuration becomes the backup configuration and the June 1 configuration is considered the current configuration. On August 1, you reconfigure even more aspects of the Sun Message Store and perform another backup. Because the Sun Message Store can save only two configuration versions, the May 1 configuration is not saved. The June 1 configuration becomes the backup configuration and the August 1 is considered the current configuration.

If for some reason you wish to use a previous Sun Message Store configuration version, you can restore one of the following configuration versions:

- Default configuration.
- Backup configuration (provided that this version exists)

## ▼ To Back Up and Restore the Sun Message Store Configuration

| AdminConsole>Sun Message Store Pulldown>Backup config |
|---|

1. **From the Sun Message Store property book, click on the Sun Message Store pull-down menu and select Backup Config.**

   The Sun Message Store makes a backup of the current configuration.

2. **If desired, restore either the default configuration or the backup configuration, if this version exists.**

**a. To restore the default configuration, click on the Sun Message Store pull-down menu and select Use Default Configuration.**

The Sun Message Store restores the default configuration.

**b. To restore the backup configuration, click on the Sun Message Store pull-down menu and select Use Backup Configuration.**

The Sun Message Store restores the backup configuration.

# Monitoring the Sun Message Store



## ▼ To Monitor Mail Store Space Usage and Settings

You can monitor the following Sun Message Store parameters:

- Current size of the directories that store user folders, indexes, messages and attachments, message hash, Sun Message Store log files, and shared or group folders
- Amount of remaining hard disk space available for each directory listed.

| AdminConsole>Sun Message Store>General Options>Message Store Space Usage |
|---|

1. **From the Admin Console home page, click the Sun Message Store icon.**
2. **Click General Options in the Sections list.**

   This section is divided into subsections for the space usage and store paths.

3. **Click the Message Store Space Usage tab.**

   The Message Store Space Usage subsection appears as shown in FIGURE 5-2.


**FIGURE 5-2** Message Store Space Usage Subsection

- Path - Directory that stores user folders, indexes, messages and attachments, message hash, Sun Message Store log files, and shared or group folders
- Device - Hard disk partition on which the directories reside
- Size - Current size of each directory
- Available - Amount of remaining hard disk space currently available for each directory

## ▼ To View Sun Message Store Paths

During SIMS installation, you provided a pathname for the directories that store the messages and attachments, indexes, user folders, shared or group folders, message hash, and log file, or you decided to use the default pathnames. Values for owner, host, and number of days were also assigned during installation. These can be viewed with the following procedure. Of the values displayed in this section, you can reconfigure the number of days to initialize the Sun Message Store only. For more information, refer to “Sun Message Store Increase” on page 155.

| AdminConsole>Sun Message Store>General Options> Store Paths |
|---|

1. **From the Admin Console home page, click the Sun Message Store icon.**
2. **Click General Options in the Sections list.**

   This section is divided into subsections for the space usage and store paths.

3. **Click the Store Paths tab (**FIGURE 5-3**).**


**FIGURE 5-3** Store Paths Subsection

- SIMS Owner - Owner (of Sun Message Store files).
- SIMS Host - Name of host on which the Sun Message Store is installed.
- FileSystem - This can either be *safe* or *unsafe*. A safe file system performs logging such that if a system crashes it is possible to rollback the data to a pre-crash state and restore all data. An example of a safe file system is VXFS. An unsafe file system does not perform logging. If the system crashes, the state cannot be recreated and some data may be lost. You must also perform an `imcheck` before activating message access to these files.
- User folders - Contains user's email folders.
- Shared or group folders - Contains folders for shared and group folders.
- Message Databases - Contains messages and attachments.
- Message Indices - Message Index files.
- Message hash - Hashing files.
- IMS Log - Sun Message Store log files
- SIMS initialization duration in days - Number of days to initialize the Sun Message Store

## ▼ To Monitor User Quotas

You can configure a maximum amount of hard disk space or *quota* for each Sun Message Store user. See "Message Store Quota Enforcement" on page 148. Messages that arrive for a user whose quota has been met or exceeded will be returned to the sender. The returned message will indicate only a failure to deliver.

**Note –** Administrators can send a warning message to users whose space usage is approaching their message storage quota by using the `imquotacheck` command. Refer to the *SIMS Reference Manual* for further details.

Monitor how much space each user is using as compared to their quota with the following procedure:

AdminConsole>Sun Message Store>View pulldown>User Quota

1. **From the Admin Console home page, click on the Sun Message Store icon.**
2. **Click on the View pull-down menu and select User Quota (**FIGURE 5-4**).**

   Note that this information is not more than two hours old. If you want a realtime snapshot of the data run `imquotacheck -v -n` (see the SIMS Reference Manual).

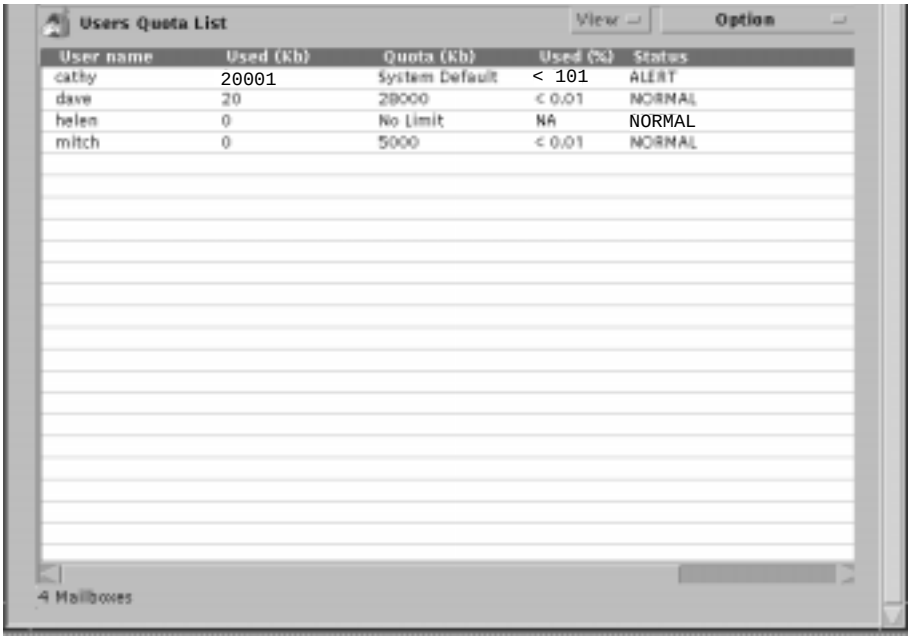



**FIGURE 5-4** User Quota Dialog

The User Quota page displays the following elements:

- User name - Name of each Sun Message Store user sorted alphabetically. Use the scroll bars to view the entire list.
- Used - Amount of hard disk space used in kilobytes.

- Quota - Maximum amount of hard disk space that can be used in kilobytes.
- Used - Amount of hard disk space used in percentage.
- Status - The Sun Message Store polls for a status every 10 minutes or if you exit then immediately reaccess the User Quota dialog, then it polls immediately. Possible states include NORMAL and ALERT. NORMAL indicates that a user is using below the configured quota. ALERT indicates that the user has exceeded the quota.

3. **If desired, you can save the information displayed in the User Quota page to a file by clicking the Option pull-down menu and selecting Save Quota List.**

   The information is at:

   `/var/opt/SUNWmail/ims/userquota.csv`

   You can move the contents of this file into a database or spreadsheet application. The contents of this file are structured in the following way:

   `username, space used, allocated quota {0 | 1}`

   where

   `allocated` = -2 indicates that the User Default Quota is enabled (see "To Configure Advanced Options" on page 155, step 3 or look at the `ims_default_quota` attribute in `/etc/opt/SUNWmail/ims/ims.cnf`)

   `allocated` = -1 indicates no quota limit.

   `0` indicates a NORMAL state and `1` indicates an ALERT state.

# Message Store Quota Enforcement



The message store quota enforcement allows administrators to limit the mail storage allocated to a user. This limit is called the *message store quota*. Once this feature is enabled, the system maintains a running tally of the disk space occupied by the user's messages. If the tally exceeds the specified message store quota, further mail

will be bounced back to the sender. User's can also be warned when their mail store usage approaches their quota so they are not surprised when people cannot send mail to them.

Message Store quota enforcement is useful for Organizations that wish to limit the amount of mail storage a user's mailboxes can occupy. In addition, this feature allows ISPs to allocate a certain amount of mail storage to their customers and to easily increase that allocation if the customer requests or purchases more.

---

**Note –** Message store quota enforcement can only be used with the Sun Message Store. This feature will not work with other message stores such as `/var/mail`.

---

## Mail Store Usage Calculation

Mail store usage is calculated by totalling the space usage of all the messages in all the user's mailboxes. If a message is sent to multiple users, SIMS adds the message size to the user's total usage—even though the message is only stored in one place with each user having a pointer to it.

## ▼ To Activate Message Store Quota Enforcement on an Installed System

When SIMS is installed, the message store quota feature is set to `OFF`. This means that user's mailboxes can occupy an unlimited amount of mail storage space. Implementing message store quota involves the following steps:

1. **Determine how much disk space is available for storing mailboxes and how much space each user can be allocated.**

   As a guideline, the default user message store quota is 20 megabytes. Change this in the Admin Console's Sun Message Store Property Book ("To Configure Advanced Options" on page 155) or by setting the `ims-default-quota` parameter in the `ims.conf` file to the desired default value.

2. **Set the message store quota for each user entry.**

   Configure the LDAP attribute `MailQuota` to one of three values, or set the quota to one of three options in the user's entry (see "To Modify a User Entry" on page 73). The options are:

   - Use Default User Quota - This option allocates the amount of storage specified in the Default User Quota set in the Message Store Property Book. (`MailQuota` = -2. If there is no `MailQuota` attribute, the system defaults to -2).

- No Store Limit - This turns off the message quota feature giving this user unlimited message store space. (`MailQuota` = -1)
- Set Individual Quota - Select a number and the unit of measure (KB or MB). This quota will not take effect until an incremental or full directory synchronization occurs (see "Alias Synchronization Schedule" on page 104 or see the `dirsync`, `iminitquota`, and `imquotacheck` command in the *SIMS Reference Manual* for more information). `MailQuota` = <*Size of Quota in Bytes*>.

The default is `MailQuota` = -2, Use Default User Quota.

3. **Shut down the SIMS server.**

   This prevents quota usage inconsistency.

   ```
   # im.server stop
   ```

4. **Activate the message store quota enforcement for the system.**

   Set the `ims-quota` parameter in the `ims.conf` file to `on`, or click the User quota enforcement option in the Sun Message Store Property Book—Advanced Options to `ON` (see page 152).

5. **Run /opt/SUNWmail/ims/sbin/iminitquota - a.**

   SIMS maintains a quota cache file for each user. This file contains the user's quota and the amount of space currently used. If a user's mailboxes exceed the amount of allocated storage, then further mail sent to the user is bounced back to the sender.

   As described in the above procedure, there is more than one way to enable the various quota options. TABLE 5-2 shows the action required to implement the desired option.

**TABLE 5-2** Message Store Quota Option-Action Matrix

| Option | Admin Console | ims.cnf | LDAP Directory |
|---|---|---|---|
| Activate SIMS quota checking | Set **User quota enforcement** to ON in Sun Message Store Property Book, Advanced Options. (See "To Configure Advanced Options" on page 155) | Set `ims-quota` to `on`. | Not Available |
| Set system default quota | Set **Default User quota enforcement** to ON in Sun Message Store Property Book, Advanced Options. (See "To Configure Advanced Options" on page 155) | Set `ims-default-quota` to <size-of-quota-in-bytes> | Not Available |
| Set user quota | In the user's property book entry select either **Default User Quota**, **No Store Limit**, or **Set Individual Quota**. See "To Modify a User Entry" on page 73) | Not Available | Set MailQuota to -1 (no limit), -2 (default quota) or N (N=quota in bytes) |

6. **Restart imaccessd.**

```
# im.server start
```

## ▼ To Set a User's Mail Store Quota

This procedure describes setting quota with the Admin Console. To set up quota at system installatin using bulk loading, refer to

1. **Bring up the user entry on the SIMS Admin Console.**

   See "To Find and View an Existing User/Group Entry" on page 69.

2. **Set the mail store quota in the Mail Information section.**

   This can be done by either setting the LDAP attribute `MailQuota` to one of three values, or setting the quota to one of three options in the user's entry (see "To Modify a User Entry" on page 73, Step 8b for details). The options are:

   - Use Default User Quota - This option allocates the amount of storage specified in the Default User Quota set in the Message Store Property Book. (`MailQuota` = -2. If there no `MailQuota` attribute, the system defaults to -2).
   - No Store Limit - This turns off the message quota feature giving this user unlimited message store space. (`MailQuota` = -1)
   - Set Individual Quota - Select a number and the unit of measure (KB or MB). This quota will not take effect until an incremental or full directory synchronization occurs (see "Alias Synchronization Schedule" on page 104 or see the `dirsync`, `iminitquota`, and `imquotacheck` command in the *SIMS Reference Manual* for more information). `MailQuota` = *<Size of Quota in Bytes>*.

   The default is `MailQuota` = -2, Use Default User Quota. For information on how to modify the LDAP directory see the SIMS Reference Guide.

3. **Quota takes effect after the next incremental directory synchronization.**

   See "Alias Synchronization Schedule" on page 104. If you don't want to wait for the next synchronization, you can activate quota enforcement immediately for a user by using the `iminitquota -u <username>` command. See the SIMS Reference Guide.

### ▼ To Warn Users When Their Mail Store Usage is Approaching Their Mail Store Quota

The `imquotacheck` command sends an email warning to users who are approaching their mail store quota. This command can be put in a `cron` file to provide a daily check on mail store users. Refer to the SIMS Reference Manual for further details.

# Configuring Advanced Options



The following is configurable in the Advanced Options section:

- User quota enforcement / default message quota
- Mail server client type
- Directory context for users
- Maximum connections permitted
- Percentage space left warning threshold
- `/var/mail` support
- Size by which to increase Sun Message Store
- Purge schedule

The following sections provide background information on each of the options that will help you decide if you want to configure them.

## User Quota Enforcement

By default, each SIMS user has no maximum amount of hard disk storage or *quota* that they can use for their mailboxes. They can use an unlimited amount of disk space for their incoming and stored messages. SIMS allows you to configure a quota for each Sun Message Store user. If you decide to implement user quotas, you can set a customized quota for each user, or you can set the default quota for a user. The default quota, which can also be changed, is 20 Mbytes. For complete details see “Message Store Quota Enforcement” on page 148.

## Mail Server Client Type

The Sun Message Store handles the parsing of messages from Internet Mail Access Protocol version 4 (IMAP4) clients and Post Office Protocol version 3 (POP3) clients in different ways. IMAP4 messages are preparsed and indexed when inserted into the Sun Message Store; no parsing is necessary when messages are accessed by mail client users. POP3 messages do not require parsing; therefore, the Sun Message Store does not parse these messages.

By default, the Sun Message Store treats all messages as messages from IMAP4 clients.

Since parsing takes CPU cycles and creates a need for more hard disk space, you may want to tune the amount of parsing that your mail server performs. If a majority of the messages stored by the Sun Message Store are from POP3 clients, you can change the default setting to POP3. The Sun Message Store will treat all messages as messages from POP3 clients and not parse them.

## Directory Context

The directory context is the point in the directory information tree (DIT) at which the search are begins for entries used to authenticate a user and password for Sun Message Store access. By default, the directory context is set to the organization layer (o=<*organization*>, c=<*country*>) in the DIT. Therefore, if a mail client user attempts to access messages in the Sun Message Store, a search for this user will begin at the organization layer of the DIT and progress through the lower layers of the DIT.

For example, imagine that you are the email administrator for the Bravo Corporation. Bravo's DIT is structured as shown in FIGURE 5-5.

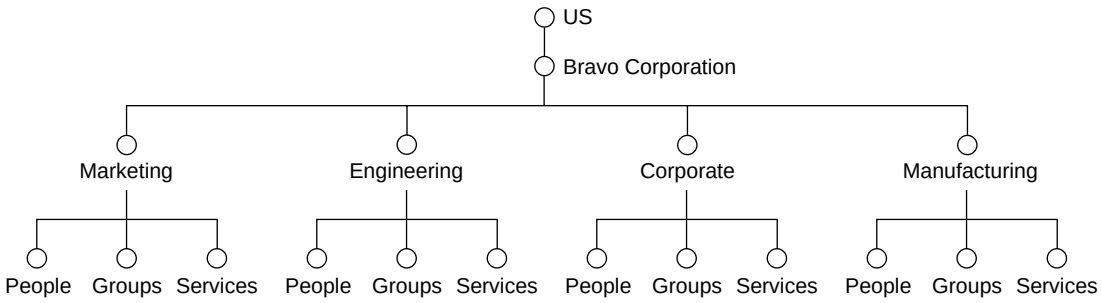



**FIGURE 5-5** Directory Information Tree of Bravo Corporation

Rather than starting a search at the default directory context of the Bravo organization, you want to reconfigure the directory context so that a search begins at the Marketing organizational unit. You can specify the following directory context:

```
ou=Marketing, o=Bravo, c=US
```

For more information on the DIT, refer to “Directory Structure” on page 39. For more information on the syntax used for the directory context, refer to “To Configure Advanced Options” on page 155.

## Maximum Connections Permitted

By default, the maximum number of active connections from IMAP4 clients that the Sun Message Store accepts is 10,000. The Advanced Option section in the Sun Message Store property book enables you to expand the maximum number of connections using the maximum connections permitted option. The up and down arrows for this option allow you to select a value in the range of 50 to 2,000,000,000 (billion).

When raising the maximum number of connections, keep in mind that the Sun Message Store daemons reserve shared memory for interprocess communication based on this number. If too high of a number is configured, the Sun Message Store fails to allocate sufficient shared memory to handle the maximum number of connections specified. The Sun Message Store will log an error message and exit.

If you attempt to configure the mail server to accept above the maximum number of connections permitted, `imaccessd` will log a message stating that the maximum connection number was exceeded and that the default number of 10,000 is being enforced.

## Percentage of Space Left Warning Threshold

By default, when the amount of hard disk space for the Sun Message Store is down to 5 percent, you will receive a warning in the System Status section of the Admin Console home page. The Advanced Option section in the Sun Message Store property book enables you to reconfigure the space threshold at which you will be warned.

## `/var/mail` Support

By default, access to mailboxes in `/var/mail` is not supported. The Advanced Option section in the Sun Message Store property book enables you to reconfigure this default so that users who have mailboxes in `/var/mail` can access the mailboxes using either the IMAP4 or POP3.

## Sun Message Store Increase

The Sun Message Store stores messages using a time-based structure. By default, a data directory contains 30 subdirectories, or one subdirectory for each day of the month. The data directory stores messages and attachments as files. For example, on May 1, all messages that enter the Sun Message Store are stored in the day 1 subdirectory whereas on May 15, all messages are stored in the day 15 subdirectory.

The Advanced Option section in the Sun Message Store property book enables you to reconfigure the time-based structure in which messages are stored using the Increase Store Size option. For example, if you specify 3 weeks, 21 subdirectories are created, 1 for each day of the 21 day interval. Messages entering the Sun Message Store on the first day of the interval are stored in the day 1 subdirectory and so on.

The distinction between the month and week intervals is that if you specify weeks, the Sun Message Store needs to allocate space more frequently than if you specify months. Modifying this feature has no impact on performance or resources.

## ▼ To Configure Advanced Options

Configuring these features is optional.

| AdminConsole>Sun Message Store>Advanced Options |
|---|

1. **From the Admin Console home page, click the Sun Message Store icon.**
2. **Click Advanced Options in the Section list.**

   The Advanced Options section appears, as shown in FIGURE 5-6.

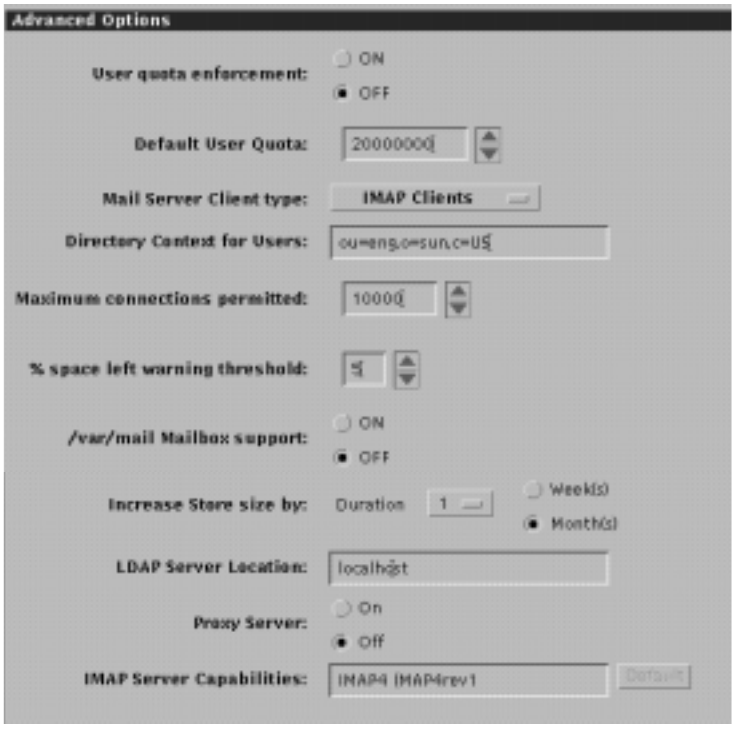



**FIGURE 5-6** Advanced Options Section (Extended View)

3. **Enable or disable the User Quota Enforcement option.**
   - Click ON to allow user message store space quotas to be set.
   - Click OFF to allow users unlimited message store space.

   See “Message Store Quota Enforcement” on page 148 for a complete discussion on user quotas. See “To Modify a User Entry” on page 73, Step 8b for instructions on how to set the message store quota for user entries.

4. **Reconfigure the default message quota by clicking the up or down arrow keys.**

   This is the default quota for new users if the User Quota Enforcement option is set to ON. See “To Modify a User Entry” on page 73, Step 8b for details on how to set a customized user quota.

5. **To configure how the Sun Message Store handles the parsing of messages from IMAP4 and POP3 clients, click the menu and choose the desired client type.**

   Mail Server Client Type has two choices: POP3 and IMAP4. However, if you set the configuration to IMAP4, the POP3 choice will be removed from the menu. Once IMAP4 client is set, you cannot change back to POP3 in the Admin Console. If the configuration is set to POP3, you can change it to IMAP4 to make the message store parse messages. (You can change from IMAP4 to POP3 if the message store has no data at all. Modify the `ims-parse-level` parameter in the `ims.cnf` file. See Message Access and Store Configuration chapter in the *SIMS Reference Guide*. If the message store has data, you cannot change it.)

6. **Configure the directory context at which the search to authenticate a user and password for Sun Message Store access begins. You must input the context in the following syntax:**

   ou = <*organizational unit*>, o = <*organization name*>, c = <*country*>

   You can specify no organizational units, or one or more organizational units. The organizational units, organization name, and country that you specify must be consistent with what is specified in your directory information tree (DIT). You can type the characters in either upper- or lowercase. Character spaces between commas (,) and equal signs (=) are permitted.

   The following is an example of a directory context:

   ```
   ou = Marketing, o = Bravo, c = US
   ```

7. **Configure the maximum number of connections from IMAP4 clients that the Sun Message Store accepts by clicking the up or down arrows.**

   The valid range includes 50 through 20 billion.

8. **Configure the "percentage of space left" warning threshold by clicking the up or down arrow keys.**

9. **Configure /var/mail support by clicking the appropriate radio button.**

10. **Configure the time-based structure in which messages are stored by choosing the desired number from the menu and clicking the radio button associated with the desired unit of measure.**

11. **Configure the LDAP server host name.**

    Typically this is localhost.

12. **Use the Proxy Server radio buttons and the IMAP Server Capabilities only to configure the server as a SIMS/Proxy server.**

    See "Setting Up a Proxy/Mail Server" on page 297.

13. **Click the Apply button.**

# Message Purge



When a message is delivered into the Sun Message Store, a reference pointing to the stored message is created in the Inbox of each of the message recipients. As each recipient reads, deletes, and removes (expunges) the message via their respective mail clients, the associated reference to the message is removed. When all references are removed or expired (see “Deleting Old Messages” on page 241), the message can be purged from the Sun Message Store.

Purge messages by manually executing the `impurge` command (see the *SIMS Reference Manual*) or by using the Admin Console to automatically run `impurge`.

---

**Note – Do not** wait until your disk is full before doing a message purge. Run a message purge while there is more empty disk space than the amount of space used by the mail store on its busiest 24 hour period. You can roughly calculate mail store disk usage by noting the disk usage increase on the `/var/opt/SUNWmail/ims` partition over a 24 hour period. If your message purge fails due to lack of disk space, refer to “Message Purge Failure” on page 256.

---

For additional information on Sun Message Store maintenance, including purge, refer to “Sun Message Store Maintenance” on page 233.

## Configuring Purge Options

The Admin Console enables you to configure the following purge options:

- Exhaustive purge - Locate all deleted messages that can be purged and purge them. (A deleted message is a message that is no longer referenced from any user or shared folder.)
- Customized purge - Perform daily computations to determine if the amount of deleted messages on a given day exceeds a percentage threshold and if the amount of disk space recovered if a purge is performed exceeds a size threshold.

Purging a message store supporting 20,000 users could take hours, so for some systems it may be preferable to choose Customized Purge rather than an Exhaustive Purge since fewer purges may be required. User’s can still send and receive mail during a purge, but the system cannot delete or expunge messages until after the purge is completed.

## Customized Purge

You can customize a policy whereby the purging of unreferenced or deleted messages from the Sun Message Store is performed on an as-specified basis rather than on a daily basis. The customized purge option enables you to set two thresholds: percentage and size.

Percentage is defined as the fraction of deleted messages of the total volume of messages handled by the Sun Message Store on a particular day. Size is defined as the total amount of disk space in kilobytes that can be recovered after deleted messages are purged. The system computes percentages and sizes that are compared against these thresholds daily.

For example, imagine that you have set the percentage threshold to 50 percent and the size threshold to 100 KB. Imagine that on the first day after a purge is performed (day 1), the system examines the total volume of messages handled by the Sun Message Store and the total volume of deleted messages. The system computes the percentage of deleted messages based on the total volume of messages. If the percentage of deleted messages is 51 percent or higher, then day 1 is purged. If the percentage of deleted messages is 50 percent or lower, then day 1 is not purged. Additionally, if he size of deleted messages exceeds 100 KB, then a purge is also performed.

# ▼ To Configure Purge Options

AdminConsole>Sun Message Store>Purge Options

1. **Click on the Sun Message Store icon in the home page.**
2. **Click on Purge Options in the Section List.**

   The Purge Options section appears as shown in FIGURE 5-7.

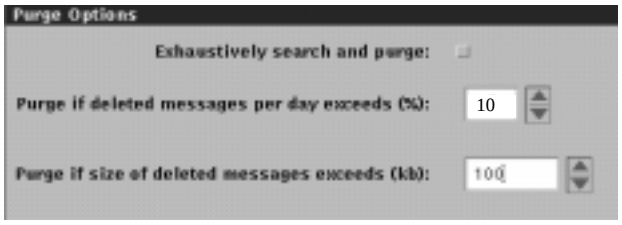



**FIGURE 5-7** Purge Options Section

3. **If you want to enable the exhaustive purge option, click the associated check box.**

4. **If you want to enable the customized purge option, click on the up or down arrow keys to specify the percentage and size thresholds.**

   The default percentage threshold is 1 percent, and the default size threshold is 100 KB.

5. **Click the Apply button.**

## ▼ To Configure the Purge Schedule

For information on Sun Message Store maintenance, including purge, refer to “Sun Message Store Maintenance” on page 233.

No default purge schedule exists. Therefore, if you want to *purge* or permanently remove messages that no longer have references from any folder on a regularly scheduled basis, you must set a schedule.

| AdminConsole>Sun Message Store>Schedule For Purging Deleted Messages |
|---|

1. **Click the Sun Message Store icon to access the Sun Message Store property book.**
2. **Click Schedule For Purging Deleted Messages in the Sections list.**

   The Schedule For Purging Deleted Messages section appears, as shown in FIGURE 5-8.

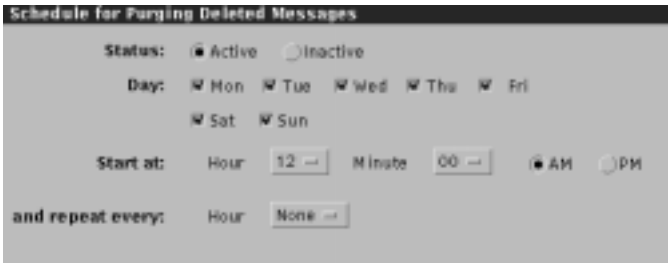



**FIGURE 5-8** Schedule For Purging Deleted Messages Section

3. **Activate purge schedule.**
4. **Configure the purge schedule.**

   Specify days and times at which you want purge to occur.

5. **If you want purge to occur at regularly scheduled intervals throughout the days specified in 4, to specify the interval at which the purge should occur.**
6. **Click the Apply button.**

# Message Access Protocol Connections



The Message Access Property Book allows you to view and monitor all user connections to SIMS, as well as start and stop message access to the message store.

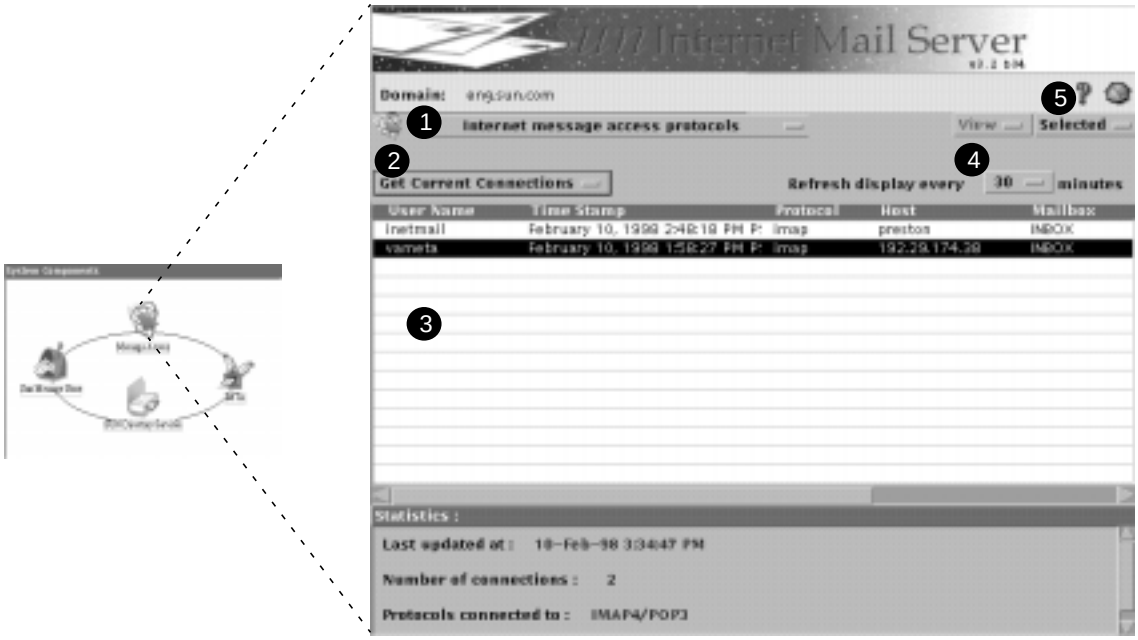



**FIGURE 5-9** Message Access Property Book

1. **Start message access protocols IMAP4/POP3**
   **Stop message access protocols IMAP4/POP3**

   Pulldown menu - allows you to start and stop client access to message store. Messages are still received and stored, but clients cannot access the messages.

2. **Get IMAP Connections**
   **Get POP3 Connections**
   **Get Both Connections**

   Pulldown menu - specify which user connections to display.

3. **Connections to SIMS**

   A list of all connections to the server by user, time, protocol, host, and open mailbox. The table displays the following fields related to each connection:

   - User Name - User name associated with the connection.
   - Time Stamp - Date and time at which user established connection to mail server.
   - Protocol - Message access protocol used to make connection.
   - Host - Hostname of machine from which the connection is made.
   - Mailbox - Mailbox that user is accessing via the connection. A user can access multiple mailboxes at any given time.

   The Statistics area provides the following information:

   - Date and time at which the connection table is accessed or updated.
   - Total number of specified connections.
   - Type of connections that you specified.

4. **(Optional) To change interval at which the connection table is updated (default: 30 minutes) click the Refresh display every X minutes**

   Pulldown menu - allows you to adjust the frequency of refresh from 30 to 300 minutes.

5. **Get Status**

   Returns status report of highlighted connection. You can also get this report by double clicking on the desired connection. The

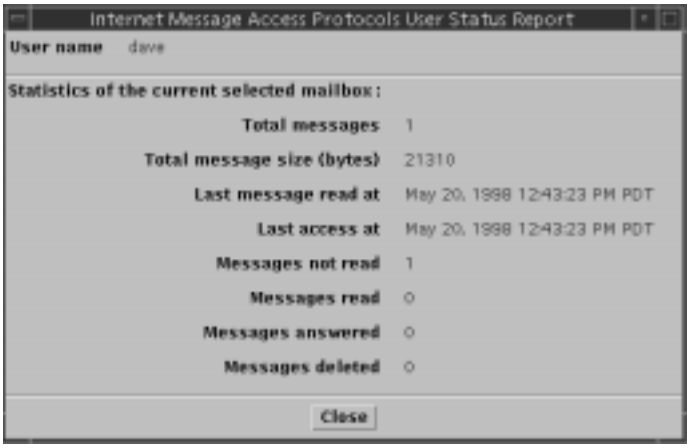



**FIGURE 5-10** Connection Status

CHAPTER 6

# Sun Directory Services Administration

This chapter provides step-by-step instructions for viewing and modifying the Sun Directory Services, also known as the LDAP Server or simply the Directory Service. To start, bring up the Sun Directory Services property book pages.

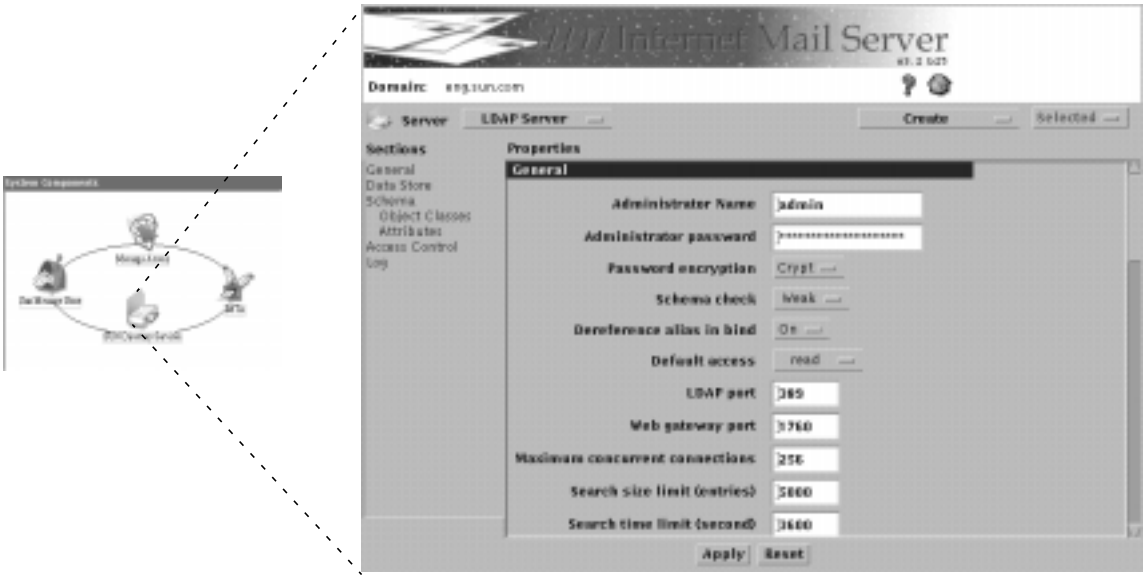



**FIGURE 6-1** Sun Directory Services Property Book

# Sun Directory Services Topics and Tasks

**TABLE 6-1** Sun Directory Services Topics and Tasks

| Topic/Task | Description | Page |
|---|---|---|
| Initial Configuration | Describes how to configure the two mandatory parameters: the administrator name/password and the distinguished name of the naming context held in the data store and the data store location. | 165 |
| Populating the Directory | - Saving and restoring existing data in the directory.<br>- Populate the directory by writing LDIF data directory into the LDBM database using `ldif2ldbm` and `ldbmcat`<br>- Populating the directory via the SLAPD server.<br>See Appendix C, “Populating the Directory Examples” on page 319 | 167 |
| General Properties Configuration | How to modify general Sun Directory Services properties:<br>- Administrator name, password, and encryption<br>- Schema check<br>- Dereference alias in bind<br>- Default access<br>- LDAP port and Web gateway port<br>- Maximum concurrent connections<br>- Search size and time limit<br>- Default referral host | 188 |
| Configuring the Data Store | Describes creating, modifying and indexing a data store. Also describes **Replication**. Examples of configuring replication are at page 301. | 190 |
| Modifying the Schema | Tells how to change and create attributes and object classes. | 200 |
| Configuring Access Control | How to modify directory access control rules and level of access granted. Also “Using the Distinguished Name Editor” on page 211. | 205 |
| Configuring Logging | Lists the information that can be logged. | 213 |
| Web Access to the Directory | How to search for and read entries, and to modify some directory information from a web browser. | 215 |
| Monitoring the Directory Service | View Sun Directory Services statistics—Global, Detailed, Operations, Associations,Interactions | 219 |
| Maintaining the Directory Service | **This section is in Chapter 7, “Maintenance.”** It describes:<br>- Maintaining the data store attribute indexes<br>- Backup and restore a data store<br>- Backup and restore directory data base<br>- Back up and restoring directory service configuration<br>- Starting and stopping the directory services | 241 |
| Directory Service Error Messages | Error Messages | 363 |

# Initial Configuration



When you install Sun Directory Services, most configurable characteristics are given default settings that enable you to start and run a directory server. The only parameters that you *must* configure are the name and password of the administrator and the distinguished name of the naming context held in the data store and the data store location. This is described in “To Specify an Administrator Name, Password, and Distinguished Name” on page 166). When you have specified these parameters, you have a server with a default configuration with the following characteristics:

- The port used for LDAP communications is 389.
- The port used by the email administrator’s configuration interface is 1760.
- Searches are limited to 5000 entries or 3600 seconds (1 hour). A search stops when the first of these limits is reached.
- The schema is checked for each add/modify directory operation.
- The data store is in `/var/opt/SUNWconn/ldap/dbm`.
- Log files are stored in `/var/opt/SUNWconn/ldap/log`.
- 1000 entries are cached.
- Default indexing is used, as described in “Indexing the Data Store” on page 194.
- Passwords are stored in encrypted format.
- Alias dereferencing on bind operations is enabled.
- The directory contains no entries (the name and password for the administrator are stored in the configuration file).
- Default access control is used, as described in CODE EXAMPLE 1-1 on page 39.
- There are no knowledge references to other servers.

---

**Note –** The SLAPD server is by default configured for only 256 maximum connections. Go to the Sun Directory Services page and in the General section change 256 to 1000.

---

## ▼ To Specify an Administrator Name, Password, and Distinguished Name

AdminConsole>SUN Directory Services>General Properties

1. **In the Admin Console home page, click the Sun Directory Services icon.**
2. **In the General Properties section of the Sun Directory Services properties, specify the name and password for the directory administrator.**

   The administrator name and password are stored in the configuration file, so that the administrator always has access to the directory. This is necessary so that the administrator can solve problems with access control, for example.
3. **Determine whether the administrator's password should be encrypted through the Password Encryption menu.**
   - To encrypt the password (the default), choose encrypt.
   - To store the password unencrypted, choose None.
4. **Click Data Store in the Sections list.**
5. **Choose Ldbm Data Store from the Create menu.**

   Alternatively, you can rename the default data store o=XYZ, c=US, by highlighting that line in the Data Store section, and choosing Modify from the Selected menu.
6. **Specify the distinguished name of the naming context that the directory server will store.**

   See "Using the Distinguished Name Editor" on page 211 for details of how to enter a DN in the Admin Console.
7. **Specify the path name to the directory where the data store is to be held.**

   For a default configuration, you do not need to specify any other information. For more information about creating a data store, see "Configuring the Data Store" on page 190.
8. **Click Apply.**

   The changes are implemented when you stop and start the `slapd` daemon.
9. **Log out of the Admin Console and then log back in.**

# Populating the Directory



This section describes two ways in which to populate the directory with entries for mail users, user aliases, and distribution lists.

1. Writing LDIF data directly into the LDBM database using `ldif2ldbm` and `ldbmcat`, bypassing the SLPAD server. This is a fast way of doing bulk loading. However, there is no schema checking done with this method. Use this method only if you are certain that your LDIF data is compliant with the schema supported by your SLAPD server.
2. Populating the directory via the SLAPD server. This is a safer, but more time-consuming way of populating the directory.

---

**Note –** By default SIMS assumes that all users receiving mail on this server have entries in the LDAP Directory. Mail will not be routed to users by SIMS until the LDAP Directory is populated with entries. SIMS can be configured to forward unroutable mail to a DNS “smarthost”; see “To Configure IMTA Position Relative to the Internet” on page 114 if you wish mail to be forwarded to the smarthost in the event the intended recipient is not in the LDAP Directory.

---

Directory entries are created from the `passwd(4)` file and the mail `aliases(4)` file. These procedures describe populating the directory for use by SIMS, not for general purpose directory use. “Populating the directory” in this scope means “adding User and Alias entries to the directory for use by SIMS.” Other attributes may be added to the directory for other uses, but they must not interfere with the attribute/value pairs used by SIMS.

For a look at some sample directory population sessions, refer to “Populating the Directory with User Data—Sample Session” on page 319, “Populating the Directory with User Aliases Data and Distribution Lists —Sample Session” on page 323, and “Populating the Directory with User Aliases Data and Distribution Lists —Sample Session” on page 323.

## Setting the Environment for Directory Population

These procedures use many different commands and several different configuration files. Add the paths to these commands to your own shell paths or `$MANPATH`.

- Executable programs and scripts:

  ```
  /opt/SUNWmail/bin
  /opt/SUNWmail/sbin
  /opt/SUNWconn/bin
  /opt/SUNWconn/sbin
  ```

- Directory man pages:

  ```
  /opt/SUNWmail/man
  /opt/SUNWconn/man
  ```

- Directory and other management scripts ("`slapd`" to start/stop the directory server, "`web500gw`" to start/stop the HTML dirsvc server, etc.):

  ```
  /etc/init.d
  ```

- Default location of directory configuration files:

  ```
  /etc/opt/SUNWconn/ldap/current
  ```

## ▼ Saving and Restoring Existing Data in the Directory

Although the directory can't be used to route mail for SIMS until after it is populated with entries, don't assume that the current directory is completely empty. Do not use `ldif2ldbm(1M)` or do any other actions which truncate the directory without first saving possible contents using the `ldbmcat(1M)`. Save the contents of the directory for later restoration (using `ldif2ldbm`) as follows:

1. **su to root**
2. **Make sure neither SIMS nor slapd are running:**

   ```
   /etc/init.d/im.server stop
   /etc/init.d/slapd stop
   ```

3. **cd /opt/SUNWconn/sbin**
4. **Decide on a destination directory which has sufficient space to store the contents of the directory in LDIF. In this example we use /tmp.**

5. **Run the `ldbmcat` command.**

   In the C shell:

   ```
   % ./ldbmcat -n /var/opt/SUNWmail/ldap/dbm/id2entry.dbb >&
   /tmp/dbm.ldif
   ```

   Note that if the directory was empty this will produce an empty file. In this case, you do not need to run the subsequent steps to restore data.

6. **After saving the existing data to a file (`/tmp/dbm.ldif` in this example), create the new LDIF for entries you plan to add (ex: `new.ldif`).**

   This process is described the sections that follow.

7. **Concatenate the new LDIF onto the old.**

   Example: `cat new.ldif >> dbm.ldif`

8. **Load the database with the new LDIF using ldif2ldbm.**

   If your database is not empty then you will have to use the `-c` argument to `ldif2ldbm` to overwrite the database.

   Example: `ldif2ldbm -c -i dbm.ldif`

   Note that faster loading can be attained by using the `-j` parameter to `ldif2ldbm`.

## Using `ldif2ldbm` and `ldbmcat` to Initially Populate Local Directories

`ldif2ldbm(1M)` is a way of writing LDIF data directly to the `ldbm` database format used by the directory provided with SIMS. `ldif2ldbm(1M)` must be done locally. It also bypasses certain checks (schema checking of attributes that are mandatory, for example), and therefore may be faster in certain circumstances for bulk-loading large amounts of data into the directory. For example, restoring up a damaged directory from stored LDIF data, or for initially populating a directory from a new batch of LDIF data. Users of `ldif2ldbm` are advised to carefully read the man pages and to practice their proposed use of this tool in an environment where any mistakes will not affect the operation of shared resources. Some important reminders about `ldif2ldbm` are:

- `ldif2ldbm` *truncates* the existing `ldbm` databases when it is invoked, to ensure that no existing data can corrupt the bulk-load it is about to carry out. If you wish to use `ldif2ldbm` on an existing, intact, database, you should use `ldbmcat(1M)`, with '`-n`' flag to first dump the existing `ldbm` database to LDIF, to which the new LDIF is then concatenated, before loading the entire new batch of LDIF data.
- `ldif2ldbm` completely bypasses the directory schema enforced by the `slapd` directory server. Administrators must be *certain* that data they are entering meets the schema which `slapd` enforces via it's `slapd.conf, slapd.oc.conf` and

`slapd.at.conf` files.There are two ways LDIF data may be added to the directory; by using the LDAP protocol (via `ldapmodify(1)`, or by direct modification to the `ldbm` database used by the directory (via `ldif2ldbm(1M)`). Use of the former method is recommended as it does not require you to be on the same system as the database, and automatic merging of existing entries with new values is done. However the latter method may be used by skilled system administrators who are familiar with the procedure, as it requires saving data already in the directory service to prevent data loss.

# Populating the Directory Via the SLAPD Server



When you populate the directory, you will perform the following steps:

1. Gather the data used to populate the directory by taking existing data from other naming or directory services (NIS, NIS+, or `/etc/passwd` and `/etc/mail/aliases`)
2. Format the data used to populate the directory to ensure that the data can be read by the `imldifsync(1M)` program
3. Convert the data to LDIF format using the `imldifsync(1M)`command (or your own custom scripts that follow the rules documented below).

---

**Note –** `imldifsync(1M)` replaces `ldapsync(1M)` in Sun Internet Mail Server 3.5, for the purposes of generating LDIF for use by SIMS. `imldifsync` supports the same interfaces as `ldapsync`, but in addition supports new features such as the client software Web Access. `ldapsync` is a deprecated interface and will be eliminated in a future release.

---

4. Add/modify LDIF data into the directory database used by the directory service daemon `slapd`. Each `passwd` and `alias` file entry generates numerous lines of LDIF data based on interpretation rules encoded in `imldifsync(1M)`.

The LDIF attributes and interpretation rules needed by SIMS are listed starting on page 175,. Use these to write your own scripts or translation programs to convert `passwd` and `alias` file data into LDIF. We recommend that you use `imldifsync(1M)` at least as an experimental tool to help you understand how to write scripts that generate LDIF.

---

**Note –** After initially populating the directory with NIS/NIS+ user entries, you must also you must repopulate the directory whenever you update NIS or NIS+ with new email user entries. The procedure we describe for initially populating the directory (in the following sections) is the same procedure for repopulating the directory.

---

## Starting and Stopping SIMS Components

You need to have `slapd` running while populating the directory, because `imldifsync` will communicate with `slapd`. The IMTA and `imaccessd` daemon should not be running as they rely on a correctly populated directory to work properly. These programs should be restarted after populating the directory.

To stop `imaccessd` and all SIMS components use:

```
/etc/init.d/im.server stop
```

When using `ldapmodify`, `ldapadd`, or `ldapdelete` to change what's in the directory, use the following command to ensure SLAPD is running.

```
/etc/init.d/slapd start
```

---

**Note –** The `imaccessd` process should never be killed using the `kill -9` command. Use `kill` without the `-9` argument. If `kill -9` is used, run `imcheck -c` before restarting `imaccessd`.

---

## Gathering Data Used to Populate the Directory

You will be adding two types of data to the directory:

- user information (from `/etc/passwd` or its equivalent)
- user mail alias and distribution list data from `/etc/mail/aliases` or its equivalent.

This data may come from /etc files or from NIS or NIS+ databases. However it must be in a concise format before it can be converted to LDIF by imldifsync(1M)

The method for extracting distribution list data depends on whether your system is using NIS, NIS+, or /etc files. The following section details how to use the supplied tools to do this for simple user installations. If you have a complex installation you may prefer to write your own tools (using the supplied client side LDAP tools); in that case it is still recommended that you understand the following process before proceeding on your own.

## ▼ Gathering Directory Data on Systems Using /etc Files

The steps below tell how to obtain user-passwd and mail-alias data from system files. When the SIMS IMTA is installed, mail alias and distribution list information is taken from the directory, rather than /etc/mail/aliases. Unless you set up a way for /etc/mail/aliases to update the directory, it will no longer be used. In this case, you should add a comment in the /etc/mail/aliases file to serve as a warning to other system administrators who attempt to add or update aliases.

1. **Log in as root.**

```
$ su
Password: <Enter your root password>
```

---

**Note –** During this process be **extremely** careful to not edit /etc/passwd!

---

2. **Change directory to** /tmp **and issue the copy command to create a single** passwd **file with all the entries required by** imldifsync(1M):

```
# cd /tmp
# sort /etc/passwd > passwd.tmp
# sort /etc/shadow > shadow.tmp
# join -j1 1 -j2 1 -o 1.1 2.2 1.3 1.4 1.5 1.6 1.7 -t: passwd.tmp
shadow.tmp > passwd
# rm passwd.tmp shadow.tmp
```

---

**Note –** You may use the passwd and shadow file directly instead of the “join” above by using the “passwd-file” and “shadow-file” options in the imldifsync.conf file discussed below.

---

3. **Change directory to** `/tmp` **and issue the copy command as shown to create a mail** `aliases` **file for use by** `imldifsync`:

```
# cd /tmp
# cp /etc/mail/aliases aliases
```

## ▼ Gathering Directory Data on Systems Using NIS

To obtain user-passwd and mail-alias data from system files, perform the following steps:

1. **Log in as root.**

```
$ su
Password: <Enter your root password>
```

**Note –** During this process be extremely careful to not edit /etc/passwd!

2. **Change directory to** `/tmp` **and issue the** `getent(1M)` **command to create a single** `passwd` **file with all the entries required by** `imldifsync(1M)`:

```
# cd /tmp
# getent passwd > passwd
```

3. **Change directory to** `/tmp` **and issue the** `ypcat(1)` **command as shown to create a mail** `aliases` **file for use by** `imldifsync`:

```
# cd /tmp
# ypcat -k mail.aliases > /tmp/aliases.tmp
# sed 's/ /: /' /tmp/aliases.tmp > /tmp/aliases
# rm aliases.tmp
```

## ▼ Gathering Directory Data on Systems Using NIS+

To obtain user-passwd and mail-alias data from system files, perform the following steps:

1. **Log in as root.**

```
$ su
Password: <Enter your root password>
```

**Note –** During this process be extremely careful to not edit /etc/passwd!

2. **Change directory to** `/tmp` **and issue the** `getent(1M)` **command as shown to create a single passwd file with all the entries required by** `imldifsync(1M)`:

```
# cd /tmp
# getent passwd > passwd
```

3. **Change directory to** `/tmp` **and issue the** `niscat(1)` **command as shown to create a mail** `aliases` **file for use by** `imldifsync`:

```
# cd /tmp
# niscat mail_aliases.org_dir > /tmp/aliases.tmp
# sed 's/ /: /' /tmp/aliases.tmp > /tmp/aliases
# rm /tmp/aliases.tmp
```

## Formatting Data Used to Populate the Directory

This section describes how to format the user, mail-alias, and distribution list data to successfully populate the directory.

User information must be in the format defined in `passwd(4)`, or as you would find in `/etc/passwd`. An LDIF file will be generated from different fields of each user entry, and user entries will be cross referenced with user alias information (from data you provide of the format found in `aliases(4)`) to create LDIF attribute definitions used by SIMS. The `imldifsync(1M)`command which generates LDIF output makes certain rigid assumptions about the format of the gecos field of a user passwd entry:

## ▼ passwd File Format Rules for imldifsync(1M)

The imldifsync command converts information in the passwd file to LDIF, which is the format required for adding entries to the directory database. If you do not specify your own conversion program or script with the option –G, imldifsync(1M) uses a default conversion program which expects the gecos field to be in the following format:

```
...:given-name surname, generation-qualifier - comment:...
```

The gecos field is the fifth field in the sequence of colon-separated fields in the passwd file.

An example gecos-parsing script which can be used with the -G option is in /opt/SUNWmail/dir_svc/samples/imgecos2cn.sh, and may be specified in the imldifsync.conf file (discussed below) via the gecos2cn-prog option.

A gecos field that does not conform exactly to this format *may* still be parsed successfully by imldifsync, and in this case an LDIF directory entry will be created for it; however, a warning message will be generated for each syntactical error that imldifsync encounters, and the resulting attributes may differ from those expected, requiring administrators to make extra efforts to manually ensure the generated LDIF is useful for SIMS.

The following rules are applied when imldifsync parses the gecos field:

### *Rule 1 — General Format*

The given-name, surname, and generation-qualifier must start with an alphabetic character, can contain alphabetic characters, dashes (-) and single quotes (’), and must end with either an alphabetic character or a period (.). With the specific exceptions described in “Rule 4 — Surnames” on page 176, uppercase and lowercase characters have no special significance.

The following examples would be converted to valid LDIF directory entries:

- :Alice Mary White:
- :Philip O’Connor, Jr.:
- :John-Paul Simon - mktg consultant:

The following examples would generate a warning message:

- :+Aaron J. Brown:
- :Esther Great!:
- :Mary Anderson *sales*:

### *Rule 2— Comments*

Anything that follows a space-dash-space sequence ( - ) is interpreted as a comment; also, anything that is enclosed in double quotes or in brackets, even in a non-matching pair, is interpreted as a comment. There can be multiple comments in a single gecos field.

For example:

- `:Kevin Ascot - Sales Mgr.:` the comment is “Sales Mgr.”
- `:Brian Scott (surgeon):` the comment is “surgeon”
- `:Ellen Chelly [CONTRACTOR]:` the comment is “CONTRACTOR”
- `:Ross "the expert" Brand:` the comment is “the expert”
- `:Janice Evans (Quality Group}:` the comment is “Quality Group”
- `:Robert (Bob) Jones - Mktg:` the comments are “Bob” and “Mktg”

### *Rule 3— Generation-Qualifiers*

If there is a comma anywhere in the gecos field (except in comments), the words that follow it are interpreted as a generation-qualifier. The generation-qualifier is optional, but if present, it must not be blank.

The following examples would be converted to valid LDIF directory entries:

- `:John Smith,Jr.:`
- `:John Smith, Senior:`

The following examples would generate a warning message:

- `:John Smith,:`
- `:John Smith, - Snr:`

### *Rule 4 — Surnames*

The surname is either the last word in the gecos field, or the last word before either a generation-qualifier or a comment. If there is only one word in the gecos field, it is assumed to be the surname. If there are no words in the gecos field, the username is assumed to be the surname.

For example:

- `:Kate Black:` the surname is “black”
- `:Ann Mary Wells:` the surname is “Wells”

- `:John Smith, Jr.:` the surname is “Smith”
- `:Erwin David BLINK - Engineer:` the surname is “BLINK”

Surnames can also consist of several words. In this case, the capitalization is used to distinguish between words that are part of the given-name and words that are part of the surname.

Words that immediately precede the surname, and that are also either *all uppercase* or *all lowercase* are interpreted to be part of the surname. This allows naming prepositions such as “le” or “de” in french, and “von” or “van” in german, to be interpreted correctly.

For example:

- `:Jean-Pierre le GAD:` the surname is “le GAD”
- `:Joe van der Graf:` the surname is “van der Graf”
- `:Jose MARCOS SOUZA:` the surname is “MARCOS SOUZA”
- `:Franz Josef von Bismark:` the surname is “von Bismark”

Note the unexpected effect that the application of this rule may have if the gecos field is all lowercase or all uppercase, or if there is an initial letter preceding the surname.

For example:

- `:gerhard ellis sumner:` the surname is “gerhard ellis sumner”
- `:ADRIENNE CHIU (sales):` the surname is “ADRIENNE CHIU”
- `:Peter K. Wolff:` the surname is “K. Wolff”

### *Rule 5 — Given-Name*

Once the other components of the gecos field have been identified, the remaining words are interpreted as the given-name.

For example:

- `:Jean-Pierre le GAD:` the surname is “gerhard ellis sumner”
- `:Joe van der Graf:` the given-name is “Joe”
- `:Jose MARCOS SOUZA:` the given-name is “Jose”
- `:Franz Josef von Bismark:` the given-name is “Franz Josef”
- `:Peter K. Wolff:` the given-name is “Peter”

If your user passwd information does not meet this criteria then you have three alternatives:

- Convert the passwd data to the above format required by `imldifsync(1M)`, using your own custom written scripts to modify the gecos field. Run `imldifsync` using that data.
- Write your own gecos parsing script as documented in the `imldifsync` manpage, using the example `gcos2cn.sh`, in `/opt/SUNWmail/ldap/samples`, and pass that to `imldifsync(1M)` via the `-G` flag
- Do not use `imldifsync` at all, but instead write your own LDIF generator that produces LDIF entries with the attributes that SIMS requires.

## ▼ `aliases` File Format for `imldifsync`

The `imldifsync`(1M) command makes assumptions about the format of the mail `aliases` file used as input. The command uses the information in the `aliases` file to generate attributes for an entry. The expected format for the `aliases` file is described below.

---

**Note –** Refer to the manpage for the `aliases(4)` file for general usage information.

---

### *Rule 1 — General Format for User Aliases*

For each user, the `aliases` file must contain two lines in the following format:

```
userid: first.lastname
first.lastname: userid@mailhost
```

where:

*userid* is the same as the user ID in the first field of the `passwd` file

*first.lastname* is usually a concatenation of the user's given name and surname

*mailhost* is the machine where the user's mailbox resides.

When the `aliases` file contains this type of information for a user, `imldifsync(1M)` creates the following attributes:

- `preferredRfc822Originator` with value *first.lastname@maildomain*
- `preferredRfc822Recipient` with value *userid@mailhost.maildomain*
- `rfc822Mailbox` with values *first.lastname@mailhost.maildomain*, *userid@mailhost.maildomain*, and also with the same value as preferredRfc822Originator
- `mailDeliveryOption` with value **mailbox**

- `mailHost` with value *mailhost.maildomain*

---

**Note –** The *maildomain* in the attribute values is the mail domain declared in the configuration file for the `imldifsync` command. This mail domain must be the same as the one declared in the `slapd.conf` configuration file.

---

For example, the `aliases` file contains the following lines for user Allyn Robinson:

```
arobin: allyn.robinson
allyn.robinson:arobin@cloud
```

The `imldifsync` command extracts the following attributes and attribute values from this information:

```
preferredRfc822Originator: allyn.robinson@Marketing.XYZ.COM
preferredRfc822Recipient: arobin@cloud.Marketing.XYZ.COM
rfc822Mailbox: allyn.robinson@cloud.Marketing.XYZ.COM
rfc822Mailbox: arobin@cloud.Marketing.XYZ.COM
rfc822Mailbox: allyn.robinson@Marketing.XYZ.COM
mailDeliveryOption: mailbox
mailHost: cloud.Marketing.XYZ.COM
```

### *Rule 2 — Handling Differing User IDs*

The user ID supplied on the first line can be different from the user ID on the second line.

```
userid1: first.lastname
first.lastname: userid2@mailhost
```

In such cases, `imldifsync(1M)` creates the `mailDeliveryOption` attribute with the value **forward**, and also creates a `mailForwardingAddress` attribute.

For example, the `aliases` file could contain the following lines for user Allyn Robinson:

```
arobin: allyn.robinson
allyn.robinson:jconnors@cloud
```

The `imldifsync` command extracts the following attributes and attribute values from this information:

```
preferredRfc822Originator: allyn.robinson@Marketing.XYZ.COM
mailForwardingAddress: allyn.robinson@cloud.Marketing.XYZ.COM
mailForwardingAddress: jconnors@cloud.Marketing.XYZ.COM
mailDeliveryOption: forward
```

### *Combining Rule 1 and Rule 2*

You can combine the general format described in Rule 1 with the format described in Rule 2, as follows:

*userid1: first.lastname*
*first.lastname: userid1@mailbox, userid2@mailhost*

In such cases, `imldifsync(1M)`creates the `mailDeliveryOption` attribute with the values **mailbox** and **forward**, and creates the `mailForwardingAddress` attribute.

For example, the `aliases` file could contain the following lines for user Allyn Robinson:

```
arobin: allyn.robinson
allyn.robinson:arobin@cloud, jconnors@cloud
```

The `imldifsync` command extracts the following attributes and attribute values from this information:

```
preferredRfc822Originator: allyn.robinson@Marketing.XYZ.COM
preferredRfc822Recipient: arobin@cloud.Marketing.XYZ.COM
rfc822Mailbox: allyn.robinson@cloud.Marketing.XYZ.COM
rfc822Mailbox: arobin@cloud.Marketing.XYZ.COM
rfc822Mailbox: allyn.robinson@Marketing.XYZ.COM
mailForwardingAddress: jconnors@cloud.Marketing.XYZ.COM
mailDeliveryOption: mailbox
mailDeliveryOption: forward
mailHost: cloud.Marketing.XYZ.COM
```

### *Rule 3— Handling Nicknames*

An `aliases` file can contain more than two lines per user, in which case, the format to observe is:

*userid: first.lastname*
`nickname1:`*first.lastname*
`nickname2:`*first.lastname*
*first.lastname: userid@mailhost*

In such cases, the `imldifsync(1M)` command creates the `rfc822Mailbox` attribute with an extra value for each nickname.

For example, the `aliases` file could contain the following lines for user Allyn Robinson:

```
arobin: allyn.robinson
arobinson: allyn.robinson
allyn: allyn.robinson
allyn.robinson: arobin@cloud
```

The `imldifsync` command extracts the following attributes and attribute values from this information:

```
preferredRfc822Originator: allyn.robinson@Marketing.XYZ.COM
preferredRfc822Recipient: arobin@cloud.Marketing.XYZ.COM
rfc822Mailbox: allyn.robinson@cloud.Marketing.XYZ.COM
rfc822Mailbox: arobin@cloud.Marketing.XYZ.COM
rfc822Mailbox: arobinson@cloud.Marketing.XYZ.COM
rfc822Mailbox: allyn@cloud.Marketing.XYZ.COM
rfc822Mailbox: allyn.robinson@Marketing.XYZ.COM
mailDeliveryOption: mailbox
mailHost: cloud.Marketing.XYZ.COM
```

### *Rule 4— Handling File Names in Aliases*

A user alias in the `aliases` file can contain a file name, following this format:

*userid: first.lastname*
*first.lastname: filename*

where *filename* must start with a slash ( / ).

In such cases, `imldifsync(1M)` creates the `mailDeliveryOption` attribute with the value “file”, and also creates a `mailDeliveryFile` attribute.

For example, the `aliases` file could contain the following lines for user Allyn Robinson:

```
arobin: allyn.robinson
allyn.robinson:/var/allyn/mail
```

The `imldifsync` command extracts the following attributes and attribute values from this information:

```
preferredRfc822Originator: allyn.robinson@Marketing.XYZ.COM
mailForwardingAddress: allyn.robinson@cloud.Marketing.XYZ.COM
mailDeliveryFile: /var/allyn/mail
mailDeliveryOption: forward
mailDeliveryOption: file
```

### *Rule 5— Handling Program Names in Aliases*

A user alias in the `aliases` file can contain a program name, following this format:

*userid: first.lastname*
*first.lastname: |programName*

Note that the pipe ( | ) symbol is required to introduce a program name.

In such cases, `imldifsync(1M)`creates the `mailDeliveryOption` attribute with the value “program”, and also creates a `mailProgramDeliveryInfo` attribute.

For example, the `aliases` file could contain the following lines for user Allyn Robinson:

```
arobin: allyn.robinson
allyn.robinson:|/bin/cat
```

The `imldifsync` command extracts the following attributes and attribute values from this information:

```
preferredRfc822Originator: allyn.robinson@Marketing.XYZ.COM
mailForwardingAddress: allyn.robinson@cloud.Marketing.XYZ.COM
mailProgramDeliveryInfo:  /bin/cat
mailDeliveryOption: forward
mailDeliveryOption: program
```

### *Rule 6— General Format for Group Aliases*

For each group (distribution list), the `aliases` file must contain two lines in the following format:

```
owner-aliasname: owner1 [owner2 ...]
[aliasname-request: processor]
aliasname: user1, user2, user3 ...
```

where:

*aliasname* is the name of the alias

*owner1*, *owner2*, ... are the names of the owners of the alias. An owner can be a member of the group, but not necessarily.

*processor* is the name of entity who will be responsible for processing requests sent to the alias

*user1, user2, user3,* ... are the members of the alias

The owner, processor and member entities defined above can be:

- A person with an entry in the directory
- A person known by an rfc822 mail address without an entry in the directory
- A program, introduced by the pipe ( | ) symbol
- A file, introduced by a slash ( / )

Depending on where you obtained your data (/`etc` files, NIS, NIS+) you may have to further format data.

## Converting the Data to LDIF Format

LDIF (LDAP Data Interchange Format) is a canonical data form used to represent entries in LDAP databases. Currently a draft-Internet-RFC, LDIF is designed to be a transportable intermediate data form that is portable between LDAP directories. Data must be converted to LDIF before it can be added to the directory. Administrators may use one of several methods to generate LDIF:

- Use the supplied `imldifsync(1M)` program to synchronize input data with data already added to the directory (if such exists).
- Write your own scripts or programs to generate LDIF based on `passwd`, user alias, and distribution list data.

---

**Note –** Although this section will involve two files; `passwd` data first, `alias`/ distribution-list data second, both user passwd and user alias information will be required in the first pass. Do not continue until you have both data-sets ready to use.

---

### *A Few Words About `imldifsync(1M)`*

`imldifsync(1M)` does several things:

- maps password and alias entry information into LDIF output.
- correlates password file user entries with alias file user entries.
- creates certain LDIF attribute values based on `passwd` and `alias` input.
- fabricates certain DNs required by `slapd` if they are not present in the ldbm.
- synchronizes changes in the input password and alias files and converts those differences to LDIF. `imldifsync` may be used to periodically synchronize the LDAP directory with changes to the password and alias files (for example, if users are added or deleted to the password file).

The default configuration file, `imldifsync.conf`, is installed in `/etc/opt/SUNWmail/dir_svc/imldifsync.conf`. Converting data using `imldifsync` is a two phase process. First, the user/passwd data is converted, then the mail-alias/groups data. You will need two separate configuration files, one for each phase. We will call these two files `passwd.conf` and `aliases.conf`.

The SIMS installation GUI will set certain values in the default `imldifsync.conf` based on your input. You should keep track of the settings by keeping an untouched copy of `imldifsync`.

**Note –** The `imldifsync.conf` file is readable only by `root`, because the file contains the “bind-DN” and “ldap-passwd” directives, which the SIMS install GUI will set based on what you enter as your Administrative password. You should be aware that anyone with this bind-DN and password can change any aspect of the Directory contents or configuration, and guard the bind-DN and password appropriately.

You should be sure that the following lines in the `imldifsync.conf` are set to values which make sense in the context of your location, organizational structure, and / or Directory Information Tree (DIT):

- bind-DN = “cn=Admin, o=XYZ, c=US”
  - CN is the administrator name provided during installation.
- ldap-passwd = “secret”
  - This is the administrator password provided during the installation.

**Note –** This information is used for `ldapadd`, `ldapmodify`, `ldapdelete` and `ldapsearch` binding and access checking. Anyone with this information can modify your directory and can get into the Administration Console. Protect your information accordingly!

- group-base = “ou=Groups, ou=eng, o=XYZ, c=US”
- base = “ou=people,ou=eng,o=XYZ,c=US”
  - This information defines the DIT within your directory, and is used as arguments to directory queries, including using the `-b` flag of `ldapsearch(1)`. These will be set to a value based on input to the SIMS installation procedure.
- mail-domain = “Marketing.XYZ.COM”
- super-domain = “XYZ.COM”
  - This information is used to generate LDIF used by SIMS to determine if incoming mail matches recipients in the LDAP Directory. They should match the DNS domain name in which the SIMS server is operating.

## ▼ Converting the Data to LDIF Format Using `imldifsync(1M)`, and Adding Data to the Directory Using `ldapmodify(1M)`.

1. **Change directories to the location shown and edit the `imldifsync.conf` file, after making a copy of the file as configured by the SIMS install process.**

   In the event the modified versions of the `imldifsync.conf` file are lost or damaged you will have the original file saved.

```
# cd /etc/opt/SUNWmail/dir_svc
# cp imldifsync.conf imldifsync.conf.SIMS3.5
# vi imldifsync.conf
```

2. **Uncomment the `mode = users`, `mail-server`, `passwd-file`, and `aliases-file` files, and change the `mail-server`, `passwd-file`, and `aliases-file` values as shown:**

```
mail-server = "<mailserverhostname>.<fully qualified domain name>"
passwd-file = "/tmp/passwd"
aliases-file = "/tmp/aliases"
mode = users
```

   In the above example, your *mailserverhostname* should be the name of the mailserver hostname on which you're running SIMS, and the *fully qualified domain name* should be equivalent to the mail-domain attribute in `imldifsync.conf`.

   By default the imldifsync.conf file contains two lines like this:

```
add-val = { "mailFolderMap: SUN-MS" }
ignore-attr = { "mailFolderMap" }
```

   "SUN-MS" is the recommended mailstore for SIMS. However if you chose to use the "/var/mail" type of message store during your Install, you should change "SUN-MS" to "UNIX V7".

   You may choose to add other attributes here as well, but there are two rules that must be followed:

   - Only attributes from the default SIMS schema may be added in this way. User created attributes cannot be added through this interface.
   - As shown in the example, only attributes which have values common to all users (for the user data generation pass) or groups (for the group/alias-data generation pass) should be added here.

3. **Create two copies of the** `imldifsync.conf` **files, one for users (for example,** `users.conf`**) and one for user aliases/distribution lists (for example,** `groups.conf`**).**

```
# cp imldifsync.conf users.conf
# cp imldifsync.conf groups.conf
```

4. **Edit the** `groups.conf` **file, change the** `add-val` **line as shown below and change the** `mode` **token from** `users` **to** `groups`**.**

```
add-val = { "associatedDomain: <same value as mail-domain> }
mode = groups
```

`mail-domain` value should be lower down in the same file.

5. **If you want to set a user mail store quota edit** `users.conf`**.**

The SIMS default setting is "no limit." To set a space limit, modify the `mailQuota` attribute as follows:

```
add-val = { "mailQuota: <quota in bytes>" , "mailFolderMap: SUN-MS" }
```

where `<quota in bytes>` would be 10000000 if you wanted to set a mail space quota of 10 megabytes. See "Message Store Quota Enforcement" on page 148 for detailed information on setting quotas.

6. **Convert the user data to LDIF format.**

Use the `imldifsync` command to generate formatted user data (`LDIF`) by issuing the following command:

```
# /opt/SUNWmail/sbin/imldifsync -c users.conf > /tmp/users.ldif
```

7. **Populate the directory with the user LDIF formatted data.**

This must be done before running imldifsync during the alias-data generation pass because imldifsync compares existing data in the directory to newly generated data to determine what will be generated as the groups data.

Use the `ldapmodify` command to add the new entries to the directory:

```
# /opt/SUNWmail/bin/ldapmodify -D bind-DN -w ldap-passwd -f /tmp/
users.ldif
```

Refer to the bulleted section above for `bind-DN` and `ldap-password` information.

---

**Note –** If you already have some entries in the Directory database you should specify the -c argument to ldapmodify in addition to those above, so that ldapmodify will continue to the new entries. `ldapmodify` will otherwise exit if it tries to add an entry that is already in the Directory.

---

8. **Convert the aliases/distribution-list data to LDIF format.**

   Use the `imldifsync` command to generate formatted user data (`LDIF`) by issuing the following command:

```
# /opt/SUNWmail/sbin/imldifsync -c groups.conf > /tmp/groups.ldif
```

9. **Populate the directory with the aliases/distribution-list LDIF formatted data.**

   Use the `ldapmodify` command to add the new entries to the directory:

```
# /opt/SUNWmail/bin/ldapmodify -D bind-DN -w ldap-passwd -f /tmp/
groups.ldif
```

   Refer to the bulleted section above for `bind-DN` and `ldap-password` information.

---

**Note –** If you already have some entries in the Directory database you should specify the `-c` argument to `ldapmodify` in addition to those above, so that ldapmodify will continue to the new entries. ldapmodify will otherwise exit if it tries to add an entry that is already in the Directory.

---

# General Properties Configuration

The General Properties section allows you to configure the following directory server properties. To change these properties, go to the General Properties section of the Sun Directory Services Property Book.

AdminConsole>SUN Directory Services>General Properties

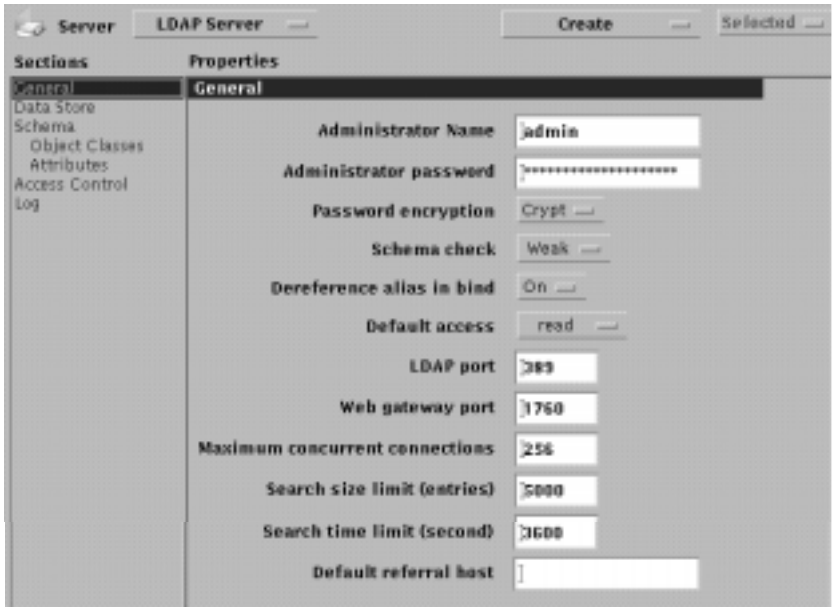



**FIGURE 6-2** General Properties Section.

- The name and password of the administrator, and how the password is stored. (Default: None. Must be specified. See “To Specify an Administrator Name, Password, and Distinguished Name” on page 166.)
- Passwords encryption. (Default: Password stored in encrypted format.)
- Whether the schema is checked when directory information is added or modified. (Default: Weak.) There are three levels of checking:
  - Off: no checking is performed
  - Weak: a check is performed when entries are created or modified
  - Strong: in addition to the previous level, a check is performed on search operations
- Whether alias entries are to be dereferenced when a bind request is received, (Default: On.)
- Default access, that is, the level of access granted to entries and attributes for which access control is not specifically defined. See CODE EXAMPLE 1-1 on page 39 for more detail

- The ports used by LDAP and the email administrator's configuration interface. (Defaults: 389 and 1760 respectively). If you change the LDAP port, the IMTA directory synchronization will not work unless you perform the following additional step:
  - Change the parameter `IMTA_LDAP_SERVER` in `/etc/opt/SUNWmail/imta/imta_tailor` to the new LDAP port value for this particular server. For example, if the LDAP port number is changed from 389 to 390, then change the entry from `xxx.eng.sun.com:389` to `xxx.eng.sun.com:390`.
- The maximum number of concurrent connections the server can accept. (Default: 256)
- Search limits, time and number of entries. (Defaults: 5000 entries or 3600 seconds.) Search stops when the first of these limits is reached. To bypass these limits, bind to the Directory Server using the admin DN and password.)
- The default directory server for referrals. Give the name of a server that holds knowledge information on subtrees not managed by the current server. (Default: No knowledge references to other servers.)

# Configuring the Data Store



A *data store* is simply the storage location of the directory information. Directory information is organized hierarchically, with entries organized in a *directory information tree*. An entry is identified by its *distinguished name* (DN) which is a unique key into the database, composed of a sequence of attributes and values which specify the hierarchical location of the entry within the DIT (example, DN: locality=Boston, organizationName=XYZ, country=US). A *naming context* is a subtree of the directory and is identified by the DN of the subtree. SIMS uses two naming contexts (see Appendix D, "SIMS Directory Schema and Directory Information Tree" for details.)

A data store can contain up to three naming contexts. The DN of these naming contexts are used to identify the data store. The distinguished name of a naming context, or of a subtree of the naming context corresponds to the base DN you specify when you perform an LDAP operation such as a search, or when you configure an LDAP client application.

A data store can contain a mixture of master and replica (slave) naming contexts, some or all of which can be replicated to other servers.

To configure a data store, you must specify the distinguished name of the naming context stored, and the name of the directory where the database files reside. Optional configuration information includes:

- Which attributes are indexed (see“Indexing the Data Store” on page 194 for information about indexing)
- Congestion thresholds (see “To Create a Data Store” on page 192 for information about the congestion monitoring system and setting thresholds)
- Cache size (see “To Create a Data Store” on page 192 for information about caching)
- Naming contexts stored (see “Configuring the Data Store” on page 190)
- Whether any of the naming contexts held in the data store are replicated to other servers (see “Replication” on page 197)

“To Create a Data Store” on page 192 gives step-by-step instructions for creating a new data store. “To Modify a Data Store” on page 194 explains how to modify an existing data store.

## ▼ To Create a Data Store

AdminConsole>SUN Directory Services>LDAP Server property book>Create pulldown>Ldbm DataStore

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, choose Ldbm DataStore from the Create pulldown.**

   The Create LDBM Data Store window is displayed, as shown in FIGURE 6-3.

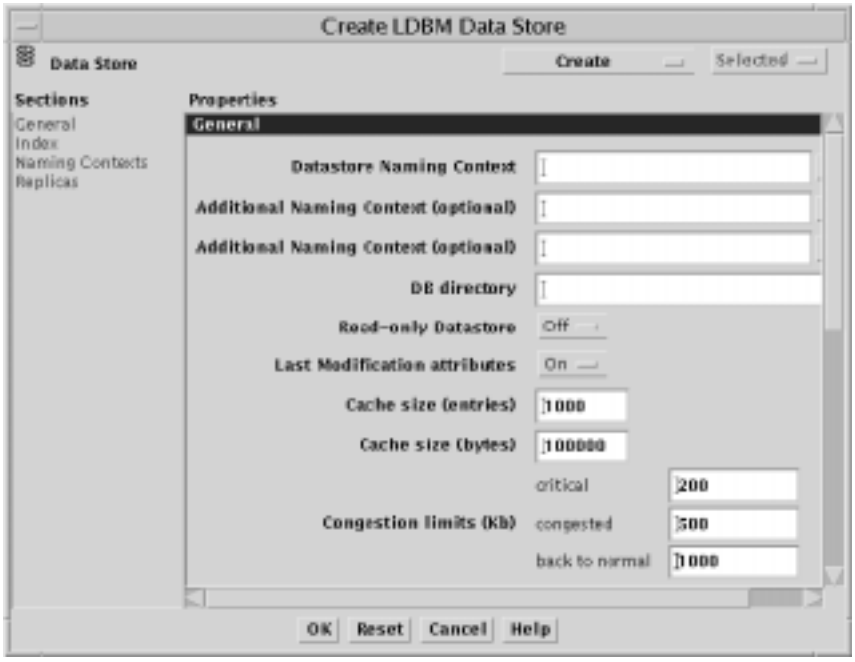



**FIGURE 6-3** Create LDBM Data Store Window

2. **Specify the distinguished names of the naming contexts stored in this data store.**
3. **Specify the path name to the directory where this database is to be stored.**
4. **Specify whether the last modification attributes should be recorded.**

   When this option is enabled, the creator's DN, a creation timestamp, the modifier's DN, and a modification timestamp are added in the entry. This enables you to search the directory for entries created or modified since a particular date or time. This can be useful if you are synchronizing changes across data stores, or you are using partial replication.

5. **Specify the cache size limit, in entries and in bytes.**

   As information is retrieved from the directory it is saved in the cache. When the cache is full, the oldest entry is discarded to make room for new information. Retrieving information from the cache is faster than retrieving entries from the database, but a large cache occupies more memory. Default values for cache size and number of entries are 100,000 bytes and 1000 entries.

6. **Specify congestion thresholds.**

   Congestion thresholds ensure that the directory does not become overloaded, by preventing new operations from starting when there are insufficient resources.

   *Critical* specifies the disk space limit at which only search, read, and delete operations are allowed, and add, modify, and modrdn operations are not permitted. The default limit is 200 Kbytes.

   *Congested* specifies the disk space limit at which add operations are no longer permitted, though modify, modify RDN (modrdn), search, read, and delete operations are allowed. The default limit is 500 Kbytes.

   *Back-to-normal* specifies amount of disk space which must be available before the *congested* and *critical* restrictions are lifted. The default limit is 1000 Kbytes.

   Threshold values are given in the number of Kbytes free on the disk holding the data store. The default location for the data store is `/var/opt/SUNWmail/ldap/dbm`.

7. **(Optional) Specify which attributes to index in the database.**

   See “To Create or Modify Indexes” on page 196” for details.

8. **In the Naming Contexts section, specify the master and slave subtrees held in this data store.**

   These are naming contexts that are subtrees or objects within the naming context used to name the data store. If you do not specify any naming contexts, the distinguished name of the data store itself is added to the list of master naming contexts automatically, but it is not displayed until you save the naming contexts information. See “Configuring the Data Store” on page 190 for details.

   a. **Choose Naming Context from the Create menu.**

   b. **Specify the subtree type (subtree or object).**

   c. **Specify the DN of the subtree or object in the Suffix field.**

   d. **Select the mode in the Mode menu.**

      If the naming context is a replica (slave), specify the name of the server from which it is replicated in the Referral field, and specify the DN that `slurpd` will use when binding to replicate changes from the master.

   e. **Click OK to save the naming contexts information.**

9. **(Optional) Replicate any of the stored naming contexts.**

   See “To Create or Modify Replicas” on page 197 for details.

10. **Click OK in the Create LDBM Data Store window to save the Data Store definition.**

11. **Add a directory entry for the root of the data store.**

    See “Adding Entries” on page 94,” for details of how to add an entry to the directory. You cannot add any entries to this data store until this root entry exists.

## ▼ To Modify a Data Store

AdminConsole>SUN Directory Services>LDAP Server property book>Data Store section>data store to modify>Selected pulldown>Modify Data Store

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, then click on the Data Store section.**

2. **Select the data store you want to modify from the Data Store list.**

3. **Choose Modify Data Store from the Selected pulldown.**

   The data store’s property book is displayed, showing the current configuration. You can modify any part of the data store configuration, apart from the distinguished name of the data store naming context. If the data store is empty, you can also modify the distinguished name of the data store naming context. See “To Create a Data Store” on page 192.

# Indexing the Data Store



A data store can contain a number of attribute indexes to help optimize the speed of access to directory information. An attribute index is a list of entries containing a given attribute or attribute value. You can index attributes using any of the following matching rules:

- Equality – Optimizes direct access to entries where an exact attribute value is supplied.
- Present – Optimizes searches with filters specifying the presence of an attribute but no specific value (cn=*, for example).

- Substring – Optimizes searches with filters containing a partially-specified attribute value (cn=ada*, for example). Substring indexing uses the first three letters of an attribute.
- Approximate – Optimizes searches with approximate match filters. The method used in approximate indexing is to discard vowels.

In a data store having the default characteristics, the following attributes are indexed:

- commonName, surname, mail, mailHost, and givenName are indexed by presence, equality, approximate match and substring match
- uid is indexed by presence and equality
- preferredRfc822Recipient, rfc822Mailbox, cCMailAddress, pROFSAddresses, mSMailAddresses are indexed by presence and equality
- employeenumber is indexed by equality.
- alias and nickname are indexed by presence, equality, approximate match and substring match.

The advantage of indexing is that it optimizes access for indexed attributes. The disadvantages are that it uses more disk space, and that adding and modifying entries takes longer.

When you add or modify an entry after an index has been created, the index is automatically updated. However, if you create a new index and the data store already contains entries, those entries are not automatically included in the index. Indexes are not automatically updated when entries are removed from the directory, so the size of the index files does not reduce as entries are removed.

To update all the indexes defined in a data store, choose Refresh Index from the Ldbm Data Store menu of the data store's property book (see "Maintaining the Data Store Attribute Indexes" on page 241). Regenerating the indexes for a data store can take several minutes, depending on the number and complexity of the indexes defined. For example, regenerating the default indexes for a data store of 20,000 entries takes approximately five minutes.

---

**Note –** Whenever you change the index directives in `/etc/opt/SUNWconn/ldap/current/slapd.conf`, you must regenerate indexes using `idxgen` or the Graphical Administrative Interface, even if the indexed attributes have not yet been added to the database. Failure to do this may result in searches for existing, indexed, attributes to fail.

---

To regenerate indexes in a shell or script, you may use the `idxgen(1M)` command. The Directory Server should not be running when you are regenerating indexes.

```
/opt/SUNWconn/ldap/sbin/idxgen /var/opt/SUNWconn/ldap/current/dbm
```

## ▼ To Create or Modify Indexes

AdminConsole>SUN Directory Services>LDAP Server property book>Index

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book.**
2. **Whether you are creating a new data store or modifying an existing one, click Index in the Sections list.**

   New: Create/Ldbm Data Store->Index

   Existing: Data Store Section->double-click data store to modify->Index
3. **To add new attributes to be indexed in the database:**
   a. **Choose Index from the Create menu**
   b. **Specify the name of the indexed attribute and the index types**

      You can specify several attributes separated by commas. These indexed attributes will have the same index types.
   c. **Click Add to add the index.**
4. **To modify existing indexing attributes:**
   a. **Double-click attribute to modify.**
   b. **Check the parameters on which to index and click Modify.**

# Replication



Any naming context held in the data store, including replica naming contexts, can be replicated to another server. By replicating a replica naming context, you create a cascading replication.

In defining the information you want to replicate, you can choose:

- A subtree: all the entries in the subtree are replicated
- An individual entry (object)

You can also define a replication synchronization schedule. This schedule determines when all updates are sent to replicas. There are three types of synchronization:

- *Immediate*, which means that the replication daemon, `slurpd`, runs permanently and sends updates to the replica immediately when modifications are made in the master.
- *Delayed*, which means that modifications are logged until the next time `slurpd` runs. If you select Delayed synchronization, specify a schedule for `slurpd`.
- *None*, which means that modifications are not automatically sent to the replica.

You can use the Synchronize control to send any outstanding modification immediately to a replica. Setting the synchronization type to *None* and using the Synchronize control to initiate replication updates manually can be useful where the update traffic is unpredictable, or where the remote server is connected by a dial-up line.

## ▼ To Create or Modify Replicas

When you create a naming context on server A and specify that it must be replicated on server B, you must create the corresponding slave naming context on server B and declare server A as the referral host in the Referral field.

| |
|---|
| AdminConsole>SUN Directory Services>LDAP Server property book |

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book.**

2. **Whether you are creating a new data store or modifying an existing one, click Create Replica in the Ldbm Data Store Property Book.**

   New: Create/Ldbm Data Store->Create/Replica

   Existing: Data Store Section->double-click data store to modify->Create/Replica

3. **To replicate any of the stored naming contexts, choose Replica from the Create menu.**

   The Add Replica dialog box is displayed, as shown in FIGURE 6-4.


**FIGURE 6-4** Add Replica Dialog Box

   a. **Specify the type of replica (subtree or object).**

   b. **In the Suffix field, specify the distinguished name of the subtree or object to be replicated.**

      To create a replica of the whole data store, specify the DN of the naming context used to identify the data store.

   c. **Select the attributes to be replicated.**

      You can specify that all attributes are replicated, or you can exclude or include certain attributes. If you choose Exclude or Include from the Attributes pop-up menu, specify the particular attributes you want to exclude from or include in the replica.

   d. **Specify the name of the destination host where the replica will be stored.**

      On the destination host, you must create this naming context and specify that it is a replica (slave). Follow the instructions in this section to configure the data store on the destination host.

   e. **Specify the distinguished name and password that the master will supply when requesting authentication.**

f. **Specify the port of the `slapd` server on the replica server to be used by the replication server, `slurpd`.**

g. **Click OK to save the replica definition and exit from the Add replica window.**

4. **Optionally, you can set up a replication synchronization schedule.**

   See "To Set Up a Replication Synchronization Schedule" on page 199."

### ▼ To Set Up a Replication Synchronization Schedule

AdminConsole>SUN Directory Services>LDAP Server property book>Data Store

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book.**
2. **Click the Data Store section.**
3. **Select the type of replication synchronization.**

   See "Replication" on page 197 for details.

4. **Click Apply to save your changes.**

## Initializing Replication

After you have configured a replica naming context, the master and replica data stores must be in the same state, so that the replica can receive replication updates from the master. If the master data store already contains entries, the Admin Console displays a dialog box giving you the option of populating the replica. Use this facility to populate the replica automatically with the entries that the master contains.

---

**Note –** Although you can start the replication process from the Admin Console, the Admin Console does not control the process and does not display error messages. You need to check the replication log file, `slurpd.log`, to ensure that the replication process has completed successfully.

---

Alternatively, you can use `slapdrepl(1M)` to create an initial replication file and populate the replica using `slurpd`. For information on these commands, see the *Sun Internet Mail Server Reference Guide*.

## Configuring Replication for XYZ Corporation

"Example: Replication in the XYZ Corporation" on page 46 explained the replication strategy designed for the XYZ Corporation. This section explains how to configure the eursales server, which holds the following information:

- The master copy of the ou=Euro-Sales, o=XYZ, c=US naming context
  This naming context is replicated to the eursale2 server.
- A replica of the l=Boston, o=XYZ, c=US naming context
  This naming context is in turn replicated to the eursale2, london, and paris servers.

FIGURE 6-5 shows the naming contexts and replicas configured for the eursales server.

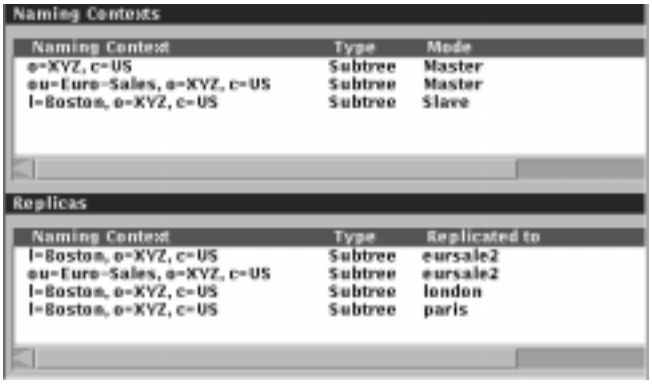



**FIGURE 6-5** Naming Contexts and Replicas Configuration Example

# Modifying the Schema



The schema is the set of rules that describe the data that can be stored in the directory. It defines the type of entries, their structure and their syntax. The schema can be modified, though certain objects and attributes cannot be changed.

To view the schema definition, click Schema in the Directory Service property book Sections list. Two tables of schema elements are displayed:

- directoryObject classes, as shown in FIGURE 6-6.

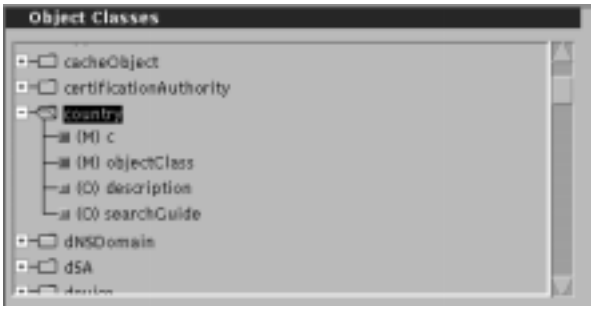



**FIGURE 6-6** Object Classes

- Attributes, as shown in FIGURE 6-7.

Click the folder icon for an object class to display its mandatory (M) and optional (O) attributes.

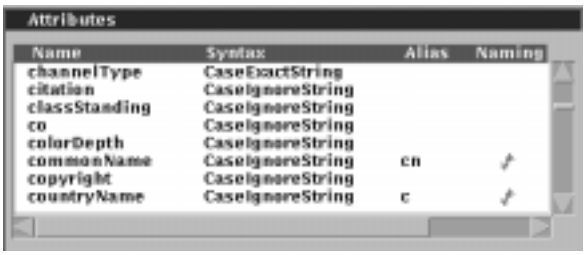



**FIGURE 6-7** Attributes

This four-column attribute table shows:

- The name of the attribute
- The attribute syntax
- Alias names for this attribute
- Whether the attribute is a naming attribute (that is, an attribute that can be used in the distinguished name of an entry).

You can modify the schema by creating new object classes or attributes, or by modifying object classes and attributes. Deleting object classes or attributes is not advisable since there might be directory entries that use the existing definitions.

**Note –** There is no automatic check to make sure that schema modifications do not invalidate entries. Therefore, to minimize the risk of entries becoming invalid, restrict your changes to addition or modification of object classes or attributes.

The schema definition contains certain information that must not be changed because it is required by components of SIMS. However, the remaining part of the schema can be modified. The Admin Console does not permit you to modify the fixed part of the schema definition. The elements that you cannot modify are marked with the keyword `frozen` in the configuration files. You *must not* remove this keyword from any standard schema item. However, if you change the schema, you can add the `frozen` keyword to any new items you want to protect.

## ▼ To Create a New Attribute

AdminConsole>SUN Directory Services>LDAP Server property book>Schema>Create pulldown>Attribute

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, and then click Schema in the Sections list.**
2. **Choose Attribute from the Create pulldown.**

   The Create Attribute window is displayed, as shown in FIGURE 6-8.


**FIGURE 6-8** Create Attribute Window

3. **Specify the name of the attribute.**
4. **If the attribute is known by another name, specify this in the Alias field.**
5. **Choose the attribute syntax from the Syntax menu.**
6. **Specify whether the new attribute can be used as a naming attribute.**
7. **Click OK to save the new attribute definition.**

   This change will take effect when you restart the `slapd` daemon.

## ▼ To Add an Attribute to an Object Class

AdminConsole>SUN Directory Services>LDAP Server property book>Schema

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, and then click Schema in the Sections list.**
2. **Select the object class to which you want to add an attribute.**
3. **Choose Modify Class from the Selected menu.**

   The Object Classes window is displayed, with the name of the object class you are modifying indicated in the General Properties section, as shown in FIGURE 6-9.

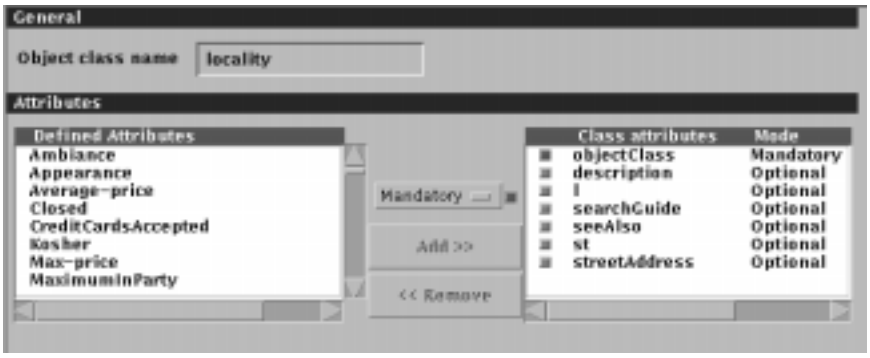



**FIGURE 6-9** Defining an Object Class

4. **Select the attribute you want to add from the Defined Attributes list.**
5. **Choose the mode of the attribute (Mandatory or Optional) from the pop-up menu.**
6. **Click Add to add the attribute to the object class definition.**

7. **Click OK to save the modified object class definition.**

   This change will take effect when you restart the `slapd` daemon.

## ▼ To Change the Mode of an Attribute

AdminConsole>SUN Directory Services>LDAP Server property book>Schema

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, and then click Schema in the Sections list.**
2. **Select the object class to which you want to change an attribute.**
3. **Choose Modify Class from the Selected menu.**
4. **Select the attribute in the Class attributes list.**

   Change the mode to Mandatory or Optional using the pop-up menu.
5. **Click OK.**

## ▼ To Create a New Object Class

AdminConsole>SUN Directory Services>LDAP Server property book>Create pulldown>Object Class

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, and then click choose Object Class from the Create menu.**

   The Create Object Class window is displayed, as shown in FIGURE 6-9.
2. **Specify the name of the new object class.**
3. **Specify the mandatory and optional attributes you want to include in this class:**
   a. **Select the attribute you want to include.**
   b. **Select the mode of the attribute (Mandatory or Optional) from the pop-up menu.**
   c. **Click Add to add the attribute to the object class definition.**
4. **Click OK to save the modified object class definition.**

   This change will take effect when you restart the `slapd` daemon.

**Note –** You cannot create new attributes in the schema while you are creating a new Object Class. Therefore you should add any new Attributes before attempting to create a new Object Class that will contain the new Attributes.

# Configuring Access Control



Access controls determine who has access to a given directory entry, and what level of access is granted. “Access Control” on page 37 explains how to design an access control policy for your directory. The following sections explain how to add, modify, and delete access control rules using the Admin Console.

An access control rule defines the level of access to specific directory information given to a particular user. There are two stages to defining a new access control rule:

- Specify the directory information to which the rule applies. This is the information that you want to protect.
- Specify the level of access granted to each user for this information.

Access control rules are ordered, with the most specific rules first, followed by more general rules. The first rule in the list that matches the requested operation is applied, the following rules in the list are ignored.

## ▼ To Add an Access Control Rule

AdminConsole>SUN Directory Services>LDAP Server property book>Create pulldown>Access Control

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, and then choose Access Control from the Create pulldown.**

   The Create Access Control Rule window is displayed, as shown in FIGURE 6-10.


**FIGURE 6-10** Create Access Control Rule Window

2. **Specify the method of selecting information to which the new rule will apply, by doing one of the following:**

   a. **From the Selected Entries menu, select the method of specifying the entries, or choose All entries.**

      You can specify entries using a DN-based regular expression, an LDAP filter, the presence of a particular attribute, or you can specify that the rule applies to all entries.

      If you want to protect only certain attributes within the set of entries defined by the regular expression, click the Attribute Set button and select the attributes to be protected. If you do not specify any attributes, all attributes in the specified entries are protected.

      i. **If you selected DN-based regular expression, type the regular expression in the Distinguished name field, or click Set to use the Distinguished Name Editor to specify the regular expression.**

         See “Using the Distinguished Name Editor” on page 211 for more details.

      ii. **If you selected LDAP filter, click the LDAP filter Set button to launch the LDAP Filter Editor. Specify the filter, and click Apply.**

         See “Specifying an LDAP Filter” on page 212 for more information.

   b. **Type the name of an attribute to be protected in the Attributes field.**

      To see a list of attributes, click the Set button. You can specify any number of attributes.

3. **Choose Access Rule from the Create menu.**

   The Add Access Rule window is displayed, as shown in FIGURE 6-11.


**FIGURE 6-11** Add Access Rule Window

4. **Choose the Rule type.**

   This defines the set of users to which the rule applies. You can specify a rule for Everyone, DN-based Regular Expression, Self (that is, the entity described by the entry), Address, Domain, or Member Attribute.

   a. **If you selected DN-based Regular Expression, specify the regular expression for the set of users to which the rule applies. The rule will apply to all users who bind with a distinguished name that matches the regular expression.**

      You can type the distinguished name directly in the field, or you can click Set to use the Distinguished Name editor to construct the distinguished name. See “Using the Distinguished Name Editor” on page 211 for more information about how to specify a distinguished name.

   b. **If you selected Address, specify an IP address.**

      The IP address can contain wildcards. The rule will apply to all users who bind from the specified IP address.

   c. **If you selected Domain, specify a domain name.**

      The domain name can contain wildcards. The rule will apply to all users who bind from the specified domain.

   d. **If you selected Member Attribute, specify an attribute.**

      The rule will allow to add or remove the distinguished name of the user to the list of members specified by the attribute.

   e. **apply to all users whose directory entries contain this attribute.**

5. **Specify the access rights to be granted to the specified set of users.**

6. **Click Add to add the rule.**

   You can then define other rules for entries you have selected, as described in Step 4. When you have created all the rules for these entries, click Cancel to remove the Add User Rule window.

7. **In the Create Access Control Rule window, click Add to store the new rules.**

   You can now select another set of entries and define access controls for them, as described in Step 2.

   Configuration changes are implemented when you restart the `slapd` daemon.

## ▼ To Modify an Access Control Rule

| AdminConsole>SUN Directory Services>LDAP Server property book>Create pulldown>Access Control |
|---|

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, and then choose Access Control from the Create pulldown.**

2. **Select the set of entries whose access control you want to modify, and choose Modify ACL from the Selected menu.**

   The Access Control property book is displayed, as shown in FIGURE 6-12.

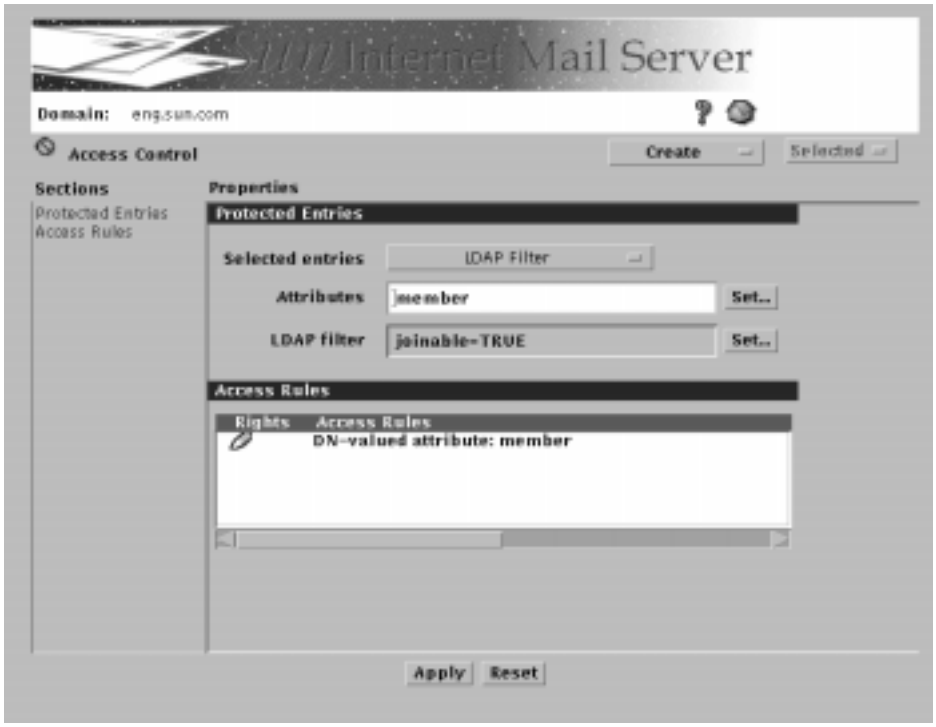



**FIGURE 6-12** Access Control Property Book

Tip - If you double-click a rule, the Access Control property book is displayed automatically.

3. **Select the rule that you want to modify, and choose Modify Access Rule from the Selected menu.**

   The Modify User Rule window is displayed.

   If you double-click the rule you want to modify, the Modify User Rule window is displayed automatically.

4. **Make the modification you require.**

5. **Click OK.**

   Make any other modifications you require. When you have made all the modifications, click Cancel to remove the Modify User Rule window.

   These changes will take effect when you restart the `slapd` daemon.

## ▼ Delete an Access Control Rule

AdminConsole>SUN Directory Services>LDAP Server property book>Create pulldown>Access Control

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, and then choose Access Control from the Create pulldown.**
2. **Select the set of entries and choose Modify ACL from the Selected menu.**
3. **The Access Control property book is displayed.**
4. **Select the rule you want to delete and choose Delete Access Rule from the Selected menu.**

   To delete all access control rules for a set of entries, select the entry set and choose Delete ACL from the Selected menu.

   You are prompted to confirm that you want to delete all access controls for the set of entries.
5. **Click Apply.**

## ▼ Reordering Access Control Rules

AdminConsole>SUN Directory Services>LDAP Server property book>Create pulldown>Access Control

1. **In the Admin Console home page, click the Sun Directory Services icon to bring up the LDAP Server property book, and then choose Access Control from the Create pulldown.**
2. **Select the rule you want to move.**
3. **Choose Move Up or Move Down from the Selected menu.**
4. **Click Apply to save the changes to the configuration file.**

   The changes are implemented when you restart the `slapd` daemon.

---

**Note –** The Admin Console will prevent you from breaking the convention of placing the rules from the more specific to the more general.

---

# Using the Distinguished Name Editor

A distinguished name (DN) is a sequence of relative distinguished names (RDNs), separated by a comma, for example o=XYZ, c=US. When you have to specify a DN in the Admin Console, you can type it directly into the field supplied, or you can construct it using the Distinguished Name Editor.

To start the DN Editor, from the LDAP Property Book click:

| AdminConsole>LDAP Property Book>SUN Directory Services>Create pulldown>Access Control>Selected entries>DN-based regular expression>Distinguished name>Set |
|---|

The DN Editor dialog box is displayed, as shown in FIGURE 6-13.


**FIGURE 6-13** Distinguished Name Editor

To edit a distinguished name, use the Previous and Next buttons to position the cursor where you want to insert an RDN, or select the existing RDN you want to replace. Select the attribute type for the RDN, and type the value in the RDN value field. If you are creating a new DN, click the Add RDN button. If you are modifying an existing RDN, this button is replaced by a Modify RDN button. Click Apply to save the new DN, and click Cancel to close the DN Editor dialog box.

You can specify a DN that contains a regular expression, to indicate a set of entries. This is useful when configuring access control, for example, but not when specifying a naming context. The Admin Console does not prevent you from entering a regular expression in any DN, but you should use wildcards only where it is appropriate.

## Regular Expressions

You can specify a set of entries using a regular expression. See the `regex(1F)` manpage for information about regular expressions.

You can specify a regular expression for the distinguished name of an entry. For example, the regular expression dn="cn=Joe Smith, ou=.*, o=XYZ, c=US" specifies the set of entries for people called Joe Smith in the whole of the XYZ Corporation. DN-based regular expressions are useful when defining access controls.

You can also use a DN-based regular expression to specify a set of values for an attribute whose values are DNs. For example, you can grant write access to a distribution list entry to any person whose DN is a value of the member attribute, using the regular expression member="dn=.*".

## Specifying an LDAP Filter

An LDAP filter is a way of specifying a set of entries, based on the presence of a particular attribute or attribute value. You can use an LDAP filter in an access control rule. For example, the default access control rules include a filter specifying that users can add their own Distinguished Names to the member attribute of any entry that contains the attribute joinable with a value of TRUE. This allows users to add or remove their names from distribution lists.

The Admin Console includes a Filter Editor for building or modifying filters. To start the Filter Editor:

| AdminConsole>SUN Directory Services>Create pulldown>Access Control>Selected Entries>LDAP Filter>Set |
|---|

The Filter Editor dialog box is displayed, as shown in FIGURE 6-14.

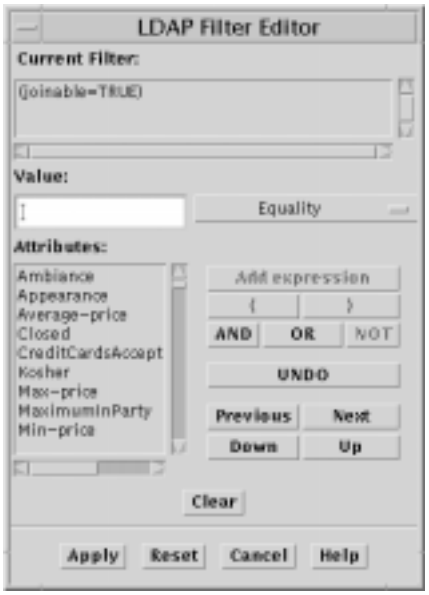



**FIGURE 6-14** Filter Editor Dialog Box

The Current Filter field shows the filter you are modifying, or the current state of the filter you are creating. To add an expression to a filter:

1. **Select the attribute from the list displayed.**
2. **Type a value in the Value field.**
3. **Select a match type from the pull-down menu.**
4. **Click AND, OR, or NOT, to indicate how this expression is used in the filter.**
5. **Click Add Expression to add the expression to the filter.**

# Configuring Logging

You can log the following information:

- Trace information for `slapd` function calls
- Debug information about packets
- Protocol trace information with extra debug information

- Connection management
- Packets received and sent
- Filter processing during search operations
- Processing of the configuration file
- Access control processing
- Starting for connections and operations
- Entry parsing debug information

By default, the server logs statistics for connections and operations. If you want to log other information, check the boxes on Logging Properties section in the configuration tool, as shown in FIGURE 6-15.

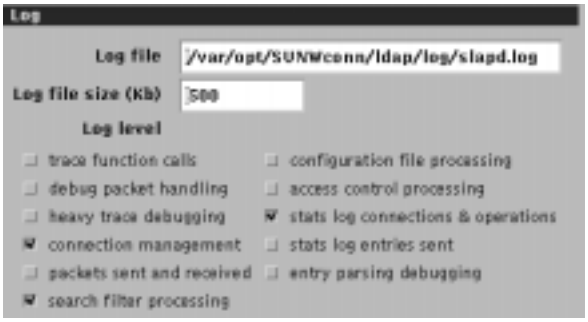



**FIGURE 6-15** Logging Properties Section

See “Directory Service Logging” on page 262 for information about reading the log files.

# SMCS Directory Synchronization

If you have licensed and installed the Sun Message Connectivity Service option, you will have an SMCS Directory Sync section in the Sun Directory Services property book. For more details, please see the *Sun Messaging Connectivity Services Channel Guides*

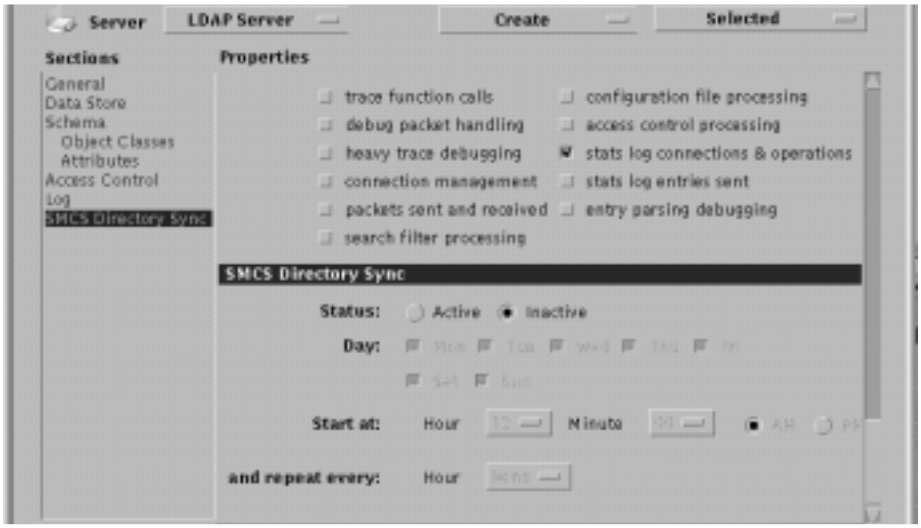



**FIGURE 6-16** SMCS Directory Sync

# Web Access to the Directory



The email administrator's configuration interface provides an interface to an LDAP directory from any web browser. You can use this interface to browse the directory, to search for and read entries, and to modify some directory information. This is

useful for checking information in the directory, but it is a general-purpose tool and should not be used in place of the User Manager. For more information, refer to Chapter 3, “User Management.”

The gateway daemon, `web500gw`, requires the `slapd` daemon to be running on the same machine. Before you start the LDAP directory browser, make sure that the `slapd` and `web500gw` daemons are running. If not, start them as described in “To Start the Directory Services” on page 244.

## ▼ To Access the Email Administrator’s Configuration Interface

1. **From any machine, point a web browser at: http://<*server*>:<*web500gwport*>/**

   where *server* is the name of a server running the `slapd` and `web500gw` daemons and *web500gwport* is the port used by `web500gw`. The default port is 1760.

   The “Top of the Directory” page is displayed. It shows the top entry in the data store to which you are connected. If the data store holds several naming contexts, several entries are listed, sorted by object class. Each entry is a hot link to a search facility that enables you to browse the naming context.

   ---

   **Note –** If entries are listed under the heading Unknown, you have probably configured a naming context without creating the corresponding entry in the database. See “To Create the Root Entry for XYZ Corporation” on page 93, for information on adding a top entry to a data store.

   ---

2. **Click a hot link for a naming context.**

   Your browser will display a search page. The search page shows the name of the entry that is the top of the naming context.

3. **Use the View All Attributes and View Main Attributes controls to display respectively all the attributes and their values, or a subset of attributes.**

   See “Configuring the Email Administrator’s Configuration Interface” on page 218 for information about configuring the attributes displayed and other aspects of the email administrator’s configuration interface.

## ▼ To Browse the Directory

1. **Specify the top of the subtree you want to browse.**

   Use the Move upwards control to move up the directory tree hierarchy. At each position, the naming context list shows you the parent naming context and any child naming contexts. Choose the naming context you want to browse. The top entry in that naming context becomes the currently selected entry.

2. **Click the Browse button.**

   A list of the entries in the naming context is displayed. The attribute used to identify the entries is configurable. To see the content of any entry, click on the attribute value listed.

## ▼ To Search the Directory

1. **Specify the top of the subtree you want to browse.**

   Use the Move upwards control to move up the directory tree hierarchy. At each position, the naming context list shows you the parent naming context and any child naming contexts. Choose the naming context you want to browse. The top entry in that naming context becomes the currently selected entry.

2. **Type the search string in the Subtree search field and click Do Search.**

   The search string can be any attribute value, or any valid LDAP filter (see “Specifying an LDAP Filter” on page 212).

   After a short delay, a list of entries that match the search string is displayed, up to the limit configured for the gateway. Click on an entry to see the attributes it contains.

## ▼ To Modify a Directory Entry

---

**Note –** If you use the email administrator’s configuration interface to modify an entry of type emailPerson, and if you modify the password that you used to bind to the directory, when you Apply the changes, you must reread the entry before you can make any further changes to it.

---

1. **Search for the entry, as described in “To Search the Directory” on page 217.”**

2. **Click on the entry to display its attributes.**

   At the end of the attribute list there is a Modify Attributes link.

3. **Click Modify Attributes.**

   A bind request screen is displayed.

4. **Specify the distinguished name and password you want to use to bind to the directory.**

   The access controls defined for the entry will determine whether you are permitted to modify any attributes in the entry.

5. **Supply new values for the attributes you want to modify.**

   When you have finished modifying the entry, click Apply.

## Configuring the Email Administrator's Configuration Interface

The behavior and appearance of the email administrator's configuration interface is determined by information stored in several files. To configure the interface, edit the following files:

- `webldapfilter.conf`
  Controls how the interface makes a search request to the directory. See the `webldapfilter.conf(4)` man page for details.
- `webldapfriendly.conf`
  Contains user-friendly equivalents of certain attribute values that might be used in the directory. By default, it contains mappings between the ISO country codes and the names of countries.
- `web500gw.help`
  Contains help text for the user interface to the LDAP/HTTP gateway. You can edit this file and change the help text to reflect any changes you make to the user interface.
- `web500gw.helpattr`
  Contains an explanation of the directory attributes visible through the user interface to the LDAP/HTTP gateway.
- `web500gw.messages`
  Contains the messages and screen text used in the user interface to the LDAP/HTTP gateway. You can customize the user interface by changing this file.
- `webldaptemplates.conf`
  Contains templates that control how information retrieved from the directory is displayed. See the `webldaptemplates.conf(4)` man page for details.

These files are configuration files and help files for the LDAP/HTTP gateway.

The default location of `webldapfilter.conf` and `webldapfriendly` is in the directory `/etc/opt/SUNWmail/ldap/current`. All other listed files are located in the directory `/etc/opt/SUNWmail/ldap/current/locale/C`.

Detailed information on `webldapfilter.conf`, `webldapfriendly`, and `webldaptemplates` is provided respectively in the manpages `webldapfilter.conf(4)`, `webldapfriendly.conf(4)`, and `webldaptemplates.conf(4)`. Sun does not recommend editing these configuration files manually.

The files `web500gw.help`, `web500gw.helpattr`, and `web500gw.messages` are internal files, and must not be modified.

# Monitoring the Directory Service

You can view directory service statistics in five categories:

- Global
- Detailed
- Operations
- Associations
- Interactions

The statistics available are the same information that is collected by the SNMP agent for the directory.The SNMP agent for the directory does not have to be running for statistics to be collected.

## ▼ To Monitor Directory Service Statistics

| AdminConsole>SUN Directory Services>LDAP ServerMenu>Statistics> |
|---|

1. **Click the Sun Directory Services icon in the Admin Console home page.**
2. **Choose Statistics from the LDAP Server menu in the Admin Console.**

   The Statistics property book is displayed, as shown in FIGURE 6-17.

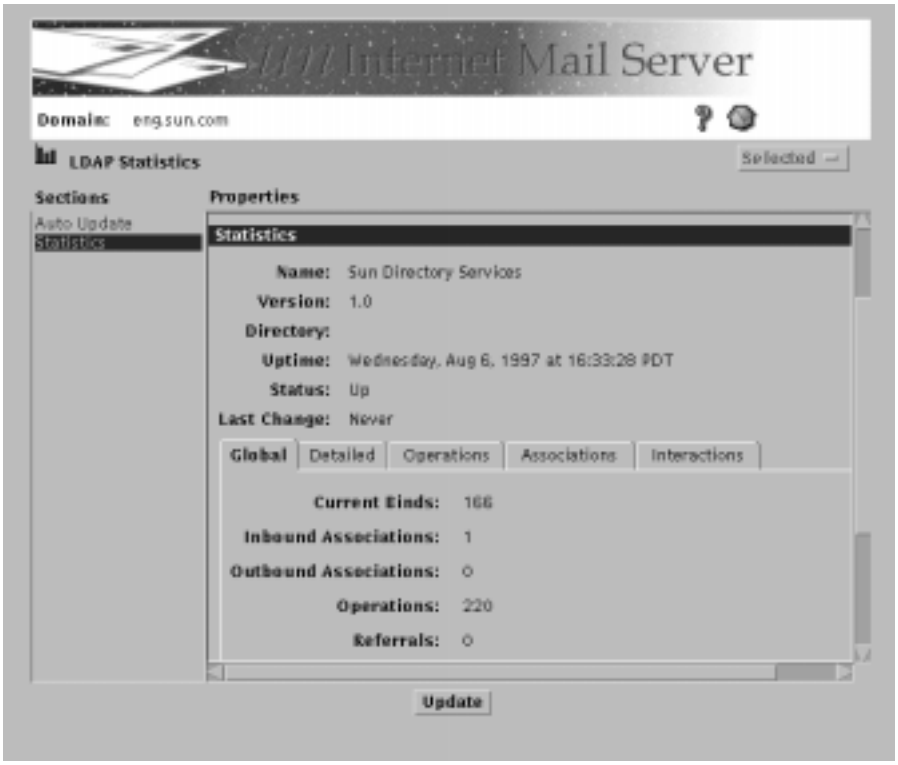



**FIGURE 6-17** Directory Service Statistics Property Book

It presents a snapshot of the statistics available.

3. **Click the tab for the category you want to view.**
4. **Click Update in the Sections list to get the latest statistics.**

   To update the statistics at regular intervals:

   a. **Click Auto Update in the Sections list**

   b. **Set the Refresh Interval field**

   c. **Click Start Auto Update**

---

**Note –** The Start Auto Update and Stop Auto Update controls apply to viewing the statistics, not to collecting the data. They only apply while the property book is open. If you close the property book, the refresh interval is reset to the default and automatic updating of the statistics view stops.

---

The Simple Network Management Protocol (SNMP) is not available from the Admin Console since SIMS does not officially support it. However, the directory service component of SIMS supports some SNMP capabilities.

If desired, you can use an SNMP management console to access the management information from the directory service. Before you use the SNMP capabilities to monitor the directory component, you must purchase and install the appropriate SNMP master (relay) agent on the same mail server that the directory SNMP subagent process is installed on.

Rather than monitoring the directory using its SNMP capabilities, Sun recommends monitoring the directory and all other SIMS components using the Administration Console.

The LDAP SNMP Agent supports the Management Information Bases (MIBs) that are part of the Mail And Directory MANagement (MADMAN) standard. In particular, the Sun Internet Mail Server 3.5 SNMP Agent conforms to the *Network Services Monitoring MIB* (RFC 1565) and the *X.500 Directory Monitoring MIB* (RFC 1567) that apply to all messaging and directory applications.

The following directory service information is monitored:

- Application information
  - Application name
  - Application directory name
  - Application version
  - Application uptime
  - Application status (*up* or *down*)
  - Last status change
  - Number of inbound associations
  - Number of outbound associations
  - Accumulated inbound associations
  - Accumulated outbound associations
  - Last inbound activity
  - Last outbound activity
  - Rejected inbound association
  - Failed inbound association

- Association information
  - The distinguished name of the remote application
  - The protocol being used to communicate
  - The type of the remote application, and whether it is an initiator or responder

    - The current duration of the association

- Directory server operations
    - Anonymous bind
    - Unauthenticated bind
    - Simple authentication bind
    - Strong authentication bind
    - Bind security errors
    - Inbound operations
    - Read operations
    - Compare operations
    - Add Entry operations
    - Delete Entry operations
    - Modify Entry operations
    - List operations
    - Search operations
    - One-level search operations
    - Whole tree search operations
    - Referrals
    - Chaining
    - Security errors
    - DSA errors

- Directory entry information
    - Master entries
    - Copy entries
    - Cached entries
    - Cache Hits
    - Slave Hits

- Interactions with other directory servers
    - Distinguished name of remote directory server
    - Time of creation of remote directory server
    - Time of last attempt to contact the remote directory server

- Time of last successful interaction with the remote directory server
- Number of failures since last successful contact
- Total number of failures to contact the remote directory server
- Total number of successful interactions with the remote directory server

CHAPTER 7

# Maintenance

This chapter describes tasks that are performed on a regular basis (either scheduled or as-needed).

**TABLE 7-1** SIMS Maintenance Tasks

| Topic/Task | Description | Page |
|---|---|---|
| Maintaining Licenses for SIMS | Procedures for obtaining/maintaining SIMS licenses. | 225 |
| IMTA Maintenance | - Adjusting Post Job Frequency<br>- Adjusting the Frequency of the Return Old Messages Program | 231 |
| Sun Message Store Maintenance | - Recommended Maintenance Schedule<br>- Message Purge<br>- Message Store Backup and Restore<br>- Folder Check<br>- Importing /var/mail Users<br>- Deleting Old Messages<br>- Delete User | 233 |
| Maintaining the Directory Service | - Maintaining the Data Store Attribute Indexes<br>- To Back Up a Data Store<br>- To Restore a Data Store<br>- Backing Up the Directory Data Base<br>- Backing Up and Restoring Directory Service Configuration<br>- To Start the Directory Services<br>- To Stop the Directory Services | 241 |

# Maintaining Licenses for SIMS

The license system software has two packages that must be installed on the same machine. They are:

- License server software (SUNWlicsw) – contains the license daemon, which maintains the product license database for applications supported by the license system. SUNWlit provides a user interface that helps you get your license and install it.
- License Installation Tool (SUNWlit) – a license server is automatically set up when you install the license server software package (SUNWlicsw) and license installation tool package (SUNWlit) on a machine and run the license installation tool.

Once you have filled out and sent your license request to the License Center (as discussed in the *SIMS Installation Manual*), and you have received your license information, use the following procedure to install your licenses.

## ▼ To Add Licenses to an Already Running License Server

If the FLEXlm daemon is running, stop it, then restart it to recognize the new licenses:

1. **Become root, then change your working directory to the license directory.**

```
% su
Password: <Enter your root password>
# cd /etc/opt/licenses
```

2. **Verify that FLEXlm is running by typing the following:**

```
# ./lmstat -a -c licenses_combined
```

You will see output similar to the following:

```
<hostname> is the hostname running a Departmental (Workgroup) product.
lmstat - Copyright (C) 1989-1994 Globetrotter Software, Inc.
Flexible License Manager status on Fri 5/9/97 16:35

License server status (License file: licenses_combined):
    <hostname>: license server UP (MASTER)

Vendor daemon status (on <hostname>):

    suntechd (v3.x): UP
    lic.SUNW (v3.x): UP

Feature usage info:

Users of solstice.mail.imta:

Users of solstice.mail.mbox:  (Total of 100 licenses available)

Users of SLAPD.1:  (Total of 1 licenses available)

  "SLAPD.1" v1.000, vendor: lic.SUNW
1 floating licenses

root <hostname> /dev/tty (v1.000000) (<hostname>/7588 109), start Thu
5/8 18:56
```

If FLEXlm is not running, skip the next step.

3. **Stop the FLEXlm Daemon**:

```
# /etc/initd/lic_mgr stop
```

4. **Update the license files by running either** `lit` **or** `lit_tty`:

```
# ./lit &
```

Refer to “To Install License Information By Hand” on page 228 or “To Install License Information From a File” on page 230 for details.

5. **Restart the FLEXlm daemon.**

```
# /etc/init.d/lic_mgr start
```

---

**Note –** If you experience problems, you can access the license manager log in the directory `/var/tmp`. The latest log file is `license.log`.

---

## ▼ To Install License Information By Hand

1. **Run `lit` to install a license password:**

```
# cd /etc/opt/licenses
# ./lit &
```

The main License Installation Tool window is displayed as shown in FIGURE 7-1.

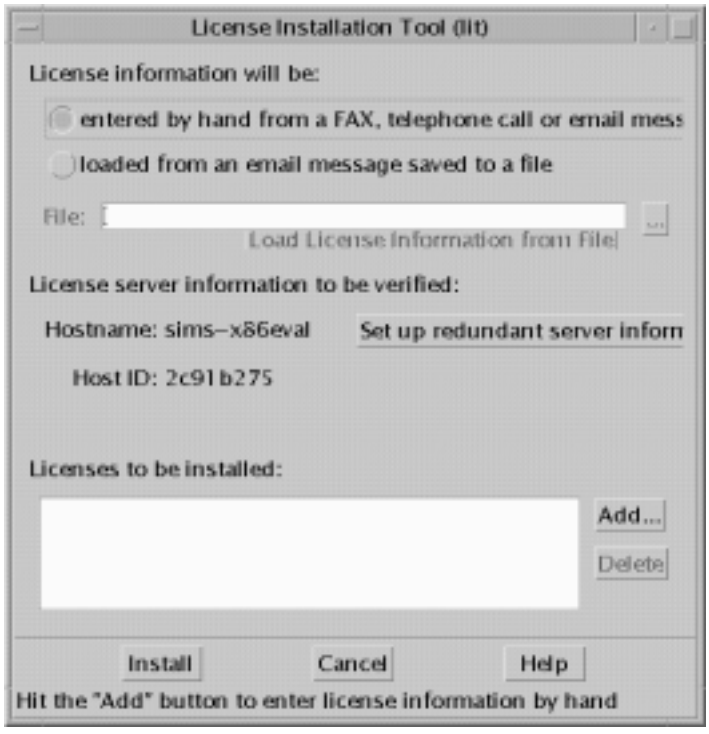



**FIGURE 7-1** License Installation Tool (lit) Window

2. **Click the button “entered by hand from a FAX, telephone call, or email message.”**

   The Add license to be installed window is displayed.

3. **Indicate the component for which you are installing a license from the feature name pull-down list (**FIGURE 7-1**).**

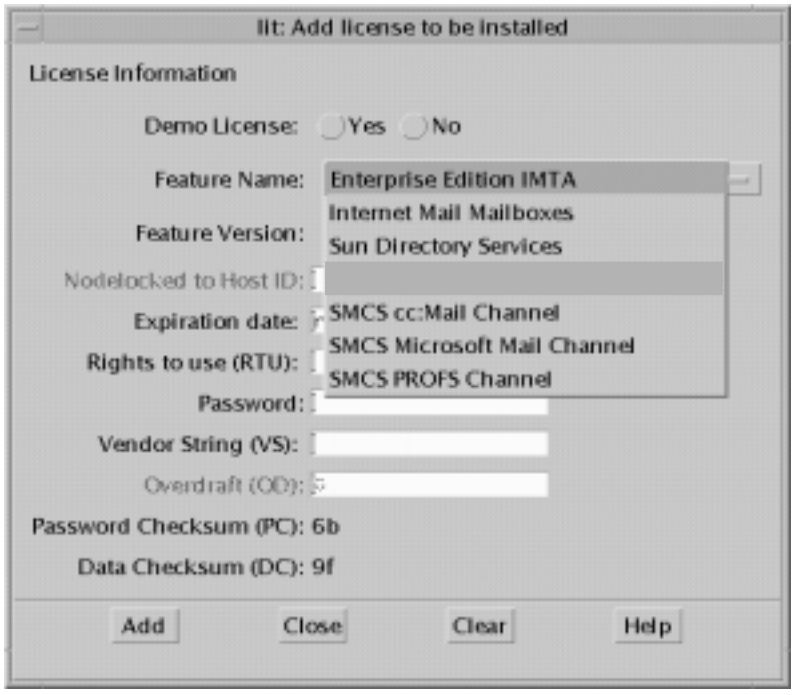



**FIGURE 7-2** Add License to be installed Window

4. **If your licenses have an expiration date, type the date in the dd-mmm-yy format in the “Expiration date” field.**

   For example, 15-aug-99. If you do not have a designated expiration date, leave this field blank.

5. **Type the number of Rights to Use you received from the Sun License Center in the RTU field.**

6. **Type the number the Sun License Center provided in the Password field.**

7. **Type in the Vendor String (VS).**

8. **Click the Add button when you have filled out this form.**

9. **The feature that you want licensed is added to your list of licenses.**

10. **Repeat Step 3—Step 8 for each component from the feature list you would like licensed.**

11. **Click Close after you have added all the licensed components to the list of licenses you want to install.**

    The LIT is redisplayed.

12. **Click on Install to install all the licenses for all the components you indicated.**

## ▼ To Install License Information From a File

1. **Run `lit` to install a license password:**

```
# cd /etc/opt/licenses
# ./lit &
```

The main License Installation Tool window is displayed as shown in FIGURE 7-3. Press *loaded from an email message saved to a file*.

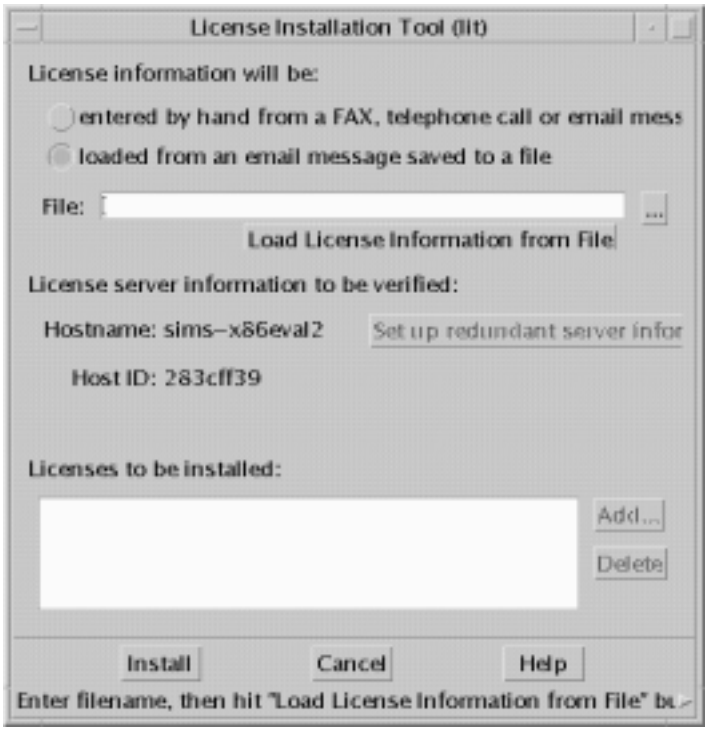



**FIGURE 7-3** License Installation Tool Window

2. **Type a path to the source file containing the password information in the *File:* field.**

   You can click the ellipsis button to find the file you saved using a file chooser.

3. **Click “Load License Information from the File” button.**

   All the components that are licensed appear in the list of licenses section of the LIT window.

4. **Click the Install button.**

   All the components listed in the LIT window are installed.

# IMTA Maintenance

## Adjusting Post Job Frequency

The IMTA runs a periodic job called `post.sh` every four hours. The `post.sh` program scans through all the channels currently defined in the configuration file and checks the corresponding queues for messages. Processing jobs are unconditionally submitted to run the master channel programs for any channels, with master programs so as to poll remote systems that cannot establish their own connections. Jobs are also submitted for channels that support master channel programs and have messages queued. After this is done `post.sh` then terminates. It will run again in another four hours.

`post.sh`, is the shell script `/opt/SUNWmail/imta/lib/post.sh`, which the `cron` daemon is normally scheduled to run every four hours. IMTA’s suggested default behavior of running the periodic delivery job once every four hours is appropriate for most sites. Indeed, at busy sites, running the periodic delivery job too frequently tends to be counterproductive.

If a site has a special need to run `post.sh` more frequently, they can change the `crontab`. Note, however, that RFC 1123, Requirements for Internet Hosts, requires that Internet mail wait at least 30 minutes before being retried. Do not run your channel to the Internet more frequently than every half hour.

Finally, IMTA normally performs some periodic clean up tasks when `post.sh` runs. IMTA’s defaults are tuned for the case where the periodic job only runs every couple of hours. If you will be running the periodic job more frequently, you should adjust IMTA’s clean up task frequency: the `IMTA_SYNCH_CACHE_PERIOD` and

IMTA_VERSION_LIMIT_PERIOD IMTA tailor options should be set to integer values so that these tasks are still performed only every couple of hours or so. (Refer to *SIMS Reference Manual* for more details on these strings.)

## Adjusting the Frequency of the Return Old Messages Program

The IMTA runs a second periodic job called `return.sh` which has as its primary job the returning of old, undeliverable messages which have sat around in the message queues for too long. `return.sh` is a shell script at `/opt/SUNWmail/imta/lib/return.sh`. The cron daemon normally schedules it to run once a day at 30 minutes after midnight.

The `return.sh` scans the channels listed in the configuration file, checking the values established with the `notices` keyword. The messages queued to each channel are then checked. A warning message is sent for every message whose age in days matches any of the values specified with the `notices` keyword on the associated channel. The default ages if no notices keyword is specified are 3, 6, 9, and 12 days. If the message is as old or older than the final notices value, the entire message is returned and the original message is deleted from the channel queue; no further attempts to deliver the message will be made. (See the *SIMS Reference Manual* for `notices` channel keyword.)

The text of the warning and failure notices issued by the message return system is contained in the pair of files `return_warning.txt` and `return_failure.txt` located in `/opt/SUNWmail/imta/locale/{C}/LC.MESSAGES` directory. These files can be edited to provide different notification text if desired.

IMTA maintains a history of delivery attempts for each message, which sometimes will include information about why the delivery attempts failed. This information is included in returned messages if `RETURN_DELIVERY_HISTORY` is set to `1` in the IMTA `Option` file (this is the default). A value of `0` disables the inclusion of this information.

`imta_tailor` file options can be used to control the periodicity of the various subfunctions of the message return system. The `IMTA_RETURN_SYNCH_PERIOD` option in `/opt/SUNWmail/imta/imta_tailor` controls queue synchronization, the `IMTA_RETURN_PERIOD` IMTA tailor file option controls the return of expired messages and the generation of warnings, and the `IMTA_RETURN_SPLIT_PERIOD` IMTA tailor file option controls splitting of the `mail.log` file. If any of these options is set to an integer value N, then the action associated with the tailor file option will only be performed every N times the message return job runs. The value of these options is taken to be 1 if the option is not specified in the IMTA tailor file.

If the IMTA return job is running once an hour, then the default will be to issue warning notices for messages which have remained undeliverable for 3, 6, or 9 hours. Message which have remained undeliverable for 12 or more hours are returned in their entirety to their sender and no further delivery attempts are made.

---

**Note –** When `RETURN_UNITS=1`, these defaults will result in mail being bounced much too soon; therefore, sites are strongly encouraged to use the `notices` channel keyword to set “bounce” ages in excess of twelve hours.

---

`return.sh` also performs various IMTA periodic cleanup tasks tuned on the assumption that the return job will only be running once a day. When `return.sh` is run more frequently, various IMTA parameters should be adjusted accordingly. In particular, the `IMTA_RETURN_SYNCH_PERIOD` and `IMTA_RETURN_SPLIT_PERIOD` IMTA tailor file options should generally be adjusted so that these tasks are still performed only once a day. See the `imta purge` and `imta cache-sync` utilities in the *SIMS Reference Guide* for details o nthe cleanup programs used.

# Sun Message Store Maintenance



The Sun Message Store contains the content of the email system—messages and attachments. This section describes the maintenance procedures for the Sun Message Store. TABLE 7-2 summarizes the maintenance utilities provided for the Sun Message Store.

**TABLE 7-2** Sun Message Store Maintenance Utilities

| Utility | Description | Supported Interface/Reference |
|---|---|---|
| Purge | Removes messages that are no longer referenced in user and group folders and returns space to the Sun Message Store file system. | Admin Console. See "Configuring Purge Options" on page 158 and "To Configure the Purge Schedule" on page 160.<br><br>Command-line utility (`impurge`). Refer to *SIMS Reference Manual*. |
| Backup | Copies contents of folders to specified backup device. Can perform full or incremental backups of all folders or the folder of a specified user or group. | Command-line utility (`imbackup`). See "Message Store Backup and Restore" on page 236 and *SIMS Reference Manual*. |
| Restore | Restores contents of all folders or one specified user or group folder from the backup device to the Sun Message Store. | Command-line utility (`imrestore`). See "Message Store Backup and Restore" on page 236 and *SIMS Reference Manual*. |
| Folder Check | Scans through the Sun Message Store and the user folders verifying links. | Command-line utility (`imcheck`). Refer to the *SIMS Reference Manual*. |
| Import mailbox | Imports existing user's mailbox in to the Sun Message Store. | Command line utility (`imimportmbox`). Refer to *SIMS Reference Manual*. |
| Delete user | Deletes the following from the Sun Message Store: Inbox, private folders, and private shared folders of a specified user; and public shared folders. | Command-line utility (`imdeluser`). Refer to "Delete User" on page 241 and *SIMS Reference Manual*. |
| Reinitialize user quota | Reinitializes a user's quota file in the user admin directory `/usr/<hash number>/<username>/Adm`. | Command-line utility (`iminitquota`). Refer to "To Activate Message Store Quota Enforcement on an Installed System" on page 149 and *SIMS Reference Manual*. |
| BackUp / Restore Message Store Configuration | Back up or restore Message Store Configuration | Admin Console. See "Sun Message Store Configuration Back Up and Restore" on page 143. |
| Deleting old messages | Deleting messages received before a specified date. | Command-line utility (`imexpire`). Refer to "Deleting Old Messages" on page 241 and *SIMS Reference Manual*. |

Some of the maintenance utilities impose a session-locking mechanism to prevent certain other maintenance utilities from being run in parallel. TABLE 7-3 outlines which utilities cannot run in parallel.

**TABLE 7-3** Maintenance Utilities Session Locking

| Utility | Purge | Backup | Restore | Import Mailbox | Delete User | Check | Quota | Delet Old Messages |
|---|---|---|---|---|---|---|---|---|
| Purge | Lock | Lock | Lock | Lock | Lock | Lock | | Lock |
| Backup | Lock | | | | Lock | | | Lock |
| Restore | Lock | | Lock | | Lock | | | Lock |
| Import mailbox | Lock | | | | Lock | | | Lock |
| Delete user | Lock | Lock | Lock | Lock | Lock | | Lock | Lock |
| Folder Check | Lock | | | | | Lock | | Lock |
| Reinitialize user quota | | | | | Lock | | Lock | |
| Delete Old messages | Lock | Lock | Lock | Lock | Lock | Lock | | Lock |

Folder Check program should not be run with any other utilities.

# Recommended Maintenance Schedule

TABLE 7-4 outlines the recommended maintenance schedule for the Sun Message Store.

**TABLE 7-4** Recommended Sun Message Store Maintenance Schedule

| Task | Frequency |
|---|---|
| Full backup | Once per week |
| Incremental backup | Daily |
| Purge | Daily |
| Folder check | At least once per week or as needed |
| Restore | As needed |

**TABLE 7-4** Recommended Sun Message Store Maintenance Schedule

| Task | Frequency |
|---|---|
| Importing user's folders and messages from `/var/mail` to the Sun Message Store | As needed |
| Delete user | As needed |
| Reinitialize user quota | As needed |

# Message Purge

When a message is delivered into the Sun Message Store, a reference is created in the Inbox of each of the message recipients. The reference points to the stored message. As each recipient reads, deletes, and removes (expunges) the message via their respective mail clients, the associated reference to the message is removed. When all references are removed, the message can be purged from the Sun Message Store.

The purge utility removes messages no longer referenced from any user or shared folder and returns the space to the Sun Message Store file system. The purge utility removes unreferenced messages starting with mail two days old and older. It does not remove unreferenced messages in today's and yesterday's mail. You must use the purge utility periodically; otherwise, the size of the Sun Message Store will grow unbounded.

For information on the other maintenance utilities with which purge can run concurrently, refer to TABLE 7-3.

You can invoke the purge utility by issuing the `impurge` command (see to *SIMS Reference Manual and* "To Configure the Purge Schedule" on page 160*)*, or you can configure purge options and a purge schedule using the Admin Console.

# Message Store Backup and Restore

Message store backup and restore allows you to:

- Restore the contents of a server's entire Sun Message Store (all referenced messages) in the event of a catastrophic failure, such as the loss of a hard disk.
- Restore group or user folders in the event that a folder is corrupted or a user accidently deletes a message.
- Migrate user or group folders or the contents of the entire Sun Message Store to another Sun Message Store.

Do backup and restore using the `imbackup` and `imrestore` commands (see *SIMS Reference Manual*), or *Solstice Backup*, a file backup and restore product that is part of the Solstice System Management Suite, which is bundled with the Solaris server or can be purchased separately (see `http://docs.sun.com` under the System Administration Library for documentation).

Each method has its advantages and disadvantages.

- Solstice Backup and Restore:
  - Easy to Use GUI.
  - Automated on-line backup and restore—backups can be scheduled.
  - Simultaneous backup and restore on more than one group of users.
  - Recognizes more types of backup devices than `imbackup`.
  - Local or remote administration.
  - Automated tape labeling and tracking, support for bar code recognition, cleaning cartridges, etc.
  - No incremental backup at this time.
- `imbackup` and `imrestore` command line utilities:
  - You can restore a single message.

---

**Note –** Message store backup and restore cannot run at the same time as some other maintenance utilities. Refer to TABLE 7-3 for session locking information.

---

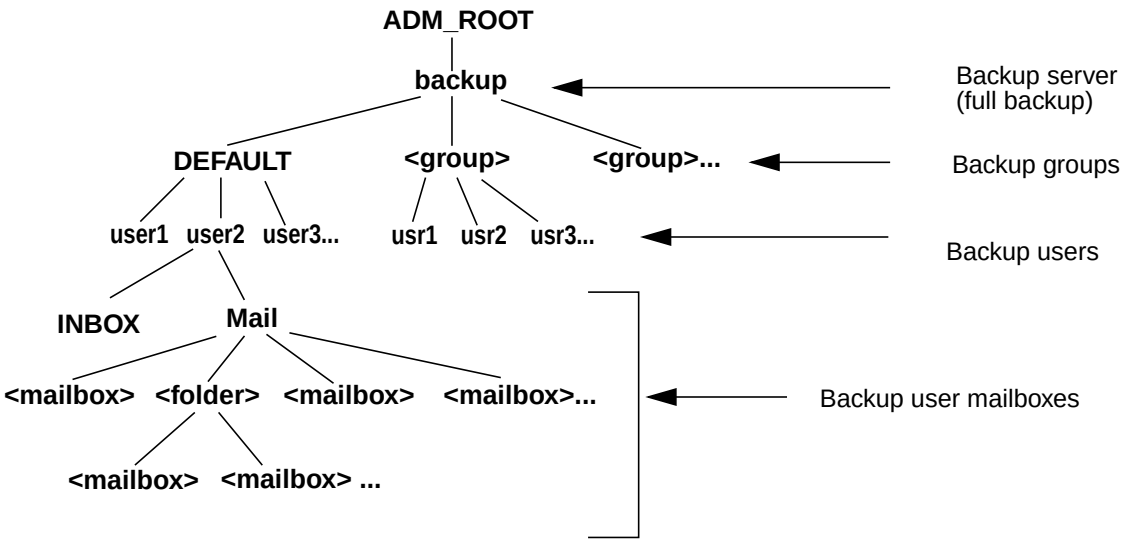
## ▼ Backing Up the Message Store Using Solstice Backup

1. **Run `mkbackupdir` at the command line.**

   This creates a backup directory in `<ADM_ROOT>`. This backup directory is not the actual directory where the mail store will be backed up, but a directory “image” of how mail is stored in SIMS. This image provides information to the backup program

on how the mail store will be stored on the backup media. The actual `mkbackupdir` files are empty files. If the directory already exists, `mkbackupdir` synchronizes the directory structure with the current folder / mailbox hierarchy.

The directory is structured as follows:




`ADM_ROOT` is the administration root directory defined in `/etc/opt/SUNWmail/ims/ims.cnf`. By default the directory is `/var/opt/SUNWmail/ims/adm/`.

`backup` is the *directory structure* under which all message store data on the server resides (no real backup data exists here). Choose `ADM_ROOT/backup` to backup all the message data on the server.

Backup *groups* are optional directories under which the mail folders for groups of users, say all users in a specific organization, can be placed. For example, folders for users in marketing can be placed in a directory called `mktg`, field service in `field_service`, and so on. Creating groups is useful for scheduling parallel

backups by group. If you do not create any groups, all folders go under a directory called `DEFAULT`. Create groups with `mkdir` and move the directories under `DEFAULT` into the desired groups. (See next step.)

The `user` directories contain the mail for each mail user. Under each `user` directory is a file called `INBOX` which stores all incoming messages and a `Mail` directory containing the user's mailboxes and folders. (Here mailboxes are files that hold messages and folders are directories that hold mailboxes.) Administrators can choose to backup at the user level or the mailbox level.

You can also do an incremental backup by running:

```
mkbackupdir -d <date (yyyymmdd) since last full backup>
```

This will create a backup directory with mailboxes that have been modified since that date. Solstice only backup those mailboxes. For example, if you run a full backup using `mkbackupdir` on 1 January 2000, you can do one week incremental backup by running:

```
mkbackupdir -d <20000108>
```

2. **(Optional) Create user groups in the directory hierarchy.**

   You can structure users into hierarchical groups in the directory. This allows you to backup specific groups of users.

3. **Start Solstice Backup.**

   Use Solstice Backup to backup `<ADM_ROOT>/backup`. Do not use the Solstice Backup incremental backup feature because the `adm/backup` directory is just an image of the user folder structure in the message store—it does not contains actual data. There are configuration files under the `adm/backup` directory that tell **Solstice Backup** to use `imbackup` to backup the mailboxes. The problem is the `adm/backup` directory doesn't get updated at the same time the users update their mailboxes. `mkbackupdir` updates the `adm/backup` directory in batch mode. Solstice Backup won't be able to tell which mailbox has been updated since the last backup by looking at the `adm/backup` directory. Therefore the Solstice Backup incremental backup won't work for us.

   Refer to the Solstice Backup documents (`http://docs.sun.com`) or the *SIMS Reference Manual* for detailed instructions.

---

**Note –** .nsr files are generated by the `mkbackupdir` command. It contains standard Networker directives and should never be edited.

---

4. **Automate this procedure.**

   The preceding steps describe how to do a manual backup. However, we recommend that you set up a cron job to run `mkbackupdir` and then use the Solstice Backup GUI to schedule backups after `mkbackupdir` is run.

## ▼ Restoring the Message Store

You cannot use `imrestore` to restore mailboxes backed up with Solstice Backup and you cannot use Solstice Restore to restore mailboxes backed up with `imbackup`. For more detailed information on how to use Solstice Restore and `imrestore`, refer to the Solstice Backup documents (`http://docs.sun.com`) or the *SIMS Reference Manual*.

If you use Solstice Restore, you will receive the message “File already exists. Do you want to overwrite, skip, backup, or rename?” Choose overwrite. This message appears because the backup tree is just the structure, i.e, it consists of empty files and stays that way permanently.

---

**Note –** If you use Solstice `recover` command, then you can use the `-A` and `-iy` arguments to suppress this message.

---

# Folder Check

The folder check utility scans through the Sun Message Store and the user folders verifying links. That is, it verifies that all the messages in the folders are accessible. In addition to running the folder check utility at regular intervals for maintenance, you can also run this utility after a system failure to ensure that all message deliveries were made while the system was in a questionable state. If the utility determines that messages are not in user folders and hence were not delivered, it will redeliver the messages.

You can invoke the folder check utility by issuing the `imcheck` command at a command line interface. For information on the `imcheck` command, refer to the *SIMS Reference Manual*.

# Importing `/var/mail` Users

The import mailbox utility automatically imports an existing user’s Inbox folder and all messages from `/var/mail` to the Sun Message Store. You must manually import a `/var/mail` user’s private folders.

You can invoke the import mailbox utility by issuing the `imimportmbox` command at a command line interface. For information on the `imimportmbox` command and importing /var/mail users refer to “Migrating /var/mail Mailboxes” on page 327 and the *SIMS Reference Manual*.

For information on the other maintenance utilities with which import mailbox can run concurrently, refer to TABLE 7-3.

## Deleting Old Messages

The `imexpire` command allows administrators to mark as permanently deleted, or "expired" any user messages older than a specified date or older than a number of specified days. The deleted messages are expunged from the user mailbox when the user connects or disconnects from the server. The actual data is removed from the message store when `impurge` is run. Refer to the *SIMS Reference Manual* for further details.

## Delete User

See "To Delete a User or Group Entry from the Directory" on page 72.

# Maintaining the Directory Service



## Maintaining the Data Store Attribute Indexes

You must rebuild the data store attribute indexes every few weeks or whenever large numbers of entries have been deleted. This also frees disk space no longer required by the attribute indexes.

To rebuild the indexes for a data store, choose Update Indexes from the LDBM Data Store menu of the data store property book. You need to do this for each data store individually. Alternatively, you can use the `idxgen` command to regenerate indexes. Refer to the *SIMS Reference Manual* for information on the `idxgen` command.

---

**Note –** If you add any additional index definitions to slapd.conf, you must regenerate indexes before running `slapd` again. This is true even if you have not yet added data matching the new indexes to the Directory.

---

## ▼ To Back Up a Data Store

Select Data Store>Selected pulldown>Modify Data Store>Ldbm Data Store pulldown>Backup

1. **To back up a data store, select the data store you want to back up from the Data Store list, and choose Modify Data Store from the Selected menu.**

   The data store's property book is displayed.

2. **Click the Ldbm Data Store pull-down menu, and select Backup.**

   A selector window is displayed.

3. **Use the selector window to select or create a directory for your backup.**

4. **Click Save.**

## ▼ To Restore a Data Store

Select Data Store>Selected pulldown>Modify Data Store>Ldbm Data Store pulldown>Restore

1. **To restore a data store, select the data store you want to restore from the Data Store list, and choose Modify Data Store from the Selected menu.**

   The data store's property book is displayed.

2. **Click the Ldbm Data Store pull-down menu, and select Restore.**

   A selector window is displayed.

3. **Select the directory containing the backed up data store that you want to restore.**

   The default selection is the previous backup directory.

4. **Click Load.**

   The directory service is restarted with the restored data store.

## Backing Up the Directory Data Base

You can make backups of your directory database by using the command `ldbmcat` which converts all your database information to LDAP Directory Interchange Format (LDIF) format. To restore your database, use the command `ldif2ldbm`. Refer to the *SIMS Reference Manual* for information on these commands.

## Backing Up and Restoring Directory Service Configuration

The Admin Console enables you to back up your directory service configuration at any time and save as many configurations as you want. You can restore any of your backed up directory service configurations.

### ▼ To Back Up the Directory Service Configuration

| AdminConsole>LDAP Server pulldown>Backup Config |
|---|

1. **In the Admin Console, click the LDAP Server pull-down menu and choose Backup Config.**

   If the directory service damon `slapd` is running, you are prompted to stop it. When it is stopped, a selector window is displayed.

2. **Use the selector window to select or create a directory for your backup.**

3. **Click Save.**

### ▼ To Restore the Directory Service Configuration

AdminConsole>LDAP Server pulldown>Restore Config

1. **In the Admin Console, click the LDAP Server pull-down menu and select Restore Config.**

   If the directory service damon `slapd` is running, you are prompted to stop it. When it is stopped, a selector window is displayed.

2. **Select the directory containing the backed up configuration that you want to restore.**

   The default selection is the previous backup directory.

3. **Click Load.**

   The directory service is restarted with the restored configuration.

## ▼ To Start the Directory Services

To start the directory service from the Admin Console, choose Start from the LDAP Server menu in the Directory Server property book.

When you install a directory server, this command is added to the system startup file, so that the server is started automatically when the machine is rebooted.

### ▼ Starting the Directory Services Using the Command Line Interface

To start the directory server daemon, `slapd`, from the Admin Console, choose Start from the LDAP Server menu in the Directory Server property book. You can also start the directory server daemon by typing the following command as `root`:

```
# /etc/init.d/slapd start
```

To start the email administrator's configuration interface, as `root` type:

```
# /etc/init.d/web500gw start
```

To start the SNMP agent, `snmpslapd`, as `root` type:

```
#/etc/init.d/init.snmpslapd start
```

When you install the directory service, these commands are added to the system startup file, so that all the server daemons are started automatically when the machine is rebooted.

## ▼ To Stop the Directory Services

To stop the directory server from the Admin Console, choose Stop from the LDAP Server menu in the Directory Server property book.

Stopping the directory server automatically stops the replication server. If you have set up a replication schedule, the replication server is restarted automatically when you restart the directory server, and will continue to follow the schedule

### ▼ Stopping Directory Services Using the Command Line Interface.

To stop the directory server daemon, `slapd`, from the Admin Console, choose Stop from the LDAP Server menu in the Directory Server property book. You can also stop the directory server daemon by typing the following command as `root`:

```
# /etc/init.d/slapd stop
```

Stopping the directory server automatically stops the replication server. If you have set up a replication schedule, the replication server is restarted automatically when you restart the directory server daemon, and will continue to follow the schedule.

To stop the email administrator's configuration interface, as `root` type:

```
# /etc/init.d/web500gw stop
```

To stop the SNMP agent, `snmpslapd`, as `root` type:

```
#/etc/init.d/init.snmpslapd stop
```

# Maintaining the Connectivity Services

For information on maintaining the Connectivity services, refer to the *Sun Messaging Connectivity Services Channel Guides*.

CHAPTER 8

# Troubleshooting

| Topic/Task | Description | Page |
|---|---|---|
| Troubleshooting Tools | - Event Log Manager<br>- Logging Facilities | 248 |
| Troubleshooting the Admin Console | - Preventing the “Warning Applet” Banner<br>- Admin Console Core Dump | 252 |
| Troubleshooting the Message Store | - User Not Able to Access INBOX<br>- Problems Turning Message Store Quota Enforcement Off and On<br>- Message Purge Failure<br>- User Can’t Perform Internationalized String Search on Mail Messages | 255 |
| Troubleshooting the IMTA | - SMTP Connection Aborted<br>- Message Queue Problems<br>- Tracking Messages<br>- Address Unknown to IMTA<br>- Multiple Reprocess Jobs Generated<br>- IMTA Error Messages | 257 |

| Topic/Task | Description | Page |
|---|---|---|
| Troubleshooting Tools | - Event Log Manager<br>- Logging Facilities | 248 |
| Troubleshooting the Directory Service | - Directory Service Logging<br>- LDIF Attributes Required By SIMS<br>- Diagnosing SIMS Problems Caused by Improper Directory Entries | 262 |
| Crash Recovery | - SIMS Crash Recovery<br>- Message Store Crash Recovery<br>- Admin Console Crash Recovery | 266 |
| Error Messages | Go to Appendix E, “Error Messages.”<br>- User Management Error Messages<br>- Log Manager Error Messages<br>- IMTA Error Messages<br>- Queue Monitor Error Messages<br>- Message Access Protocols Error Messages<br>- Directory Service Error Messages | 357 |

# Troubleshooting Tools



## Logging Facilities

The logging of events, email messages, and various debug information provides a means through which you can troubleshoot a problem with your SIMS components. TABLE 8-1 provides an overview of the SIMS logging facilities available via the Admin Console.

**TABLE 8-1** Overview of Logging Facilities

| Logging Facility | Description | For More Information, Refer To |
|---|---|---|
| Log Manager | Logs transactional event messages generated by the SIMS administration server components. You can search the log for events that occurred up to 7 days ago. | "Event Log Manager" on page 249 |
| IMTA log | If configured, an IMTA channel logs in each email as it enters and is removed from the channel queue. | "Message Logging" on page 130 and "Tracking Messages" on page 259 |
| Directory services log | If configured, logs various information such as protocol trace information for function calls and debug information about packets. | "Configuring Logging" on page 213 and "Directory Service Logging" on page 262 |
| Message Store / Message Access | Logs all `imaccessd` errors and events. | Search for SUNWMail.ims in `/var/log/syslog` |

# Event Log Manager

The Log Manager logs event messages generated by the SIMS administration server. It does not log system events (see "Logging Facilities" on page 248). Each event message contains the following information:

- Log Type - By default, the log type is audit trail. The audit trail log contains event messages generated by the administration server. Typically, events logged include starting, stopping, and modifying a configuration file associated with the components that the administration server interfaces (IMTA, message access protocols, Sun Message Store, and directory service). For more information on the administration server, refer to "SIMS Administration" on page 2.
- Event Type – Type of event message. Currently only logs transactional events.
- Severity – An assigned ranking of the event severity. Possible rankings include debug, informational, warning, critical, and fatal error.
- Time Stamp - Date and time that event message was generated.

---

**Note –** See "Log Manager Error Messages" on page 359 for error message information.

---

While the System Components and System Status enable you to determine if a component is in an alert or down state, the Log Manager enables you to trace event messages that may help you determine why the component is in an alert or down state. For more information on the System Components and System Status, refer to “SIMS Component Status” on page 52.

The Log Manager also enables you to take a snapshot of the event messages compiled in the current table. The snapshot is saved as a file to the `/var/log/simsauditlog` directory. You can open the file and edit the contents using your preferred text editor.

## ▼ To Trace Log Entries

The Log Manager in the Admin Console helps trace entries logged from the administration server only. This logging utility is primarily provided for audit trail debugging purposes. By viewing those audit trail messages, the user can find out who, when, and where an administrative task was performed on the SIMS mail server. For other logging facilities, see TABLE 8-1 on page 249.

| AdminConsole>Log Manager> |
|---|

1. **From the Admin Console home page, click on the Log Manager icon.**

   The Log Manager property book appears as shown in FIGURE 8-1.

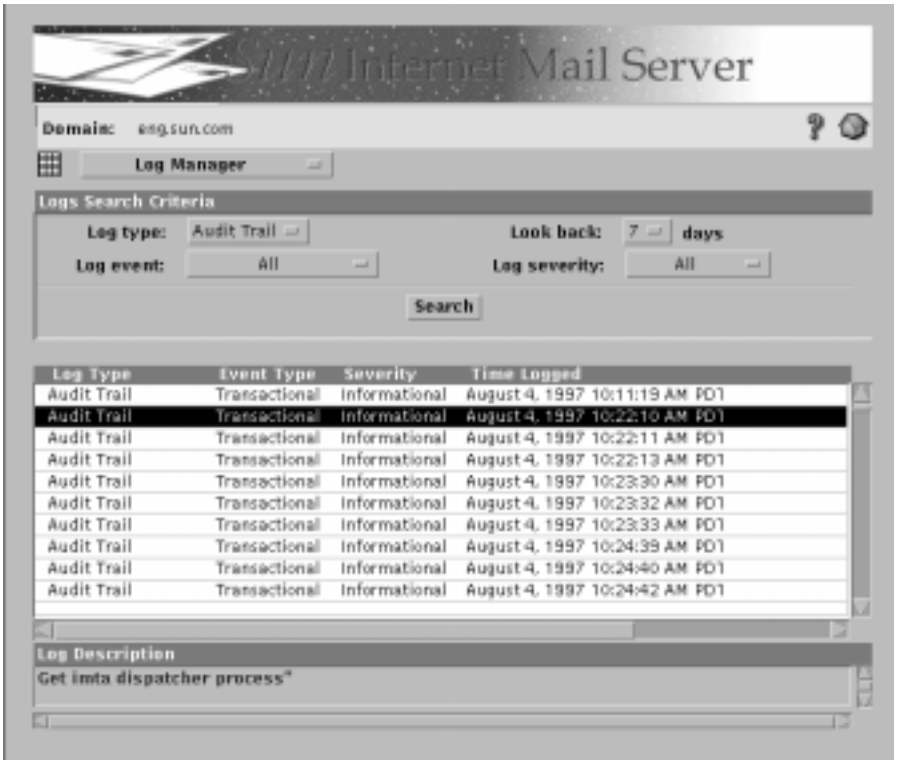



**FIGURE 8-1** Log Manager Property Book

The Log Manager property book is divided into three portions: the top portion displays an area where you can specify search criteria for log entries, the middle portion is composed of a table that displays the specified log entries, and the bottom portion displays more information for a particular entry.

2. **Specify the criteria for the log entries in which you are searching.**

   The default log type is Audit Trail. The Log Manager will search for detailed event messages for the Administration Server components.

   a. **Specify the type of event for which you want to search.**

      The following choices are provided *Log event*, but at this time only transactional events are logged.

      **All** - The result is the same as choosing Transactional.

      **System** - Not operational. Reserved for future use.

      **Transactional** - Displays all the audit trail logs from the Administration Server. Logs transactional events.

**b. Specify how far back in the log you want to search for an event.**

Use the “Look back” pull-down menu to specify a time frame from one to seven days. (One day is the 24-hour period back from when you perform the search.)

**c. Specify the severity of the event for which you want to search.**

Use the associated pull-down menu to specify fatal error, critical, warning, information, debug, or all.

**d. Click the Search button.**

Depending on the parameters of the search you specified, it could take from a few seconds to a few minutes. When the search is complete, the event messages will display as entries in the table in the Log Manager property book.

3. **Click on the entry to highlight it and display more information in the bottom portion of the Log Manager property book.**

4. **(Optional) Save a snapshot of the event messages currently compiled in the table by clicking on the Log Manager menu and selecting Save the current display.**

The snapshot is saved as a file to the `/var/log/simsauditlog` directory. You can open the file and edit the contents using your preferred text editor.

# Troubleshooting the Admin Console



## ▼ Preventing the “Warning Applet” Banner

When accessing the Admin Console using a Hotjava browser, you may see a “Warning Applet” banner in all dialogs. To prevent this banner from appearing, add the following two lines to your `.hotjava/properties` file:

```
hotjava.default.security=low
hotjava.default.signed.security=low
```

1. **cd to the .hotjava location:**

```
% cd
% cd .hotjava
```

2. **Edit the properties file to show the following:**

```
hotjava.default.security=low
hotjava.default.signed.security=low
```

3. **Save the file and restart the Admin Console.**

---

**Note –** It is always a good idea to change `hotjava.default.security` to `low` as some Admin Console functions won't work otherwise.

---

## ▼ Admin Console Core Dump

Under certain situations, most often while switching back and forth between pages, the Admin Console may crash and the browser may core dump on the Java virtual machine. To prevent this from happening, set your `JDK_HOME` to `/usr/java`. Change the setting in your Preferences menu or enter the command below:

```
% setenv JDK_HOME /usr/java
```

Then start the browser.

## ▼ Forgetting the Admin Password

If you forget the Admin Console password, you can change it in the `slapd.conf` file:

1. `cd` **to** `/etc/opt/SUNWconn/ldap/current`
2. **Edit the** `slapd.conf` **file.**

   Change the rootpw. Use {crypt} before the passord if it is encrypted.

3. **Restart slapd.**

   If you are using an unencrypted password, you can change it to be encrypted once you log in to the Admin Console. Go to the Sun Directory Services property book, and set Password encryption to Crypt.

# Troubleshooting the User Manager



## Can’t Create New Users

If you reinstall the SIMS server, make sure you reboot your system after the installation of the new server. If you do not reboot, you could have problems creating new calendar or mail users.

## New Entries Created in Old Domain

If domain information in DNS server is changed, SIMS needs to be reinstalled. If isn’t reinstalled, the Admin Console uses the old value for the domain and new user entries will be created under the old domain, not the new domain.

When the administration server starts, it compares the value of DNS domain and domain record in the administration server. If they are different, a warning message displays:

```
WARNING - domain value of this mail server is different from
the one in Admin Server property file - please check SIMS
manual for more information.”
```

To fix this inconsistency you can reinstall SIMS or you can manually change the value of `MTA.domainname` in the `adminserver.properties` to be the same as the DNS domain. In general, we do not recommend manually editing the `adminserver.properties` since a careless edit can adversely affect SIMS. But if you want to avoid reinstalling SIMS and are very careful, you can stop the adm.server and manually change the domain information. Use the following procedure:

1. **`stop adm.server`**
2. **`cd /etc/opt/SUNWmail/admin`**

   Change the value of `MTA.domainname` to the new domain name in adminserver.properties

3. **`start adm.server`**

# Troubleshooting the Message Store



## User Not Able to Access INBOX

If an administrator created a new `/var/mail` account on SIMS using the Admin Console, and she forget to create a UNIX account, the user will not be able to read the INBOX. The following error message will appear:

```
Unknown uid: <username>
```

The solution is create the UNIX login account and the home directory:

1. **All /var/mail store users must have a shell account and HOME directory setting.**
2. **Create the UNIX shell account using the desired operating system tools (e.g., admintool).**
3. **Using the Admin Console set the “Home Directory” field.**

## Problems Turning Message Store Quota Enforcement Off and On

If for some reason you turn the message store quota enforcement off, modify a user’s quota, and then turn message store quota enforcement back on again, you should shut down SIMS. If you don’t, it’s possible that for a short period incoming mail to the modified user will be temporarily not delivered and returned to the sender. We absolutely recommend that you NOT turn the quota enforcement off to modify a user’s quota, but in the very rare and unusual circumstance that you do, use the following procedure:

1. Shut down SIMS
2. Do whatever work you need to do with quota turned off
3. Turn quota back on

4. Run `iminitquota -a`

5. Restart SIMS

Here's an example of how a mail store full situation could occur, even though the mail store is not full: let's assume a user, *Galaxion*, has a user quota of 20 megabytes and is currently using 15 megabytes. Now, let's say the message store quota enforcement is turned off, and while it is turned off three things happen: 1) Galaxion's quota is reduced to 10 megabytes, 2) Galaxion deletes a bunch of mail and reduces her mail store usage to 7 megabytes, 3) An incremental directory synchronization occurs. Now, when quota enforcement is turned back on, `iminitquota -a` must be run to tally up mail storage usage. As soon as an incremental directory synchronization occurs, the system will start to enforce the new quota—in this case 10 megabytes. However, the system will read the old mail store usage value of 15 megabytes, until `iminitquota -a` is finished running. For 10,000 users this may take 30 minutes. So for that 30 minutes, the new quota of 10 megabytes may be used, but the old mail store usage value of 15 megabytes will be read. This causes the enforcement to stop accepting new mail.

## Message Purge Failure

`impurge` will fail if the file system is full. You will see the following error message in the `syslog` file:

```
Jul 11 12:52:27 mcm-charmed SUNWmail.ims.impurge[17436]: PURGE
erro: Cannot create expungedir tmp file
Jul 11 12:52:27 mcm-charmed SUNWmail.ims.impurge[17436]: PURGE
erro: Cannot create ADM expungedir tmp file
```

There is no convenient workaround, but following procedure will work. Be extremely careful.

1. **Shutdown SIMS.**

2. **Move some data to another file system**

   You must make enough disk space for the day that has that largest total data+index byte size. Move **all** the data buckets, the `index` and `indexdir` files for that day at once. For example:

   ```
   $ mv /var/opt/SUNWmail/ims/data/1998/0627 /tmp/data.0627
   $ mv /var/opt/SUNWmail/ims/index/1998/0627 /tmp/index.0627
   ```

3. **Run impurge.**

4. **Move the data you moved in step 2 back to the message store.**

   For example:

```
$ mv /tmp/data.0627 /var/opt/SUNWmail/ims/data/1998/0627
$ mv /tmp/index.0627 /var/opt/SUNWmail/ims/index/1998/0627
```

5. **Run impurge again.**

## User Can’t Perform Internationalized String Search on Mail Messages

SIMS supports internationalized string searches in mail messages using any major character set. Search strings are no longer limited to ASCII/English.

Some mail clients, however, perform searches locally. That is, they use client code to perform searches. This client code may or may not support an internationalized search. To do an internationalized search using the SIMS search code, mail clients must send an IMAP SEARCH command to the server.

If a mail user cannot do an internationalized search, make sure that the mail client used sends an IMAP SEARCH command to the server.

# Troubleshooting the IMTA



## SMTP Connection Aborted

If the SMTP connection aborts, check whether the IMTA is running (SMTP server included):

```
% imta process
```

The result should list the three following processes (exactly one of each)

```
job_controller
dispatcher
<SMTP>
```

Restart the IMTA if it is not running.

If the IMTA continues to abort, look at the `tcp_smtp_server` log files to determine the problem.

`/var/opt/SUNWmail/imta/log`

To debug the IMTA problem, set `debug=1` in `/etc/opt/SUNWmail/imta/dispatcher.cnf`. Also enable the diagnostic output for the slave program in the intranet channel. (From the IMTA Property Book double click on the relevant SMTP channel and go to the *Diagnostics Output Section*. Check the box for *Enable diagnostic output for slave program*.

Try to start the IMTA again, and look at the debug output in the `tcp_smtp_server` log files at `/var/opt/SUNWmail/imta/log/tcp_local_slave*`.

## Sent Message Can’t Find Server Name

If DNS is not working, the administration server will display the following warning message in a console window:

```
***Can’t find server name for address <ip_address>: No response
from server. ***Default servers are not available
```

In addition, a java exception stack will be displayed and users will not be able to send mail until the DNS is once again operating.

The administration server does not itself depend on the DNS server—it can continue to operate and an Admin Console can connect to it. But the DNS server needs to be returned to normal operation before mail can be sent.

## Message Queue Problems

Undeliverable messages are probably either not being dequeued from the IMTA, being saved in `.HELD` file because it is looping between another server or channel, or it is stuck at another server. This section describes various message queue problems.

### Unjamming a Message Queue

If the IMTA stops processing messages in a queue, enter the following command as `root` for the queue that appears jammed:

```
# /opt/SUNWmail/sbin/imta run <channel name>
```

where <channel name> is specified in imta.cnf

### Message Not being Dequeued

If a message is not being delivered and remains in the message queue, it may be that it's not being processed by the master program for that channel. Try this:

1. Check whether the path of the *_master program is correctly set in /etc/opt/SUNWmail/imta/job_controller.cnf
2. Check whether inetmail has the permission to run the *_master program
3. Check whether inetmail can find all the libraries it needs to run the master program, by running

   ```
   % ldd <master_program_name>
   ```

4. For more information, look at /var/log/syslog
5. If you still have problems, set debug to 1 in job_controller.cnf, and run # imta restart job_controller. Resend the message and look at the newly created job_controller log file in /var/opt/SUNWmail/imta/log.

### .HELD Messages

If the IMTA detects a mail loop, that is, messages that bounce between servers/channels (this occurs because each server/channel thinks the other is responsible for delivery to an address), delivery is halted and the messages are stored in a file with the suffix .HELD in /var/opt/SUNWmail/imta/queue/<channel>. The message will be ignored by the IMTA and no further delivery will be attempted. Look at the headers in the message to determine which server/channel is bouncing the message. Fix the entry as needed and run the command:

```
% imta queue -retry_delivery <channel-name>
```

## Tracking Messages

Track a message by looking at the log files:

```
/var/opt/SUNWmail/imta/log/mail.log-current
```

| date/time | src-channel | dest-ch. | type | size(kb) | sender | rcpt |
|---|---|---|---|---|---|---|
| 3-Dec-1997 08:48:56 | tcp_locall | sims-ms | E | 1 | usr042@akwaba.eng.sun.com | usr041@sims-ms-daemon |
| 3-Dec-1997 08:48:58 | sims-ms | | D | 1 | usr.042@eng.sun.com | usr041@sims-ms-daemon |
| 3-Dec-1997 08:54:47 | tcp_locall | sims-ms | E | 1 | usr042@akwaba.eng.sun.com | usr041@sims-ms-daemon |
| 3-Dec-1997 08:54:47 | tcp_local | pipe | E | 1 | usr042@akwaba.eng.sun.com | usr041+autoreply@pipe-daemon |
| 3-Dec-1997 08:54:49 | sims-ms | | D | 1 | usr.042@eng.sun.com | usr041@sims-ms-daemon |
| 3-Dec-1997 08:54:49 | pipe | | D | 1 | usr.042@eng.sun.com | usr041+autoreply@pipe-daemon |

**Type:**
E: enqueued
R: returned
D: delivered
Q: transient failure on delivery

You can also view the message queue by running:

```
% imta cache -view <channel_name>
```

`imta test -rewrite` verifies that an address is properly handled by the IMTA.

If you still cannot find a lost message, check whether it's being held, by looking for `*.HELD` messages in the channel queues.

# Address Unknown to IMTA

If a sender sends a message to a valid address and receives a returned message with the error “User unknown,” verify the address by using the `imta test -rewrite` command. Most likely the user entry in the directory is not correct. Retrieve the full directory entry for this user, and verify it. IMTA `dirsync` can also check directory entry errors. Enter the command:

```
# imta dirsync -t
```

If no invalid entries are reported, try running `dirsync` again with the `-v` switch, and look at the newest `/var/opt/SUNWmail/log/dirsync.trc-*` for more information.

Once you've fixed the directory user info, you may run `dirsync` again, or wait until the periodic incremental `dirsync` runs. Run the command:

```
# imta dirsync [ other options ... ]
```

Finally, check the ownership of all files in `/var/opt/SUNWmail/imta/db` is set to `inetmail:mail`. Verify that all the files are writable by the owner.

## Multiple Reprocess Jobs Generated

If you are having severe performance problems, check if multiple reprocess jobs are being generated. `imta process<ret>` should show only a single `reprocess` running. If more are showing, look in `/etc/opt/SUNWmail/imta/aliases`, and verify that it contains the following line:

```
postmaster: <user-name>@FDQN
```

Make sure the postmaster address is valid and is receiving mail. If it doesn't, add this line do a fresh restart of the IMTA.

```
# imta restart
```

## Addresses Not Reversed

The IMTA has the ability to reverse envelope `from:` and `header` addresses. Generally, this is used to turn the address form `username@host.domain` to the more public form `first_name.last_name@domain` so that recipients outside of the domain only see the latter form.

This functionality is not always activated. For instance, it is not active if the routability scope is set to Local System Users Only. To enable, set the option `USE REVERSE DATABASE` to `5` in `/etc/opt/SUNWmail/imta/option.dat` and make sure the list of channel keywords for the delivery channel does not contain `noreverse`.

Another cause of reverse address failure is the absence or incorrect configuration of the `preferredrfc822originator` LDAP attribute.

To further diagnose the problem, set `MM_DEBUG` to `5` in `option.dat` and activate the slave diagnostic output for the queuing channel. Restart the IMTA and reproduce the problem. Examine the debug output file created in `/var/opt/SUNWmail/imta/log`. For information on how to use diagnostic output, see "To Configure Diagnostics Output" on page 128.

## SMTP Access Restrictions Not Working As Expected

Common reasons for such problems include:

1. Interference between access control rules and rewrite rules. The address is rewritten before it is processed by the access rules, and the access rules handle the original and rewritten addresses differently.

2. Interference between access rules. A typical case arises when a message is blocked using IP addresses only. Other rules involving addresses are never applied since IP level envelope information is sufficient to make an access decision.

The problem can best be diagnosed using the debugging functionality. See “To Configure Diagnostics Output” on page 128

# Troubleshooting the Directory Service



## Directory Service Logging

You can configure the information logged by a directory server, as described in “Configuring Logging” on page 213. By default, the directory server daemon logs information about connections and operations in `/var/opt/SUNWconn/ldap/log/slapd.log`. The replication daemon and the LDAP/HTTP gateway daemon also maintain log files. These files, `slurpd.log` and `web500gw.log`, are stored in the same directory as the `slapd` log file.

In the event that you need more diagnostic ability than is provided by SIMS default Directory configuration, you may run `slapd` from a shell manually, providing additional diagnostics. See the `-d` and `-s` arguments to `slapd`, documented in `slapd(8)` man page. Note that running with additional diagnostics turned on will run `slapd` in the foreground, and will use up disk space quickly if you have all optional diagnostic output enabled.

If you are running SNMP monitoring for the directory service, there is also an `snmpslapd.log` file.

In addition, certain messages may be logged to the console of the machine on which `slapd` is running, and/or to `/var/adm/messages`.

TABLE 8-2 contains an example of the information that is logged during a search operation. This example shows two interactions with the directory, the first to add an entry, and the second to search the directory for all entries that have a `commonName` attribute. The log includes details of the bind and unbind, and operational information.

**TABLE 8-2** `slapd.log` Example

```
Thu May 15 16:03: conn=9 fd=15 connection from unknown (127.0.0.1)
Thu May 15 16:03: conn=9 op=0 BIND dn="CN=admin,O=sun,C=us" method=128
Thu May 15 16:03: conn=9 op=0 RESULT err=0 tag=97 nentries=0
Thu May 15 16:03: conn=9 op=1 ADD dn="O=sun,C=us"
Thu May 15 16:03: conn=9 op=1 RESULT err=0 tag=105 nentries=0
Thu May 15 16:03: conn=9 op=2 UNBIND
Thu May 15 16:03: conn=9 op=2 fd=15 closed errno=0
Thu May 15 16:26: conn=10 fd=15 connection from unknown (127.0.0.1)
Thu May 15 16:26: conn=10 op=0 BIND dn="CN=admin,O=sun,C=us" method=128
Thu May 15 16:26: conn=10 op=0 RESULT err=0 tag=97 nentries=0
Thu May 15 16:26: conn=10 op=1 SRCH base="O=sun,C=us" scope=2 filter="(cn=*)"
Thu May 15 16:26: conn=10 op=1 RESULT err=0 tag=101 nentries=150
Thu May 15 16:26: conn=10 op=-1 fd=15 closed errno=0
Thu May 15 16:26: conn=10 op=2 UNBIND
```

# LDIF Attributes Required By SIMS

Even if you are going to write your own LDIF generation scripts, it is recommended that you use `imldifsync` on a very small subset of your own data as a learning tool to see what entries are generated from which input data. However, we provide the following information as a Quick Reference aid in generating your own LDIF (refer to Appendix D, “SIMS Directory Schema and Directory Information Tree for a complete list):

---

**Note –** The complete set of attributes and their syntaxes are defined in `/etc/opt/SUNWconn/ldap/current/slapd.at.conf` and in other schema files accessed by `slapd` via the *include* directive in `/etc/opt/SUNWconn/ldap/current/slapd.conf`.

---

**TABLE 8-3** Some of the Default LDIF Attributes and Syntax

| LDIF Abbreviation | LDIF Full Name | Created From Format |
|---|---|---|
| SN | surname | The “last name”/”family name” part of each gecos entry |
| CN | common name | The “first name”/”given name” part of each gecos entry |
| none | initials | |
| none | givenName | CN |
| none | freeformName | CN |

**TABLE 8-3** Some of the Default LDIF Attributes and Syntax *(Continued)*

| LDIF Abbreviation | LDIF Full Name | Created From Format |
|---|---|---|
| none | preferredrfc822Recipient | User alias entry in `/etc/mail/aliases` or equivalent, of value `userid@mailhost.maildomainname`. |
| none | preferredrfc822Originator | User alias entry `/etc/mail/aliases` or equivalent, of value `first.lastname@maildomain` |
| none | rfc822Mailbox | User alias entry `/etc/mail/aliases` or equivalent, of value `first.lastname@mailhost.maildomain` and `userid@mailhost.maildomain`, the same values as `preferredrfc822Originator`. Additional entries will be generated for nickname aliases, etc |
| none | mailProgramDeliveryInfo | Set by SIMS to a user supplied string for mail aliases that pipe incoming mail through a program, as specified in the `/etc/mail/aliases` file or its' equivalent |
| none | mailDeliveryOption | Set by SIMS to one or more of "mailbox", "forward", "file", or "program" depending on what type of mail alias generates this attribute. |
| none | mailForwardingAddress | Set by SIMS for `mailDeliveryOption` equal to "forward", of format `userid@mailhost.maildomain`, and `first.lastname@mailhost.maildomain`. |
| none | mailHost | Manually configured mail-server field in `imldifsync.conf` |
| none | userPassword | Password field from user password entry |
| none | uid | UNIX user ID (account name) from user passwd entry |
| none | homedirectory | From user password entry |
| none | datasource | `imldifsync` sets the datasource. |
| none | mailFolderMap | Set by SIMS to one of Sun-MS or UNIX V7 for message store or `/var/mail` files respectively. |
| none | creatorsname | DN of admin account used to create the entry |
| none | createtimestamp | Created by directory during `ldapmodify` |

# Diagnosing SIMS Problems Caused by Improper Directory Entries

Because SIMS relies completely on the presence and values of certain attributes in the directory, the single most common set of problems installing and maintaining SIMS is with LDIF generated by sources other than `imldifsync`, or by `ldapmodify`

if accurate information is not entered into the SIMS install HTTP forms. This section discusses some common problems caused by incorrectly configuring attributes, and provides hints in diagnosing them.

## General Hints

LDIF-generating scripts should be careful not to accidentally include non-printing characters or white space characters in attribute values. This is most commonly seen when a space (‘ ‘) is added at the end of an attribute value when not appropriate. For example, you should enter the value for `mailFolderMap` as:

```
“mailFolderMap: SUN-MS”
```

and not as:.

```
“mailFolderMap: SUN-MS ”
```

where the quotes show the inappropriate spaces.

It is always best to try and scope a particular problem before trying to diagnose it. This may sound obvious, but within the context of SIMS this means:

- If some users are having problems, dump the LDIF for users that are known to work (using `ldapsearch(1)` or the Admin Console) and compare this with the entries of users that do work.
- If data was added with an LDIF generating script, or even with `imldifsync`, for users that don’t work, try adding the same information via the Admin Console. If this works then look for white space or unprintable characters and determine if they are being added by the LDIF-generating scripts.

## Users Can’t Login to Their IMAP Mail Server

Check the following attributes:

- `userPassword` - Should match the encrypted password entry in `/etc/passwd` or the equivalent. Note this will not be true if the attribute is changed directly sometime after the directory has been in use for a while.
- `uid` - Should match the uid in `/etc/passwd` or the equivalent.
- `mailFolderMap` - This currently has only one of two valid values: “Sun-MS” for users receiving mail via the SIMS messages store, and “UNIX V7” for users who use the `/var/mail` inbox format.

### Mail Inbound to the SIMS MTA Bounces

Check the following attributes:

- `rfc822Mailbox` - Should have valid user@FQDN (fully qualified DNS domain name) value.
- `mailDeliveryOption` - Mailbox, native, program, forward, and file for users; shared, program, and file for groups.
- `mailFolderMap` - See “Some of the Default LDIF Attributes and Syntax” on page 263

### Mail Delivered Does Not Arrive

- `mailQuota` - Set incorrectly
- `mailFolderMap` - See “Some of the Default LDIF Attributes and Syntax” on page 263

### Mail Forwarded Between SIMS and Other Servers Isn’t Received.

- mailForwardingAddr - Not set or set incorrect

# Crash Recovery



## ▼ SIMS Crash Recovery

If your mail server becomes nonfunctional, you must perform the following procedure:

1. **Enter the following command as root:**

```
# imta queue-recover_crash
```

This command invokes a utility that rebuilds the queue caches for the Sun Message Store and `/var/mail`. These caches may become corrupted during the time that the mail server becomes nonfunctional. For example, a message may be partially written to the Sun Message Store cache during the time that the mail server becomes nonfunctional. Running the utility will enable the mail server to clean up these types of corruption.

2. **Stop any components that may still be running by entering the following:**

```
# /etc/init.d/im.server stop
```

3. **Start all components by entering the following:**

```
# /etc/init.d/im.server start
```

## ▼ Message Store Crash Recovery

In the event of a catastrophic system failure, the message store may be left in an inconsistent state and in some instances require data recovery from backup media. If you see the following message, it means that SIMS did not start and the store was shut down abnormally:

```
/var/log/syslog.1:Oct 2 16:40:24 mcm-nitro
SUNWmail.ims.imaccessd[1373]:Message store may be inconsistent.
Please run imcheck -c
```

Follow the procedure below to recover the message store.

1. **Change to the proper directory:**

```
% cd /opt/SUNWmail/sbin/imta
```

2. **Make sure IMTA is not running. To stop the IMTA, run:**

```
% imta stop
```

3. **Make sure imaccessd is not running. To stop the imaccessd, run:**

```
% mt.scheduler stop
```

---

**Note –** The imaccessd process should never be killed using the `kill -9` command. If this should accidently occur, run `imcheck -c` before restarting imaccessd.

---

4. **Run the following as `root`:**

```
% imcheck -c -f <filename>
```

   a. **If imcheck completes successfully, check the `/var/opt/SUNWmail/ims/adm/restore_log` file. If this file is present, restore the users whose names are in this file, and then remove the file. For example:**

```
% imrestore -i
% rm /var/opt/SUNWmail/ims/adm/restore_log
```

   b. **If imcheck fails, save the syslog file and the store report, then contact your Sun Service Provider. The store report is at `/var/opt/SUNWmail/ims/adm/<filename>`. Do not restart the MTA and imaccessd.**

## ▼ Admin Console Crash Recovery

If the Admin Console hangs, kill the browser process, restart the browser, and reconnect to the Admin server. If the Admin Console crashes or vanishes, restart the browser and reconnect to the Admin Server. Note that switching back and forth between pages may cause Admin Console hang or crash while communicating with the Admin server. To kill a browser process on Solaris, perform the following steps:

1. **Find the process id.**

2. **Bring up a UNIX shell window and type the following command as root:**

```
# ps -ef | grep java <cr>
```

You will see something like this:

```
root 26141     1  0 hh:mm:ss pts/x    0:04 /usr/java/bin/java
                      sun.rmi.registry.RegistryImpl
myuid 350 10767  0 hh:mm:ss pts/x    0:00 grep java
root 26145     1  0 hh:mm:ss pts/x    0:46 /usr/java/bin/java
                      -Dlegacy.host= -Ddirectoryhost=motmot
                      -Dconsole.domain=eng.s
myuid 336 10767 24 hh:mm:ss pts/x    0:15 /usr/dt/appconfihotjava/
                   runtime/bin/sparc/green_threads/jre -classpath ..
```

3. **Kill the browser process using the following command:**

```
# kill <browser process number (in this case 336)> <cr>
```

4. **On rare occasions the Admin Server will crash and you may need to restart it using the following command:**

```
# /opt/SUNWmail/sbin/adm.server stop
# /opt/SUNWmail/sbin/adm.server start
```

---

**Note –** For Win 95 and NT environments, please refer to the corresponding Administrator's Guides for instructions on killing a browser process.

---

CHAPTER 9

# Secure Sockets Layer (SSL) Support in SIMS



## SSL Overview

SSL is an open, non-proprietary security protocol. It has been submitted to the W3 Consortium (W3C) Working Group on Security for consideration as a standard security approach for world wide web browser and servers on the Internet. SSL provides data encryption, server authentication, message integrity, and optional client authentication for a TCP/IP connection between a client and a server, or server and a server.

Using SSL with SIMS ensures security between a mail client and SIMS by encrypting email content sent by the SIMS server to the email client.

## SSL Encryption

SSL uses a method called *public-key encryption*. In a public-key encryption two keys are used. One for encrypting data (public key) and one for decrypting data (private key). A server sends its public key to any requesting client. This key is used by the client to encrypt data sent to the server. When the server receives the encrypted data from the client (called *ciphertext*), the server uses its private key to decrypt the ciphertext.

Conversely, a client can also send a public key to a server so that the server will encrypt data sent to the client. When the client receives the encrypted data from the server, the client uses its private key to decrypt the ciphertext.

## Authentication by Certificate

A certificate is a non-transferable digital file that contains certain identifying information. Specifically a certificate contains the issuers identity, the receivers identity, and the public key. The certificate is issued from a third-party whom both parties trust. This third party is known as a Certificate Authority (CA).

A Certificate Authority (CA) can be *internal*—you create certificates within your organization, or *external*— a third party can issue a certificate for you.

Both servers and clients can have certificates. When a server sends a certificate to a client, the process is called *server authentication*. When a client sends a certificate to a server the process is called *client authentication*. **If you plan on using encryption and SSL on your server, you must obtain a server certificate from a valid CA.**

# Setting Up SIMS with SSL

Since SSL provides encryption at the level of the network connection, different ports are used for secure and non-secure communications. Port 993 is the default port for secure IMAP4 connections; port 995 is the default port for secure POP3 connections.

---

**Note –** Mail read by the client from IMAP/POP over Secure Socket is encrypted. That is, the message read and the authentication is encrypted. Mail sent by the client to IMTA is not encrypted at this time.

---

SIMS supports SSL version 3.0. SIMS also supports the ability to create a local Root Certificate Authority (CA), and to create server credentials signed by that CA. The following instructions outline the steps needed to accomplish this. SIMS is also able to use server credentials signed by an external Root CA. Please contact a Sun Support representative for more information regarding this feature.

To set up SIMS with SSL, you must

- Create a Root Certification Authority
- Create a Key Pair and Certificate for Your Mail Server
- Install the Certificate and Key Package on the Machine Running SIMS

## ▼ To Create a Root Certification Authority

1. **Create the local root CA user, `skirca`.**

   You can use User Manager to create users. You will need to be able to log in as `skirca`, so be sure to specify a login shell and password for the new user.

   ---

   **Note –** The username must be `skirca`.

   ---

2. **Set up the FNS naming context on each machine that will use security tools.**

   a. **As root, run `/usr/http/bin/setup_fns`**

      This script will select FILES as the naming service and set up the naming context for all the users and the host. For example:

```
# <install_root>/usr/http/bin/setup_fns
Setting up FNS Naming context...
Done
#
```

   b. **You will need to run `setup_fns` on each host where you will use SSL certificates (at least the local root CA machine and the machine running SIMS).**

3. **Create a local root certification authority (CA).**

   a. **You will need to log in as skirca.**

   b. **Run `/usr/bin/create_rootca`.**

**c. You will be prompted for a Distinguished Name (DN) for thelocal root CA.**

We suggest that the `cn` (common name), `o` (organization), `st` (state), and `c` (country code) be chosen as the minimum attribute set of the root CA's DN.

```
#
# Distinguished Name:
#

Enter Distinguished Name (e.g. "o=SUN, c=US")
or q[uit]: cn=rootca, o=ABCD, st=california, c=us
```

**d. You will next be prompted for the name of a directory into which will be stored the key package and certificate.**

You may wish to locate this new directory in the home directory of the `skirca` user. You must specify an absolute pathname, i.e., the pathname must begin with a /.

```
#
# Directory for Storing RootCA Credentials:
#
Enter directory pathname under which the key package and
certificate will be stored, or q[uit].
Directory name ? /home/skirca/rootca-creds

keypkg: Generating RSA key pair for user "CN=ROOTCA, O=ABCD,
ST=CALIFORNIA, C=US"
```

**e. You will next be prompted, twice, for the password you wish to use for the local root CA key package.**

```
keypkg: Enter your NEW key package password: <enter password>
keypkg: Reenter your NEW key package password: <re-enter password>
keypkg: Key package generation succeeded
certify: Certificates issued:11, certificates available:1009
```

**f. You will next be asked if you want to store the root CA credentials in the naming service. Answer in the affirmative and follow the remaining instructions.**

```
#
#  Do you want to store RootCA creds in the naming service[y/n]: y

# Storing the RootCA creds in the naming serive
# You need to enter the root password
Password: <enter-root-password>
skistore: keypkg /home/skirca/rootca-creds/keypkgs/skirca.KEYPKG
successfully stored
skistore: certificate /home/skirca/rootca-creds/certs/skirca.CERT
successfully stored
skistore: Operation Completed

#
# The Rootca creds are stored in the naming service
```

## ▼ To Create a Key Pair and Certificate for Your Mail Server

1. **Log in as skirca on the local root CA machine.**
2. **Create a directory where you can store key packages and certificates.**

   The `skirca` user must be able to write to this directory:

   ```
   $mkdir /home/skirca/server-creds
   ```

3. **Find the IP address of the SIMS host for which you would like to create a certificate.**

   You can use ypmatch `<hostname>` hosts if you are using NIS or nslookup `<hostname>` if you are using DNS. For example:

   ```
   $ ypmatch bob hosts
   121.131.141.5 bob
   Run /usr/http/bin/setup_creds <certs_directory>
   <ip_address_of_server>
   ```

   For example, if our output directory is `/home/skirca/server-creds/` and the IP address of the mail server is 121.131.141.5:

   ```
   $ <install_root>/usr/http/bin/setup_creds /home/skirca/server-
   creds 121.131.141.5
   Creating Public/Private key pairs and certificates
   ```

```
for your server...
Enter Host Name on which you run the server:
(Hit return to use localhost)
```

4. **You will be asked to enter the following information about the server for which you are creating a certificate:**

   **hostname** - Enter only the host name of the machine running SIMS. For example, `bob`.

   **domain name** - Enter the domain name of the SIMS host. For example, `ABCD.com`.

   **Distinguished Name Suffix** - Enter the DN information without the `cn` (common name) attribute. The fully qualified domain name—the hostname plus the domain name you entered—will be used for the common name field.

   **certificate password** - Enter a password twice for this server's key. You will need this password when you install the key package and certificate on the mail server.

---

**Note –** You are creating a password for this key pair and certificate; it is different from the local root key password.

---

5. **You will be asked to enter the local root key password (the password for the local root CA key pair).**

   Once you have entered this, the key package for the mail server will be generated and a certificate will be created.

6. **The certificate will be stored in the file `<certs_directory>/certs/ <ip_address>.CERT`**

   For example, `/home/skirca/server-cred/certs/121.131.141.5.CERT`

## ▼ To Install the Certificate and Key Package on the SIMS Host

You will need to transport the output directory you used with `setup_creds` to the SIMS host (if it is not the same as the local root CA machine), and copy it to a permanent location on that machine.

1. **Transport the directory on floppy disk or any other transportable medium.**

   You may want to create a directory on the mail server where you can store the credential directories for many servers. For example, a directory named `/var/sims/SSL/` may contain directories such as `host1_pkg/`, `host2_pkg/`, `etc`. for each host for which you create a certificate.

2. **Log in as root on the mail server.**

3. **You must have set up the FNS naming context on the mail server machine. See "To Create a Key Pair and Certificate for Your Mail Server" on page 273.**

4. **Run `intall_certs` command.**

   The syntax is:

   ```
   /usr/http/bin/install_certs -p <certs_directory> -i <ip_address>
   <inetmail_uid>
   ```

   where,

   **certs_directory** - Is the directory containing the key package and certificates that were created on the local root CA machine.

   **ip_address** - Is the IP address of the server. This field is optional; if you do not specify anything, the default IP of the machine will be used.

   **inetmail_uid** - Enter the uid for inetmail user on your system.

   For example:

   ```
   # <install_root>/usr/http/bin/install_certs -p /var/sims/SSL/
   bob_creds/ -i 121.131.141.5 9870
   /usr/bin/skistore: certificate /var/sims/SSL/bob_creds/certs/
   skirca.CERT successfully stored
   /usr/bin/skistore: keypkg /var/sims/SSL/bob_creds/keypkgs/
   129.146.114.74.KEYPKG successfully stored
   /usr/bin/skistore: certificate /var/sims/SSL/bob_creds/certs/
   129.146.114.74.CERT successfully stored
   /usr/bin/skistore: Operation Completed
   /usr/bin/skilogin: Enter host key package password:
   ```

   ---

   **Note –** Take special care to ensure that the numeric UID value is specified properly for the `install_certs` command. This value should be the user id for "inetmail" user. For example:

   ```
    #id inetmail
    uid=72(inetmail) gid=6(mail)

   /usr/http/bin/install_certs -p /var/sims/SSL/bob_creds -i
   121.131.141.5 72
   ```

   See syntax for `install_certs` below.

   ---

5. **You will be prompted for the certificate password you used when you created the key package.**

   Once you have completed these steps, you have a working environment for running SSL with SIMS.

6. **Verify that the SSL ports are running.**

**a. Stop and start im.server or reboot the system.**

```
# /opt/SUNWmail/admin/sbin/im.server stop
# /opt/SUNWmail/admin/sbin/im.server start
```

**b. Enter the following commands to verify SSL port operation:**

```
# netstat -a -n | grep 993 | grep LISTEN
  *.993        *.*          0  0    0    0 LISTEN
# netstat -a -n | grep 995 | grep LISTEN
  *.995         *.*         0  0    0    0 LISTEN
```

If you do not receive this output, then SSL port operation is not running.

## Re-initializing the Credential Repository

It may be necessary, possibly as the result of an administrative error, to re-initialize the repository in which credentials are stored. The following instructions demonstrate the simplest way to accomplish this. Please be aware that after performing this operation, it will be necessary to repeat the steps outlined in "Setting Up SIMS with SSL."

### ▼ To Re-initialize the Credential Repository

1. **#su root**
2. **#skilogout -h**
3. **#/etc/init.d/skiserv stop**
4. **#rm -rf /var/fn**
5. **#/etc/init.d/skiserv start**

   It should now be possible to repeat the steps outlined in "Setting Up SIMS with SSL."

# Using Netscape Messenger and Microsoft Outlook Express with SIMS and SSL

## ▼ Using Netscape Messenger with SIMS and SSL

The first time you setup Netscape Messenger to read IMAP messages from SIMS setup for SSL, you will be prompted with the message:

"...Netscape does not recognize the signer of this Certificate"

This is because SIMS is using internally generated Certificate. When you are prompted you need to click on:

Accept this certificate forever (until it expires)

You should not get prompted after this point with the above message.

## ▼ Using Microsoft Outlook Express with SIMS and SSL

Once you configure Outlook Express for SSL, you should not need any other changes to make it work with SIMS.

CHAPTER 10

# User Administration

| Topic/Task | Description | Page |
|---|---|---|
| To Change the Mail Password | Change a users mail passwords | 280 |
| To Start and Stop the Vacation Notice | Automatically send a pre-written email message to anyone who sends mail to this address. | 280 |
| Alternative Delivery Programs | Making alternative delivery programs available through the user interface. | 102 |
| To Use Alternative Delivery Programs | Describes how users can access alternative delivery programs. | 281 |
| To Forward Mail | How to forward mail to a different email address. | 282 |

Most SIMS configuration is done by the system administrator, but some configuration is accessible to users through the *User Profile*. To set a User Profile start a web browser (not HotJava) and go to the following URL:

`http://<mailserver_name >/sims/en/emailuser.html`
where `<locale>` is `en` for English `ja` for Japanese and so on.

You are presented with the following choices:


**FIGURE 10-1** User Profile Update Menu

**Note –** For any changes describes in this section to take effect, an incremental or full directory synchronization must be run. Hence, there may be delay between making the changes and the changes actually taking effect. See "Alias Synchronization Schedule" on page 104.

## ▼ To Change the Mail Password

From browser (don't use HotJava) go to URL:

`http://<mailserver_name >/sims/<locale>/emailuser.html`
where `<locale>` is `en` for English `ja` for Japanese and so on.

Choose *Change your mail password*, enter login name, current mail password, and new mail password.

## ▼ To Start and Stop the Vacation Notice

From browser (don't use HotJava) go to URL:

`http://<mailserver_name >/sims/en/emailuser.html`
where `<locale>` is `en` for English `ja` for Japanese and so on.

Choose *Start, stop, or change vacation notice*, enter login and password to bring up the Vacation Mail Update Page. Enter or modify the following fields and press Apply to start.

- **Enable vacation notice checkbox**. Check this checkbox to start the vacation notice program. Uncheck this box to stop the program.
- **Disable vacation stop date** (optional). Vacation notice program will stop on this date. If not entered, vacation notice will continue until it is manually turned off by unchecking the **Enable vacation notice checkbox**.
- **Subject line for automatically generated replies.** You can type a subject line, or, if you specify $SUBJECT, the subject line of the incoming message will be used.
- **Body for vacation notices to be sent outside your organization**. Text to be sent outside your organization (i.e., different domain names than yours). If you only enter a message for the outside organization, then the message will go to both internal and external senders.
- **Body for vacation notices to be sent inside your organization**. Text for notices sent inside your organization (i.e., same domain name). If you only enter a message for inside organization, then the message will go only to internal senders. External senders will receive no message.

## ▼ To Use Alternative Delivery Programs

Alternative delivery programs include things like auto-reply vacation mailers, mail sorting programs, or other programs that manipulate incoming messages. To start a delivery program, bring up a browser (not HotJava) and go to the following URL:

`http://<mailserver_name >/sims/en/emailuser.html`
where `<locale>` is `en` for English `ja` for Japanese and so on.

Choose *Add, remove, or change Delivery Program options*. For users click on **Delivery Program Option** and enter user login and mail password and press **Apply**. For distributions lists, enter the *common name* of the list and the password to login. The programs available to the user are displayed. Some programs may only be available to users with UNIX accounts while others are available to users with just mail accounts. Programs that can only be run by users with UNIX accounts only are specified by running the `imta program` command with the `-e` argument set to `user`.

After any user profile change is made, it will not take effect until after the next incremental or full alias synchronization.

If the example delivery programs on “To Make Delivery Programs Available to Users” on page 102 were added, FIGURE 10-2 shows what the screen will look like for a user with a UNIX account and login name “unixuser.” FIGURE 10-3 shows the screen for a user without a UNIX account and with the login “nonunixuser.”

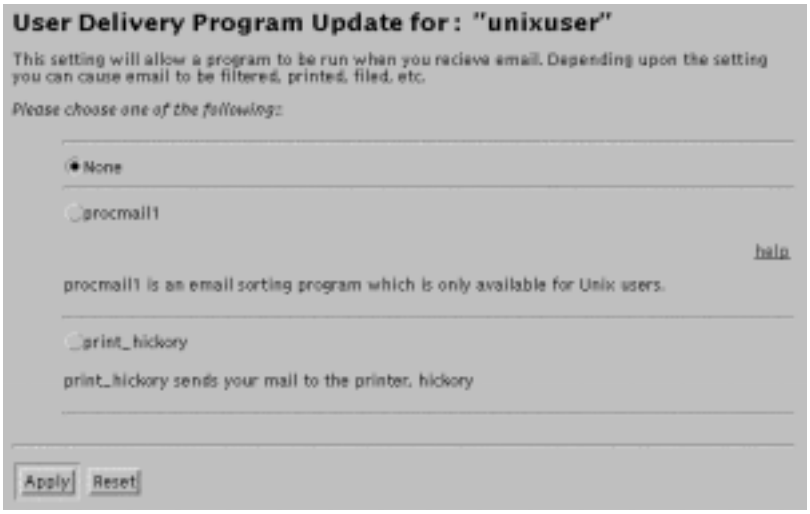



**FIGURE 10-2** User Delivery Programs for User’s with UNIX Accounts

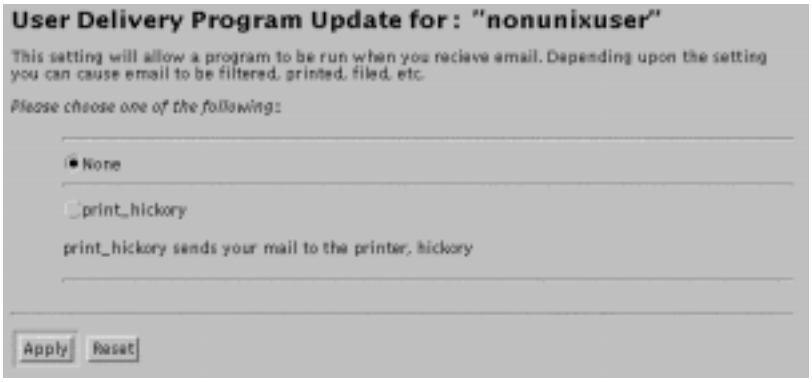
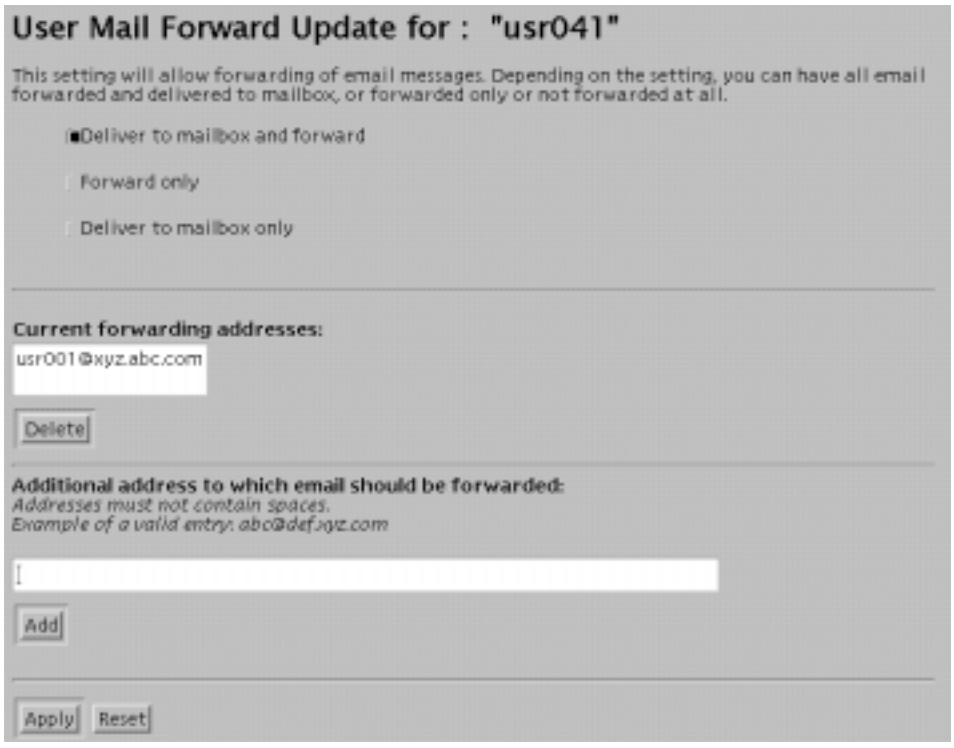



**FIGURE 10-3** User Delivery Programs for Users without UNIX Accounts (procmail was added as a program for UNIX accounts only and is not displayed)

## ▼ To Forward Mail

1. **In the User Profile Update Menu (page 279, don't use HotJava) select Mail Forwarding Option, enter login name and password, and press apply:**



2. **Choose one of the three delivery services.**

   *Deliver to mailbox and forward* puts a copy of the message in the user's mailbox then forwards the message to specified addresses. *Forward only* sends the message to specified addresses. *Deliver to mailbox only* stops all forwarding.

3. **Enter each address to forward messages and press add.**

   Make sure that you add valid email addresses. To delete a forwarding address, highlight the address and press delete.

4. **Press** *Apply* **to save the changes.**

APPENDIX **A**

# Configuring SIMS as a Proxy Message Access Server



## Proxy Message Access Servers Overview

A *proxy message access* server differs from a typically configured SIMS server in that instead of serving the POP/IMAP requests itself, it forwards to the request to the message access server with the requested mail folders (FIGURE A-1). The proxy may or may not have a local message store, but it acts as a virtual message access server by forwarding POP/IMAP requests to the appropriate message access server. Message access proxies are useful for a number of reasons such as horizontal scalability and internet access to private intranet mail systems. These are discussed in "Proxy Server Models" on page 286.

Proxies accept client POP/IMAP requests for mailbox access and authenticates requestor's passwords. The proxy's mail access daemon (`imaccessd`) forwards the request to the appropriate mail access server (i.e., the server containing the desired mailbox). The requestor is again authenticated and the mail access daemon on the server accesses the requested mailbox. At this point, the proxy acts as a simple pipe between the client and the mail server, forwarding whatever one sends to the other, until either the client or the server closes the connection.

Although a proxy server may not have the requested message store, from the client's point of view, the proxy acts like the requested mail access server. Because the proxy communicates with the requested mail server using POP/IMAP, from the requested mail access server's point of view the proxy appears as another mail client.

SIMS message access proxies have two configurations, a *pure proxy,* which acts only as a proxy for SIMS mail servers, and a *message access proxy/message access server* which can act as a proxy for some mail addresses and as a full mail server for local addresses. Note that a pure proxy does **not** have an IMTA installed, and only supports POP and IMAP but not SMTP. An example of these is shown in FIGURE A-1.

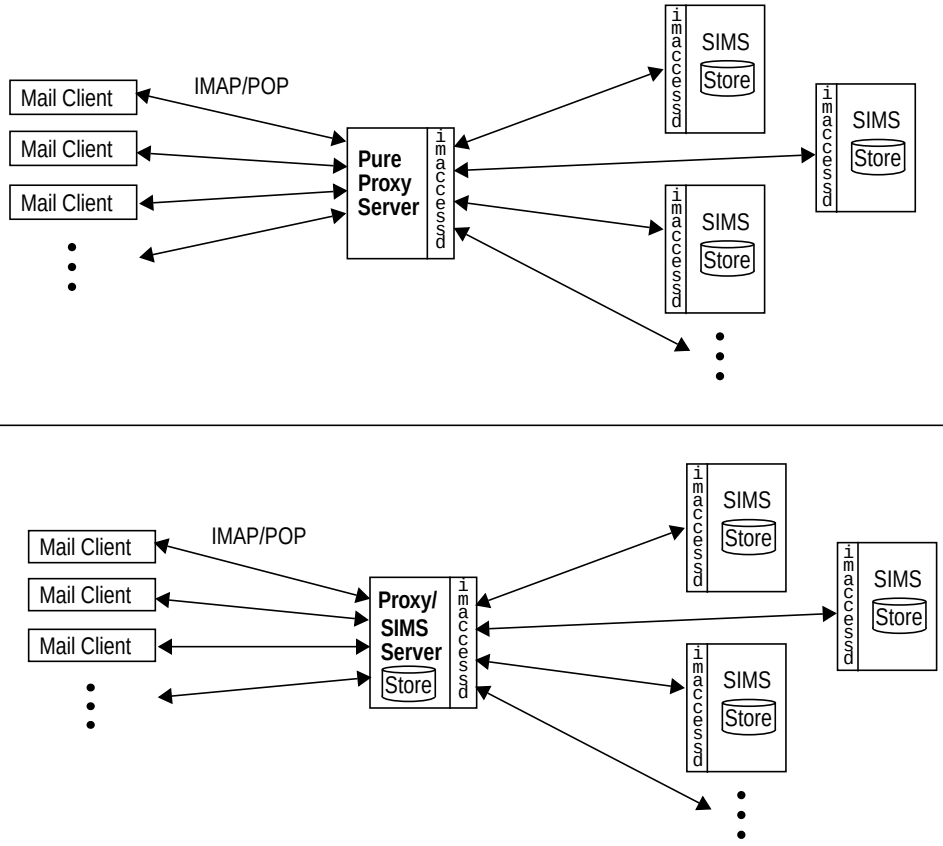



**FIGURE A-1** Pure Proxy Server and Mail Access Proxy / Mail Access Server

# Proxy Server Models

Proxy servers are useful for a number of applications. How you deploy proxy servers depends on the configuration of your email system and what your goals are. This section describes three possible scenarios and models where proxy servers could be used.

# Proxy Servers for Horizontal Scalability

Horizontal Scalability is the ability to expand the capacity of a SIMS environment by adding more servers. Message Access Proxy servers make horizontal scalability possible by having clients point to a single host name which provides access to their mail. Proxies do the work of routing the protocol traffic to and from the appropriate Message Store server. Since proxies allow clients to access their mail folders through a host name which is independent of the actual message store host name, capacity can be added without any burden or reconfiguration on the clients. (For example, having to reconfigure the message access server on each client.)

A single SIMS server cannot support the hundreds of thousands, or even millions of users that Internet Service Providers (ISPs) need to support. Multiple servers are needed to support such populations. Without proxy servers, each user would have to specify their server host name to retrieve mail. By using proxy servers, messages can be accessed through one virtual mail server, while any number of actual mail servers perform actual message storage and retrieval.

By offering only one single virtual mail server, ISPs can add additional mailbox capacity by simply adding more servers behind the proxies.

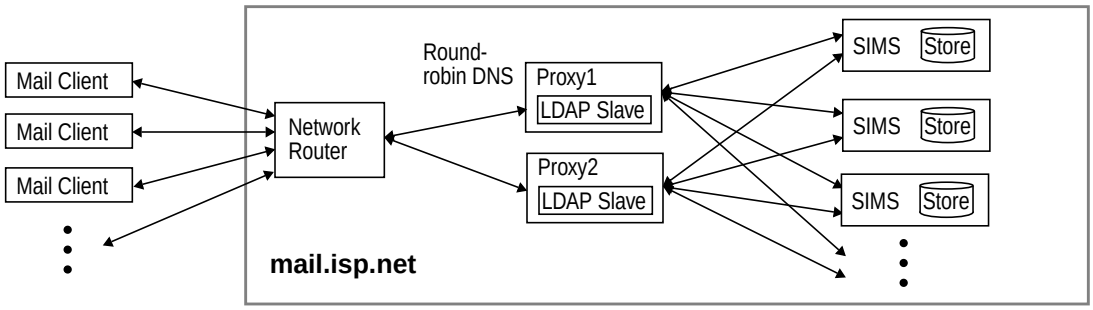



**FIGURE A-2** Proxy Server in an ISP Environment

In the figure above, users log in to the system using the domain name, `mail.isp.net`. Mail requests are routed through the system and sent to a proxy server via round-robin DNS (DNS that can return more than one IP address in round-robin fashion to distribute load among multiple proxy servers). The message access proxy authenticates the user through a replicated LDAP directory, then sends the request to the appropriate message access server. Additional capacity is achieved by adding more message access servers behind the proxies.

This deployment allows for easy expansion of capacity and by virtue of round-robin DNS allows mail access proxies to be treated as field replaceable units. If `mail.isp.net` needs to expand message store capacity to accommodate new customers they can do so either by expanding the capacity of an existing Message

Store server by adding system resources or they can add a new Message Store server. In either case, clients will not be required to change their mail server hostname setting.

## Proxy Servers for the Internet Mail Access

A company that protects its network behind a firewall could, by using a proxy server, allow employees to access their e-mail outside the firewall through the global internet instead of having to maintain a private modem-pool to connect to the private intranet directly.

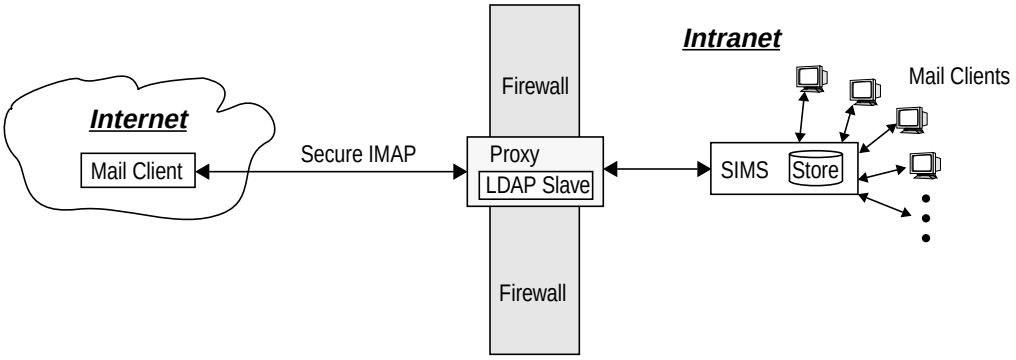



**FIGURE A-3** Proxy Server for Internet Access

In the figure above, an internet mail client accesses his mail through a proxy server on the firewall via a secure IMAP connection. The proxy authenticates the user, then forwards mail store requests to the SIMS. The SIMS then sends message data to the proxy which forwards it to the mail client.

## Proxy Servers for Migrating Users

As an organization grows, additional SIMS servers may be added, and users may be migrated from an old server to a new server. As users are migrated, it would be nice if they could maintain the same server domain address rather than have to adopt a new domain address.

This can be done by activating the message access proxy feature on the SIMS Server. When migrated users make mail access requests, the proxy will forward their requests to the new server. Local users will continue to have their requests at the local host.

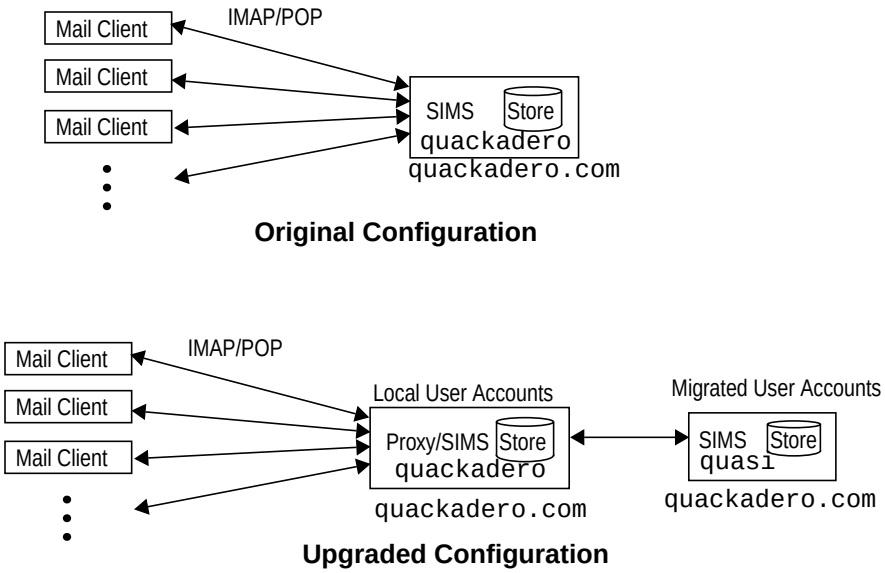



**FIGURE A-4** Proxy / Mail Server for Migrating Users

In FIGURE A-4 the top drawing shows the company's original email configuration. Users connect to a server called *quackadero* and access mail using the domain name `quackadero.com`. In the bottom drawing, quackadero has been converted from a regular mail server to a message access proxy / message access server, and a new server called *quasi* has been added with a number users having been migrated to quasi. However, even though these users are now on quasi, they can still access their mail using the same `quackadero.com` domain name. quackadero provides service for users whose account it supports and forwards mail store requests for users supported by quasi.

# How to Deploy a SIMS Message Access Proxy



The first step in deploying a SIMS proxy model at your site is to choose a model which will address the issues and problems you face. In this section we will describe how to deploy a pure proxy and a proxy / mail server within an organization. We will not specifically describe proxy deployment for horizontal scalability since how this is done will depend upon the platform used for the round-robin DNS which will differ from site to site.

## Setting Up a Pure Proxy

In this section we will describe a generic configuration of a pure message access proxy. Details, such as where in relation to the firewall your proxy is placed or the configuration of a round-robin DNS server for a multiple proxy setup, will not be described.

After proxy is installed, it needs to be configured with the SIMS LDAP directory before it can be operational. The proxy uses the directory to authenticate users and forward requests to the appropriate server. The proxy directory must be designated as a replicated slave to the master SIMS directory located on one of the mail servers. This is depicted in FIGURE A-5.

---

**Note – IMPORTANT!** The master and slave server must have the same replication configuration for the updates to work properly. It is also required that the master and slave have the same schema since replication between servers with dissimilar schemas may lead to unpredictable results. Also, `slapdrepl` must be run on the system with the master server if it is to be replicated. See the SIMS Reference Manual.

---

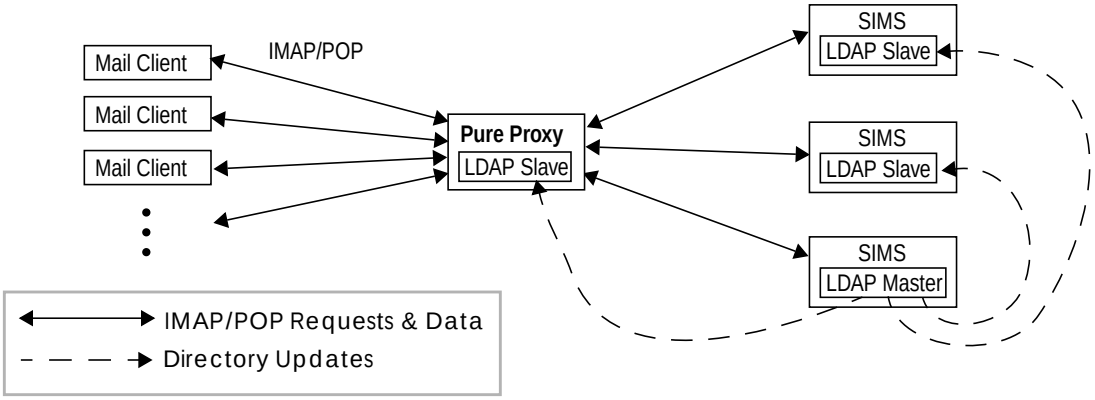



**FIGURE A-5** Proxy Mail System Showing Master to Slave Directory Updates

## ▼ To Configure the Proxy Slave and Master Directories

This section describes how to configure the message access proxy server with a replicated LDAP directory slave server, and to specify to the LDAP directory master server that is has another slave server to support. In this example the fully qualified proxy hostname is called slave1.eng.adagio.com. The master LDAP directory server is called master.eng.adagio.com.

1. **Access the SIMS Proxy LDAP Server Admin Console.**

   On a HotJava browser, go to `http://<proxy_hostname>/sims`

   Log on with your login name (default: admin) and password (default: secret). The SIMS proxy LDAP Server property book appears.

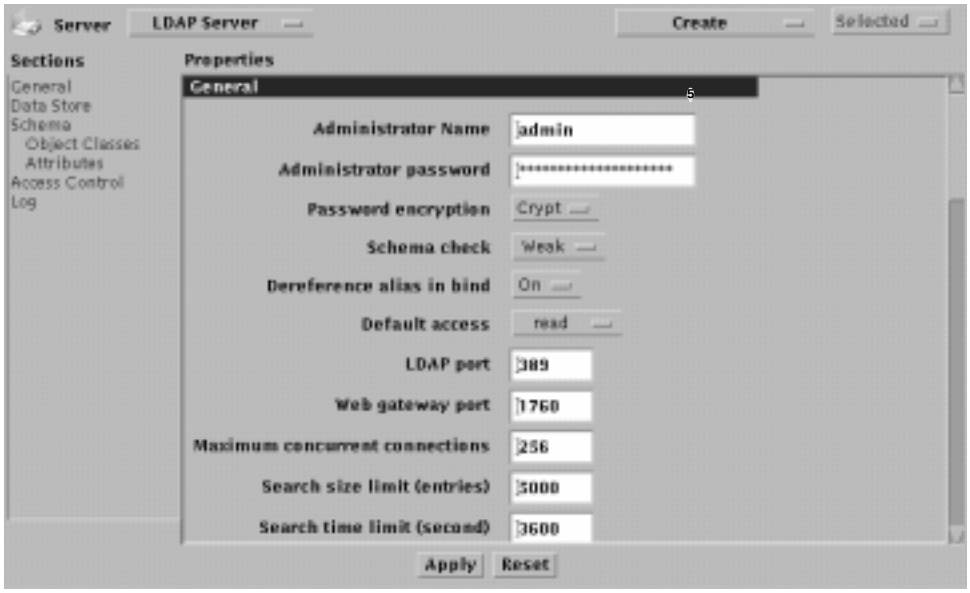



**FIGURE A-6** SIMS Proxy Directory Interface

**2. Click on Data Store.**

Directory information is organized in a tree structure called the *Directory Information Tree (DIT)*. A naming context refers to a particular branch or *subtree* of the DIT. A data store is where directory information is stored in naming contexts.

In this example there is a single data store called: o=adagio,c=US. (The name of the data store is also the same name as the top naming context.)


**FIGURE A-7** Proxy Data Store

3. **Double click the data store name (o=adagio,c=US) to bring up the data store property book. Then click on Naming Contexts section.**

   Two naming contexts are displayed.

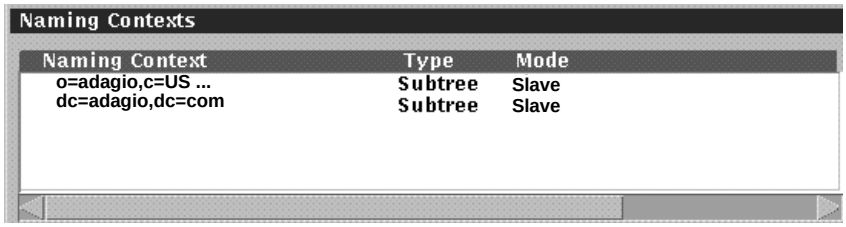



**FIGURE A-8** Proxy Naming Contexts

Although two naming contexts are shown, they actually refer to the same subtree. The top naming context in the figure above is called the OSI tree and the one below is called the Domain Component tree, and they are mapped to each other. The reason for having two naming contexts referring to the same subtree has to do with dual standards. You must configure both as slaves if they are not already configured as such.

4. **Modify the naming contexts to be slaves.**

   Double click the first naming context (OSI subtree).

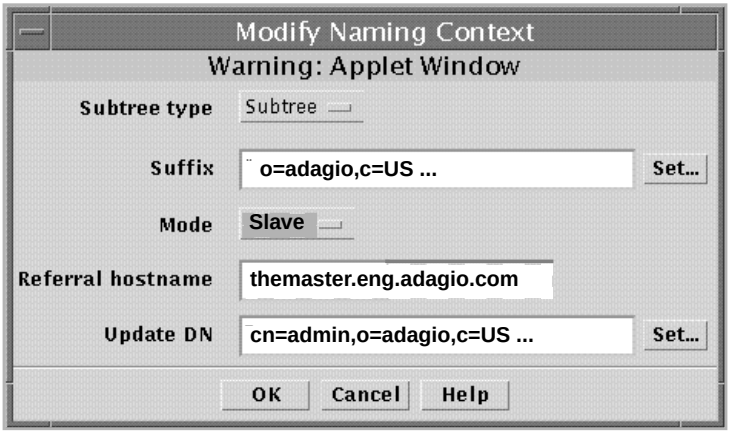



**FIGURE A-9** Proxy Modify Naming Context Window

Subtree type should be left as subtree and the suffix should be left as o=adagio,c=us. Change the mode to Slave. Type in the fully qualified name of the host, where the master server is entered next to Referral hostname.

Update DN is the distinguished name of a user under whom the master server will login to the slave server to modify entries. This DN must have the appropriate ACL to modify entries in the specified suffix of the slave server. Remember this DN because you will have to enter it when you configure the master server, which must be configured to update this new slave. The example shows that Update DN is set to cn=admin,o=adagio,c=us. After making all changes, press OK to save.

Now double click the second naming context (Domain Component subtree). In the Modify Naming Context window, repeat the above steps for the Domain Component subtree (naming context: dc=adagio,dc=com). After making both naming contexts slaves, press Apply on the property book. The directory server on the proxy-only system is now set up as a slave.

The next step is to set up a new replica on the master LDAP server. Many of the steps are similar to the setting up the proxy as a slave directory.

5. **On a HotJava browser, go to the Naming Context section of the Data Store property book on the LDAP master host.**

   Load http://<master_hostname>/sims. Log on with your login name (default: admin) and password (default: secret).

   a. **Click on the Sun Directory Services icon.**

   b. **From the property book, select the Data Store section and double click on the "o=adagio,c=us" naming context.**

6. **Create a directory replica.**

   Select the "Create->Replica" menu option.

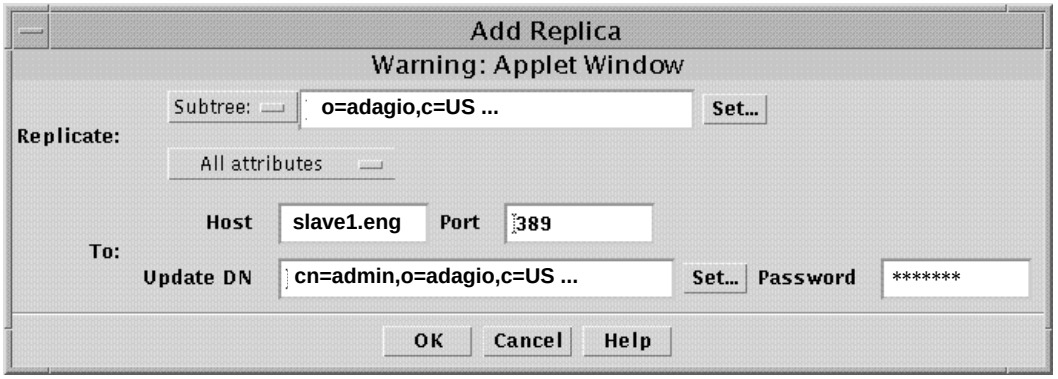



**FIGURE A-10** Add an LDAP Replica from Master Server Admin Console

Select Subtree and fill in the subtree that you want to replicate to the slave server slave1.eng.adagio.com. In our example, we are replicating the entire directory, so type "o=adagio,c=us" in the subtree field. Select All attributes. In the Host field,

enter the fully qualified domain name of the slave server, slave1.eng and the port number on which the slave LDAP server is listening (default=389). Update DN field should be the same name of Update DN that you entered in the slave server (in this example cn=admin,o=adagio,c=us). The password must be the password for the Update DN on the slave. This is the password used by the master server to make updates to the slave server, so make sure that this DN has the appropriate access control permissions for making the changes to the slave.

Since you need updates to both the OSI tree and the Domain component tree, define another replica, this time entering dc=adagio,dc=com in the subtree field.]

After you have created both replicas press Apply at the bottom of the LDAP Data Store Property Sheet. and respond yes to the dialog that follows.


**FIGURE A-11** Applying Modifications to the Data Store.

7. **Execute** `slapdrepl(1M)` **on the system with the master server if it has never been executed on this server before.**

   This command puts the master and replica data stores in the same state so that the replica can receive replication updates from the master. `slapdrepl(1M)` creates an initial replication file and populate the replica using `slurpd`. See the SIMS Reference Manual.

8. **Synchronize the replica and set synchronization schedule.**

   At this point you have set up a LDAP slave and configured its LDAP master to create a replica for the slave. You now need to synchronize the slave directory with the master directory. On the Admin Console, go back to the previous page (LDAP Server property book) and press Data Store.

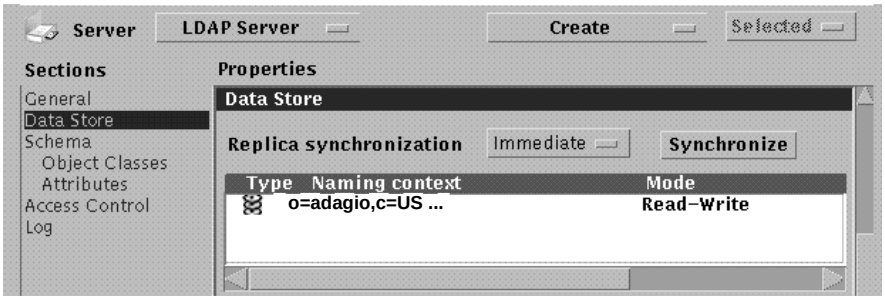



**FIGURE A-12** Applying Modifications to the Data Store.

   At Replica Synchronization, above the Naming Context table, select “immediate” for immediate updates (every time an entry is modified, added, or deleted, the change is sent to the slave) or if you choose Delayed, specify a schedule for the updates. You can also perform a complete synchronization of the master with the slave by pressing the “synchronize” button.

9. **When synchronization occurs, the proxy is operational.**

## Pure Proxy Administration

Because the SIMS pure proxy does not have an Admin Console, administration procedures are performed on the command line. This section describes these procedures.

### ▼ To Change the Maximum Number of Connections on a Proxy

To change the maximum number of connections that can be simultaneously supported on the proxy open the `ims.cnf` file and set parameter `ims-maxconnections` to the desired number. The default is 10,000.

### ▼ To Start/Stop imaccessd

To start imaccessd use the `im.server start` command. To stop use `im.server stop`.

### ▼ To Configure IMAP Capabilities in the Proxy

**Note –** Read this section if you are configuring a SIMS proxy with a non-SIMS back end mail server.

`CAPABILITY` is an IMAP command that lists commands in addition to the standard (RFC2060) commands that a given server will support. Since `CAPABILITY` is valid even before the client has been authenticated (capabilities can include authentication mechanisms), the proxy has no way of knowing in advance to which server the user will be connected to, and therefore can't list the capabilities supported by this server.

So, when the proxy is enabled in imaccessd, the only capabilities that will be returned to the client when `capability` is executed are:

```
* CAPABILITY IMAP4 IMAP4rev1
```

plus the authentication mechanisms supported by the proxy.

This means that all the remote server(s) MUST support IMAP4 and IMAP4rev1. If you have servers connected to the proxy that do not support both protocols, or, if you need to have the proxy advertise capabilities supported by the real servers, then you need to define the parameter `ims-caps-proxy` in `ims.cnf` that will contain these capabilities. This can also be done in the Admin Console (see "To Configure IMAP Capabilities in the Proxy" on page 297).

This parameter, if absent, is equivalent to `IMAP4 IMAP4rev1`. You can disable either IMAP4 or IMAP4rev1 if the back end server doesn't support both, or you can add new capabilities to the list.

One caveat: some additional capabilities include commands that are supported once the client is authenticated (example: the `SCAN` command in SIMS). There is no harm in advertising these in the proxy since the client can only issue them at a time the real server will receive and process them. However, for some extensions that enable a behavior of the server (such as `IMAP4SUNVERSION` in SIMS), it is not recommended that you add these to the list because the client could send the command before authentication is completed, and the proxy server would not forward the command to the real server.

## Setting Up a Proxy / Mail Server

This section describes how to convert a SIMS mail server to a message access proxy / message access server. We will use the hypothetical example of an administrator migrating a number of users on the original SIMS mail server to a new SIMS server, and converting the original SIMS mail server to a proxy / message access server (FIGURE A-4 on page 289). The basic steps for completing these steps are as follows:

1. Install new SIMS machine.
2. Convert the old SIMS machine to a proxy / mail server.
3. Establish the new machine a slave LDAP server to master server on the old machine.
4. Disable mailbox accounts of users who are to be migrated so that mail sent to these users will be put on hold until the accounts are restarted.
5. Using the command imbackup and imrestore, migrate the users from the old machine to the new machine.
6. Change the `Mailhost` attribute in the entries of the migrated users
7. Enable the disabled mailbox accounts so that they can receive mail.
8. Delete all the migrated accounts from the old machine.

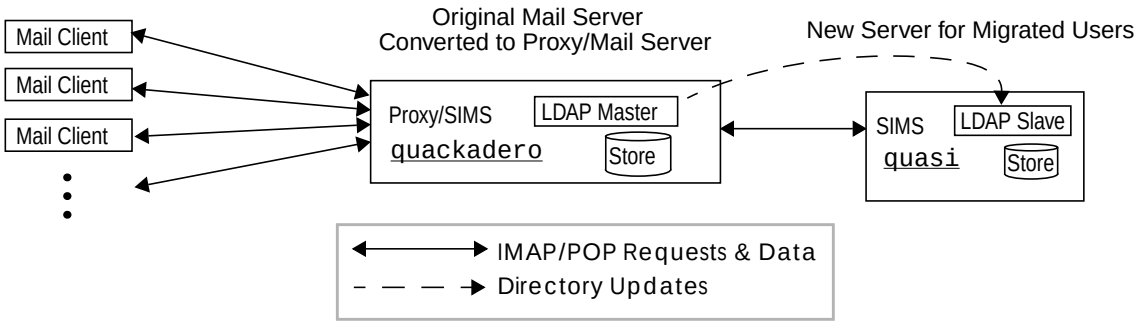



**FIGURE A-13** Converting Mail Server to Proxy / Mail Server.

## ▼ To Migrate Users by Converting a Mail Server to a Proxy / Mail Server

In this example the original mail server to be converted to a proxy / mail server is called *quackadero*, and the new pure mail server is called *quasi*. The company is *Adagio*.

1. **Install new SIMS server.**
2. **Convert the old SIMS mail server to a proxy/mail server.**

   Bring up the Admin Console on the old SIMS machine. Go to the Advanced Options section in the Message Store property book. Press Proxy Server On button. Also, set the IMAP Server Capabilities. This is typically `IMAP4 IMAP4rev1`, which you can

enter by pressing the Default button (see “To Configure IMAP Capabilities in the Proxy” on page 297 for additional information). You must stop then start the message access protocols (see “Message Access Protocol Connections” on page 161).

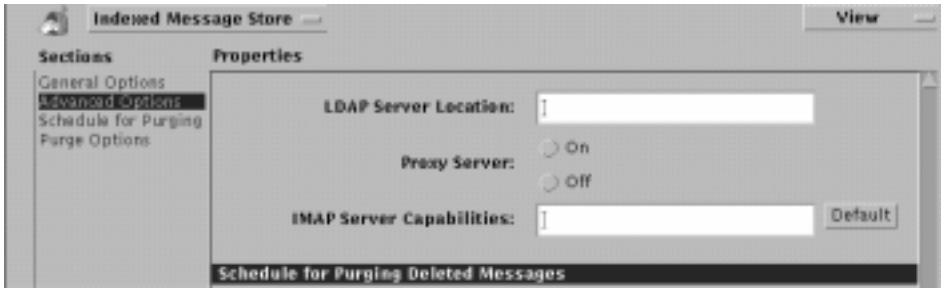



**FIGURE A-14** Admin Console (Sun Message Store->Advanced Options)

3. **Configure the new machine as an LDAP slave server and the old machine (now proxy/mail server) as an LDAP master server.**

   Follow the instructions described in “To Configure the Proxy Slave and Master Directories” on page 291.

4. **Disable mailbox accounts of users who are to be migrated so that mail sent to these users will be put on hold until the accounts are restarted.**

---

**Note –** After completing this step, mail will not be delivered to the message queue and clients of this IMTA will not be able to send mail. Messages sent to these users will be bounced back to the sender or held at another MTA. The IMTA will be disabled until you run `imta start dispatcher` (step 8).

---

   a. **Shutdown the IMTA dispatcher.**

      ```
      # imta stop dispatcher
      ```

   b. **Deliver all queued mail.**

      Run `imta cache -view` to view entries for queued mail.

      Run `imta submit [channel name]` to queue mail.

5. **Migrate mailboxes from one machine to another.**

   a. **Identify users to be moved to the new machine and have these users log out of their mail clients.**

      They must not use their mail client until after migration is complete.

**b. Use** `imbackup -f bak -u username_file` **to backup the mailboxes to be migrated.**

`bak` is the name of the file in which to back up the mailboxes. `username_file` is a file containing a list of user names to be migrated. Each name must be separated by spaces, tabs, or carriage returns. See the *SIMS Reference Manual* for details.

**c. Use** `imrestore -t3 -f bak -u username_file` **to restore the backed up mailboxes to the new SIMS machine.**

**6. Change the** `Mailhost` **attribute in the entries of the migrated users.**

You can do this from the Admin Console by modifying a group entry. See To Modify a Group Entry, Step 2, Section c. You can also use the ldapmodify command if you prefer to do this in a UNIX script. See the *SIMS Reference Manual* for ldapmodify details.

**7. Delete all the migrated accounts from the old machine.**

Log in to the old machine. As root, use `imdeluser -u <username_file>`. Do not run this command on the new machine or you will delete all your migrated users. See the *SIMS Reference Manual* for details.

**8. Enable the disabled mailbox accounts so that they can receive mail.**

**a. Synchronize the alias databases.**

```
# imta dirsync -F
```

**b. Start the IMTA dispatcher.**

```
# imta start dispatcher
```

Proxy / mail server should be running as planned.

APPENDIX **B**

# Replication Configuring—Examples



This section describes setting up the replication feature in three common scenarios. It by no means attempts to describe setting up replication in each imaginable scenario, but rather in scenarios that Sun anticipates a majority of customer sites will use.

## Example 1 - Replicating Data From A Master Server To One Replica Server

This section walks you through the process of setting up a replication scenario in which data is replicated from a master server to one replica server. (A *master server* contains the data that is to be replicated; a *replica server* (also called a *slave server*) receives the replicated data from the master server.) This replication example assumes the following:

- A data store was created during the installation of the mail server software and its subsequent initialization.
- All attributes will be replicated rather than customizing which attributes will or will not be replicated.

1. **Bring up the Admin console on the slave host (the host which will hold a replicated LDAP directory).**

   In this example the slave host is called slave1. Load `http://slave1.eng/sims`

2. **Click on Sun Directory Server to bring up the LDAP Server Property Book.** **(**FIGURE B-1**)**

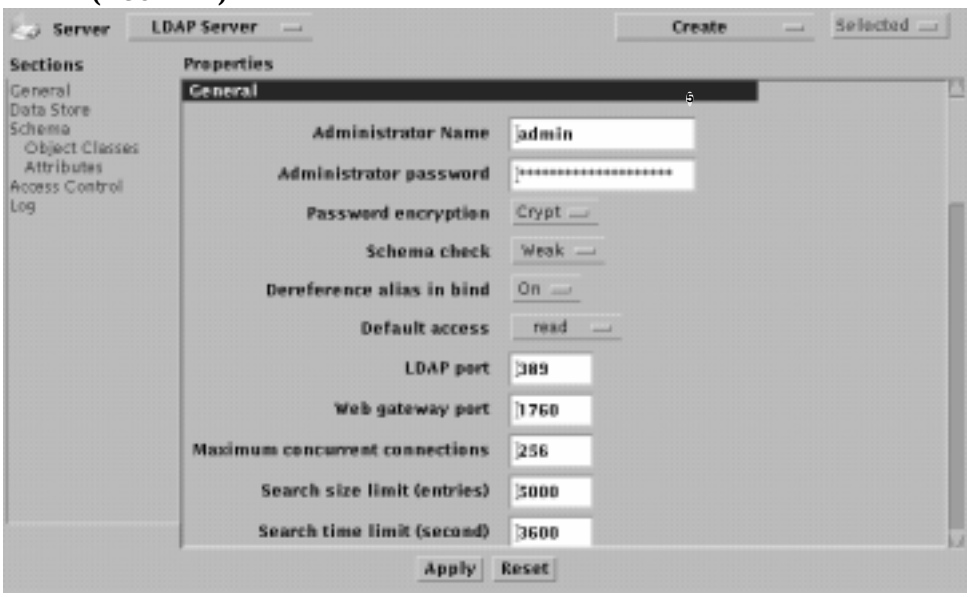



**FIGURE B-1** SIMS Directory Interface

3. **Click on Data Store.**

Directory information is organized in a tree structure called the *Directory Information Tree (DIT)*. A naming context refers to a particular branch or *subtree* of the DIT. A data store is where directory information is stored in naming contexts.

In this example there is a single data store called: `o=adagio,c=US` (The name of the data store is also the same name as the top naming context.)


**FIGURE B-2** Data Store

4. **Double click the data store name (`o=adagio,c=US`) to bring up the data store property book. Then click on Naming Contexts section.**

   Two naming contexts are displayed.

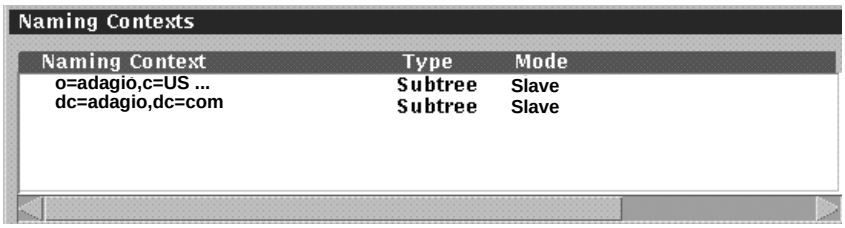



**FIGURE B-3** Naming Contexts

Although two naming contexts are shown, they actually refer to the same subtree. The top one is called the OSI tree and the bottom one is called the Domain Component tree, and they are mapped to each other. The reason for having two naming contexts referring to the same subtree has to do with dual standards. You must configure both as Slaves if they are not already configured as such.

5. **Modify the naming contexts to be slaves.**

   Double click the first naming context (OSI subtree) `o=adagio,c=US`.

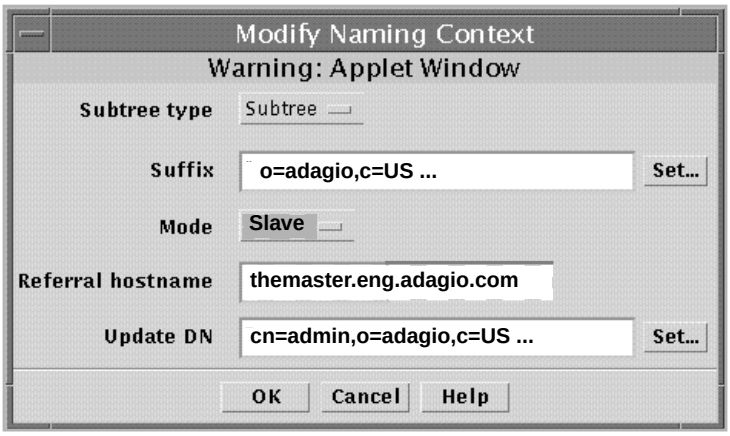



**FIGURE B-4** Modify Naming Context Window

Leave Subtree type as Subtree and leave suffix as `o=adagio,c=US`—unless you wish to only replicate a branch of the such as `ou=eng,o=adagio,c=US` or `ou=mktg,o=adagio,c=US`. Change the mode to Slave. Next to Referral hostname, enter the fully qualified name of the master LDAP directory host is entered.

Update DN is the distinguished name of a user under whom the master server will login to the slave server to modify entries. This DN must have the appropriate ACL to modify entries in the specified suffix of the slave server. Remember this DN because you will have to enter it when you configure the master server, which must be configured to update this new slave. The example shows that Update DN is set to `cn=admin,o=adagio,c=US`. After making all changes, press OK to save.

Now double click the second naming context (Domain Component or DC subtree) `dc=adagio,dc=com`. In the Modify Naming Context window, repeat the above steps for the Domain Component subtree (naming context: `dc=adagio,dc=com`). After making both naming contexts slaves, press Apply on the property book. The directory server on the replicated system is now set up as a slave.

The next step is to set up a new replica on the master LDAP server. Many of the steps are similar to the setting up the slave server.

6. **On a HotJava browser, go to the Naming Context section of the Data Store property book on the LDAP master host.**

   Load `http://themaster.eng/sims`, go to Sun Directory Services->Data Store, and double click on the Data Store "`o=adagio,c=US`"

7. **Create a directory replica for the OSI data tree.**

   Select the Create->Replica menu option.

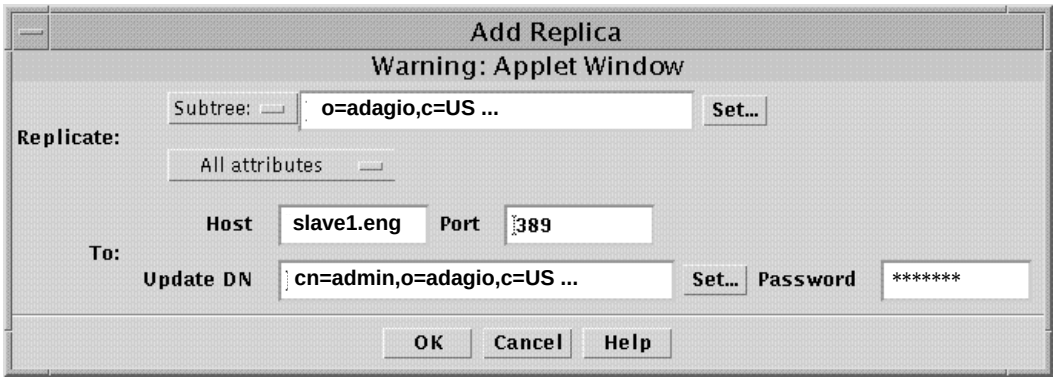



**FIGURE B-5** Add an LDAP Replica from Master Server Admin Console

Set Subtree to naming contest that you want replicated to the slave server (`slave1.eng.adagio.com`). In our example, we are replicating the entire directory, so type "`o=adagio,c=US`" in the subtree field. If you only wish to replicate a branch of the tree such as `ou=eng,o=adagio,c=US` or `ou=mktg,o=adagio,c=US`, then enter that as the naming context.

Select All attributes. In the Host field, enter the fully qualified domain name of the slave server (slave1.eng) and the port number on which the slave LDAP server is listening (default=389). Update DN field should be the same name of Update DN

that you entered in the slave server (in this example `cn=admin,o=adagio,c=US`). The password must be the password for the Update DN on the slave. This is the password used by the master server to make updates to the slave server, so make sure that this DN has the appropriate access control permissions for making the changes to the slave.

8. **Create a directory replica for the DC data tree.**

   Since you need updates to both the OSI tree and the Domain component tree, define another replica, this time entering `dc=adagio,dc=com` in the subtree field and `dc=admin,dc=adagio,dc=com` in the Update DN field.

9. **Apply the replica modifications.**

   After you have created both replicas press Apply at the bottom of the LDAP Data Store Property Sheet. and respond yes to the dialog that follows.


**FIGURE B-6** Applying Modifications to the Data Store.

10. **Synchronize the replica and set synchronization schedule.**

    At this point you have set up a LDAP slave and configured its LDAP master to create a replica for the slave. You now need to synchronize the slave directory with the master directory. On the Admin Console, go back to the previous page (LDAP Server property book) and press Data Store.

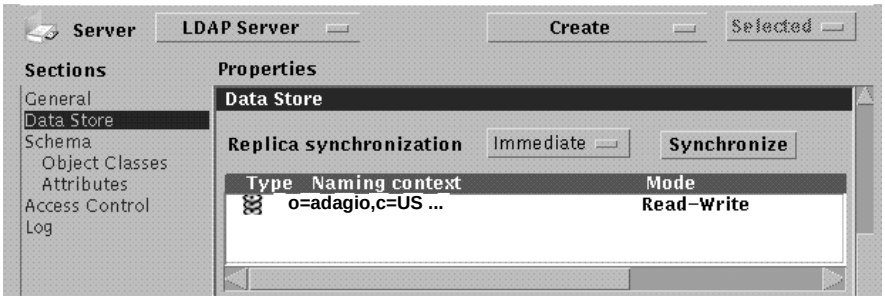



**FIGURE B-7** Applying Modifications to the Data Store.

At Replica Synchronization, above the Naming Context table, select “immediate” for immediate updates (every time an entry is modified, added, or deleted, the change is sent to the slave) or if you choose Delayed, specify a schedule for the updates. You can also perform a complete synchronization of the master with the slave by pressing the “synchronize” button.

11. **When synchronization occurs, the slave server is operational.**

# Example 2 - Replicating Data From A Master Server To Two Replica Servers

This section walks you through the process of replicating data from a master server to two replica servers. (A *master server* contains the data that is to be replicated; a *replica server* receives the replicated data from the master server.) This replication example assumes the following:

- A data store was created during the installation of the mail server software and its subsequent initialization.
- All attributes will be replicated rather than customizing which attributes will or will not be replicated.

1. **Bring up the Admin console on the first slave host (the host which will hold a replicated LDAP directory).**

   In this example the slave host is called slave1. Load `http://slave1.eng/sims`

2. **Click on Sun Directory Server to bring up the LDAP Server Property Book.** **(**FIGURE B-1**)**

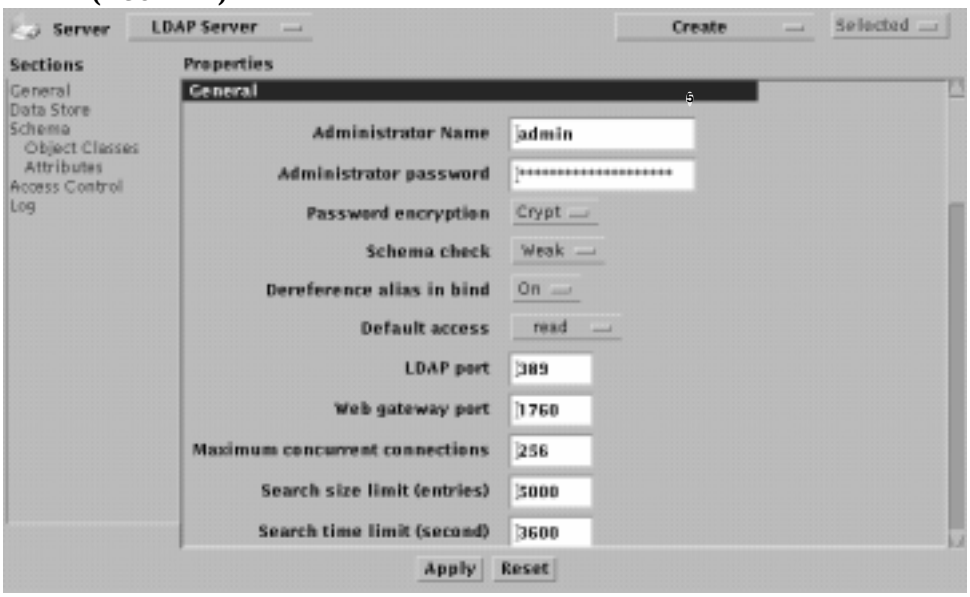



**FIGURE B-8** SIMS Directory Interface

3. **Click on Data Store.**

Directory information is organized in a tree structure called the *Directory Information Tree (DIT)*. A naming context refers to a particular branch or *subtree* of the DIT. A data store is where directory information is stored in naming contexts.

In this example there is a single data store called: `o=adagio,c=US` (The name of the data store is also the same name as the top naming context.)


**FIGURE B-9** Data Store

4. **Double click the data store name (`o=adagio,c=US`) to bring up the data store property book. Then click on Naming Contexts section.**

   Two naming contexts are displayed.

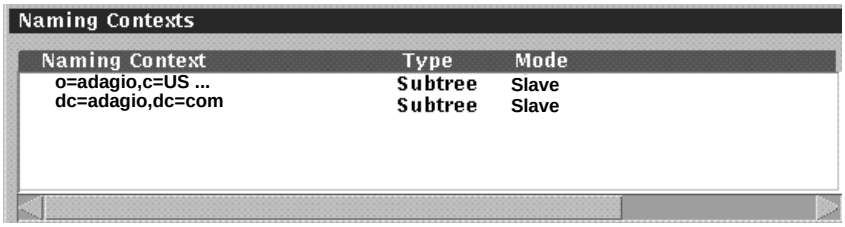



**FIGURE B-10** Naming Contexts

Although two naming contexts are shown, they actually refer to the same subtree. The top one is called the OSI tree and the bottom one is called the Domain Component tree, and they are mapped to each other. The reason for having two naming contexts referring to the same subtree has to do with dual standards. You must configure both as Slaves if they are not already configured as such.

5. **Modify the naming contexts to be slaves.**

   Double click the first naming context (OSI subtree) `o=adagio,c=US`.

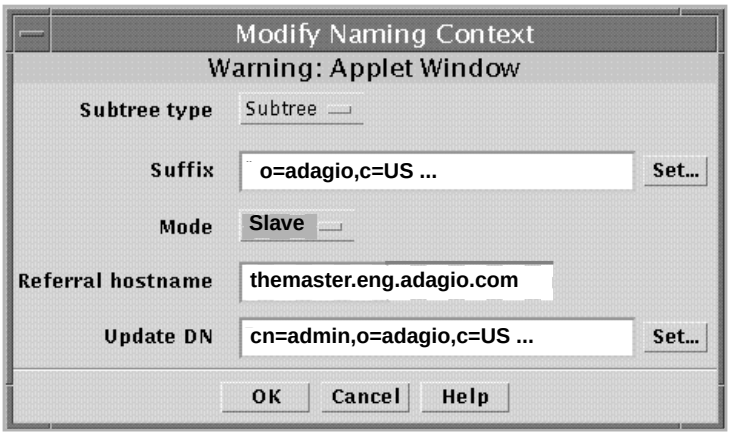



**FIGURE B-11** Modify Naming Context Window

Leave Subtree type as Subtree and leave suffix as `o=adagio,c=US`, unless you wish to only replicate a branch of the such as `ou=eng,o=adagio,c=US` or `ou=mktg,o=adagio,c=US`. Change the mode to Slave. Next to Referral hostname, enter the fully qualified name of the master LDAP directory host is entered.

Update DN is the distinguished name of a user under whom the master server will login to the slave server to modify entries. This DN must have the appropriate ACL to modify entries in the specified suffix of the slave server. Remember this DN because you will have to enter it when you configure the master server, which must be configured to update this new slave. The example shows that Update DN is set to `cn=admin,o=adagio,c=US`. After making all changes, press OK to save.

Now double click the Domain Component subtree naming context (`dc=adagio,dc=com`). In the Modify Naming Context window, repeat the above steps for the Domain Component subtree (naming context: `dc=adagio,dc=com`). After making both naming contexts slaves, press Apply on the property book. The directory server on the replicated system is now set up as a slave.

6. **Repeat the same steps on the second slave server (slave2.eng).**

   `http://slave2.eng/sims`

   After setting up both slave servers, set up two new replicas on the master LDAP server. Many of the steps are similar to the setting up the slave server.

7. **On a HotJava browser, go to the Naming Context section of the Data Store property book on the LDAP master host.**

   Load `http://themaster/sims`

   a. **Click on the Sun Directory Services icon.**

   b. **From the property book, select the Data Store section and double click on the Data Store “`o=adagio,c=US`”.**

8. **Create a directory replica for the first slave server.**

   Select the “Create->Replica” menu option.

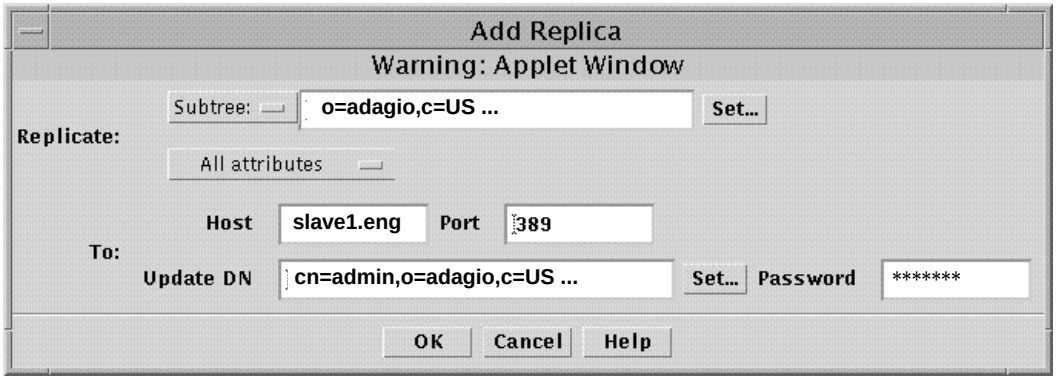



**FIGURE B-12** Add an LDAP Replica from Master Server Admin Console

Set Subtree to the naming context that you want replicated to the slave server (slave1.eng.adagio.com). In our example, we are replicating the entire directory, so type “`o=adagio,c=US`” in the subtree field. Select All attributes. If you only wish to replicate a branch of the tree such as `ou=eng,o=adagio,c=US` or `ou=mktg,o=adagio,c=US`, then enter that as the naming context. In the Host field, enter the fully qualified domain name of the slave server (slave1.eng) and the port number on which the slave LDAP server is listening (default=389). Update DN field should be the same name of Update DN that you entered in the slave server (in this example `cn=admin,o=adagio,c=US`). The password must be the password for the Update DN on the slave. This is the password used by the master server to make updates to the slave server, so make sure that this DN has the appropriate access control permissions for making the changes to the slave.

Since you need updates to both the OSI tree and the Domain component tree, define another replica, this time entering `dc=adagio,dc=com` in the subtree field, and `dc=admin,dc=adagio,dc=com` in the Update DN field.

9. **Create a directory replica for the second slave server.**

   Follow the same instructions as described in the previous step for slave2.eng.

10. **Apply the replica modifications.**

After you have created both replicas press Apply at the bottom of the LDAP Data Store Property Sheet and respond yes to the dialog that follows.


**FIGURE B-13** Applying Modifications to the Data Store.

**11. Synchronize the replica and set synchronization schedule.**

At this point you have set up a LDAP slave and configured its LDAP master to create a replica for the slave. You now need to synchronize the slave directory with the master directory. On the Admin Console, go back to the previous page (LDAP Server property book) and press Data Store.

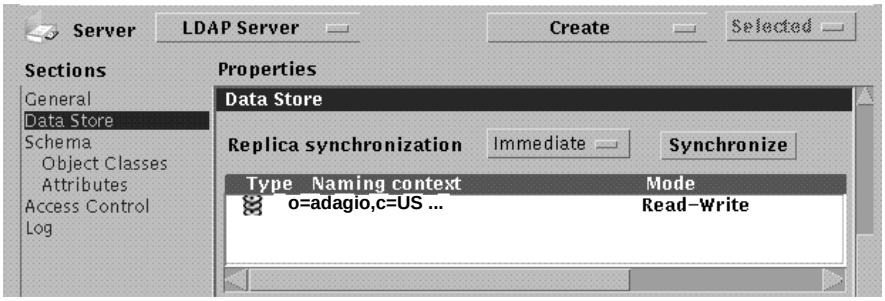



**FIGURE B-14** Applying Modifications to the Data Store.

At Replica Synchronization, above the Naming Context table, select “immediate” for immediate updates (every time an entry is modified, added, or deleted, the change is sent to the slave) or if you choose Delayed, specify a schedule for the updates. You can also perform a complete synchronization of the master with the slave by pressing the “synchronize” button.

**12. When synchronization occurs, the slave server is operational.**

# Example 3 - Bidirectional Replication

You can configure one directory server to act as both a master server and a slave server. A *master server* contains the data that is to be replicated; a *slave server* receives the replicated data from the master server.

In this example we have a company, Adagio, with two domains and two servers in different locations. One is in Texas (domain: texas.adagio.com, host: yellowrose) and the other in California (domain: calif.adagio.com, host: surfergirl). The directory information tree for Adagio is as follows:

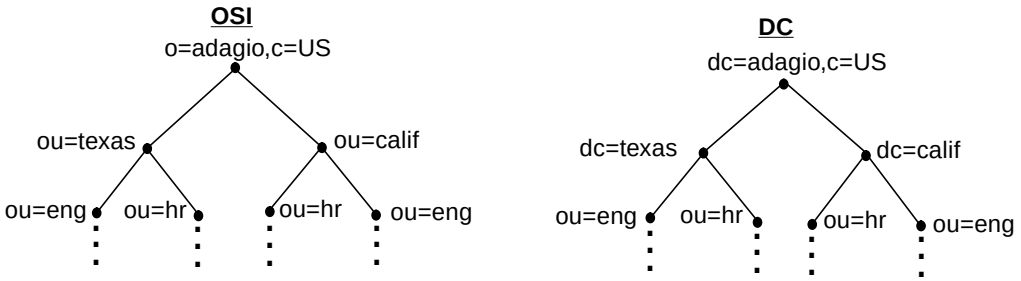



**FIGURE 10-4** DIT Structure for Adagio Corporation

In each domain, we want the local servers to have a complete copy of the SIMS LDAP directory. However, on yellowrose we want the naming context `ou=texas,o=adagio,c=US` to be master, and the naming context `ou=calif,o=adagio,c=US` be a replicated slave. On surfergirl we want the naming context `ou=texas,o=adagio,c=US` to be a replicated slave, and the naming context `ou=calif,o=adagio,c=US` be the master.

1. **Bring up the Admin console on the yellowrose.**

   Log in to `http://yellowrose.texas/sims`

2. **Click on Sun Directory Server to bring up the LDAP Server Property Book.** **(**FIGURE B-1**)**

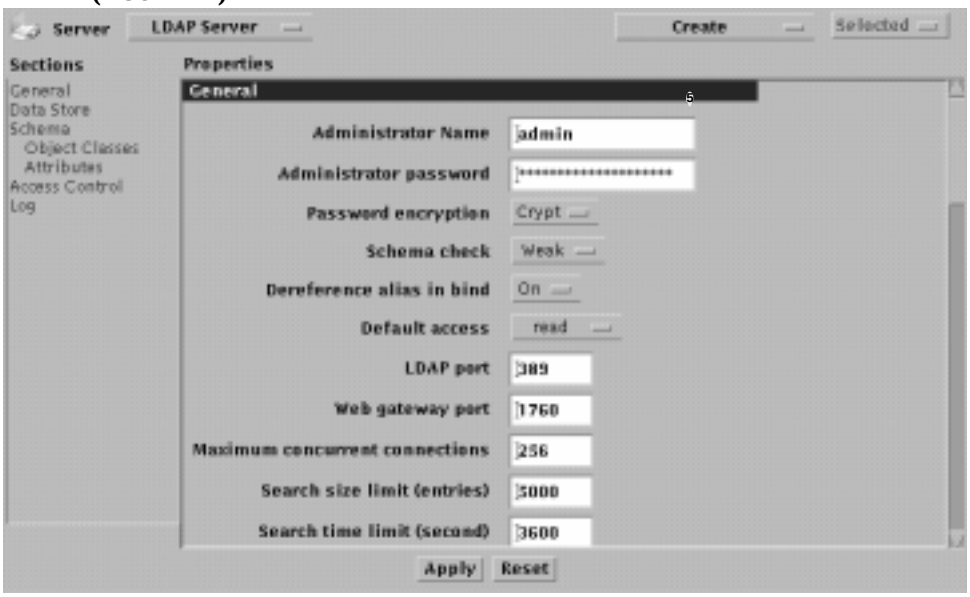



**FIGURE B-15** SIMS Directory Interface

3. **Click on Data Store.**

In this example there is a single data store called: `o=adagio,c=US` (The name of the data store is also the same name as the top naming context.)


**FIGURE B-16** Data Store

4. **Double click the data store name (`o=adagio,c=US`) to bring up the data store property book. Then click on Naming Contexts section.**

   Two naming contexts are displayed.

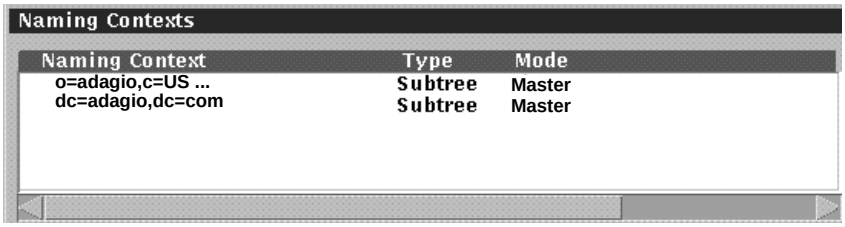



**FIGURE B-17** Naming Contexts

   Although two naming contexts are shown, they actually refer to the same subtree. The top one is called the OSI tree and the bottom one is called the Domain Component tree, and they are mapped to each other. The reason for having two naming contexts referring to the same subtree has to do with dual standards.

5. **Modify the OSI naming context to be `ou=texas,o=adagio,c=US`.** Designate it to **be a master.**

   Double click the first naming context (OSI subtree). Leave Subtree type as Subtree and the mode as Master, however change the suffix from `o=adagio,c=US` to `ou=texas,o=adagio,c=US`. After making all changes, press OK to save.

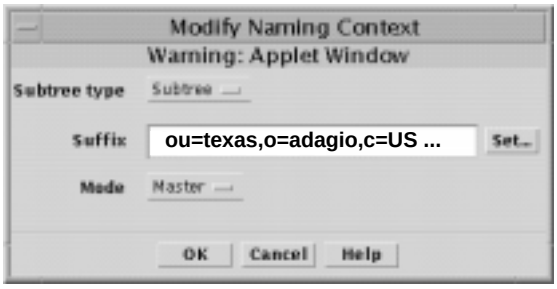



**FIGURE B-18** Modify Naming Context Window

6. **Modify the Domain Component (DC) naming context to be `dc=texas,dc=adagio,dc=com`. Designate it to be a master.**

   Double click the DC naming context `dc=adagio,dc=com`. In the Modify Naming Context window, repeat the above steps for the DC subtree. Change dc=adagio,dc=com to `dc=texas,dc=adagio,dc=com`. After making all changes, press OK to save.

7. **Create OSI naming context called `ou=calif,o=adagio,c=US`. Designate it to be a slave.**

Click on Create—>Naming Context.

The Add Naming Context window appears. Change the mode from Master to Slave. Enter the name of the new naming context. Enter the master hostname for the naming context next to Referral hostname.

Update DN is the distinguished name of a user under whom the master server will login to the slave server to modify entries. This DN must have the appropriate ACL to modify entries in the specified suffix of the slave server. Remember this DN because you will have to enter it when you configure the master server, which must be configured to update this new slave. The example shows that Update DN is set to `cn=admin,o=adagio,c=US`. After making all changes, press OK to save.

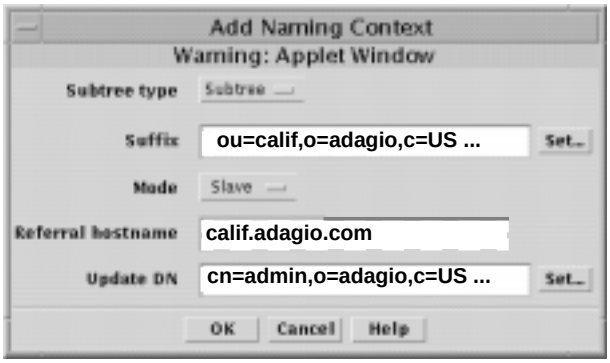



**FIGURE B-19** Modify Naming Context Window

**8. Create DC naming context called `dc=calif,dc=adagio,dc=com`. Designate it to be a slave.**

Click on Create—>Naming Context. Use the following parameters:

Suffix: `dc=calif,dc=adagio,dc=com`
Referral hostname: `calif.adagio.com`
Update DN: `dc=admin,dc=adagio,dc=com`

After making all changes, press OK to save. The Naming Context section looks like this

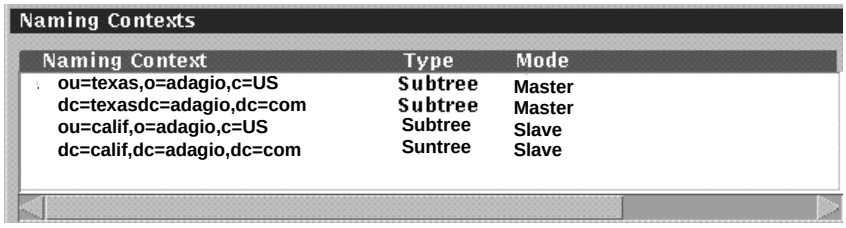



**FIGURE B-20** Completed Naming Context Window for yellowrose

9. **Create a directory replica for `ou=texas,o=adagio,c=US`.**

   Select the “Create->Replica” menu option. Select Subtree and fill in the subtree that you want to replicate, i.e., `ou=texas,o=adagio,c=US`. Select All attributes. In the Host field, enter the fully qualified domain name of the slave server (surfergirl.eng) and the port number on which the slave LDAP server is listening (default=389). Update DN field should be the same name of Update DN that you entered in the slave server (in this example `cn=admin,o=adagio,c=US`). The password must be the password for the Update DN on the slave. This is the password used by the master server to make updates to the slave server, so make sure that this DN has the appropriate access control permissions for making the changes to the slave.

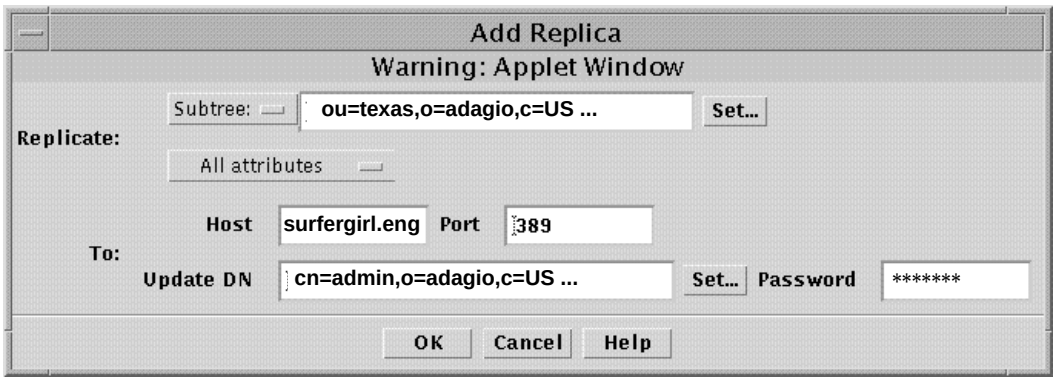



**FIGURE B-21** Add an LDAP Replica from Master Server Admin Console

10. **Create a directory replica for `dc=texas,dc=adagio,dc=com`**

    You need updates for the DC tree as well as the OSI tree. Define another replica using the procedures described in Step 9 with the following parameters:

    Subtree: `dc=texas,dc=adagio,dc=com`
    Host: `surfergirl.eng`
    Update DN: `dc=admin,dc=adagio,dc=com`

11. **Press Apply on the property book.**

    The master and slave directory servers are now set up on host yellowrose.

12. **On host surfergirl, set up slave and master naming contexts.**

    Use the same procedure described from Step 1 through Step 8 on surfergirl. However, make the naming context `ou=texas,o=adagio,c=US` to be a replicated slave and `ou=calif,o=adagio,c=US` be the master. The Naming Context section look as follows.

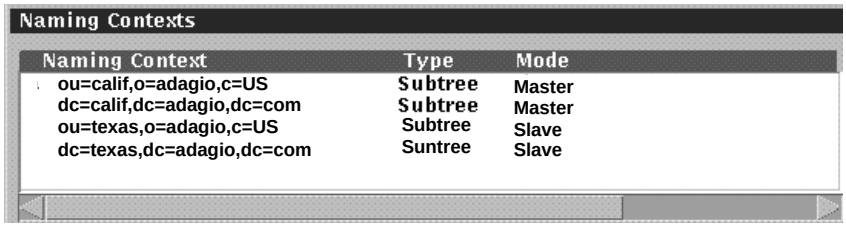



**FIGURE B-22** Completed Naming Context Window for surfergirl

13. **Create directory replicas for `ou=calif,o=adagio,c=US` and `dc=calif,dc=adagio,dc=com`.**

    Host: `yellowrose.eng`
    Update DN: `dc=admin,dc=adagio,dc=com` (DC) and `cn=admin,o=adagio,c=US` (OSI)

14. **Press Apply on the property book.**

    The master and slave directory servers are now set up on host yellowrose.

15. **Synchronize both replicas and set synchronization schedule.**

    At this point you have set up your bidirectional LDAP slave and LDAP master servers. You now need to synchronize the slave directories with the master directories. Log onto the Admin Consoles of both yellowrose and surfergirl, go to LDAP Server property book and press Data Store.

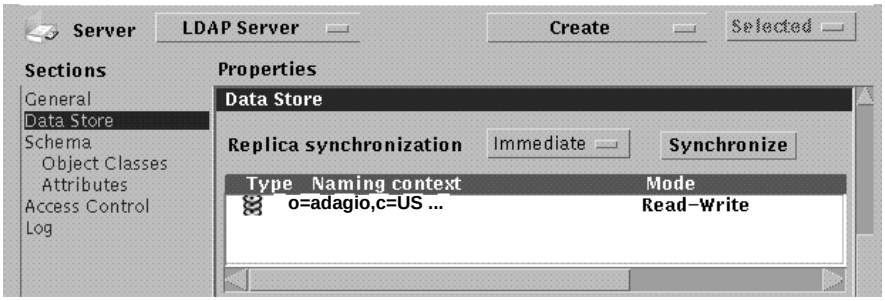



**FIGURE B-23** Applying Modifications to the Data Store.

Next to Replica Synchronization select Immediate for immediate directory updates (every time an entry is modified, added, or deleted, change is sent to the slave) or choose Delayed and specify an update schedule. You can also perform a complete synchronization of the master with the slave by pressing the “synchronize” button.

16. **When synchronization occurs, the bidirectional replicas will be operational.**

APPENDIX **C**

# Populating the Directory Examples



## Populating the Directory with User Data—Sample Session

Alpha Corporation is setting up a pilot test of the directory with two users on a lab machine called `testserver`. The test machine uses NIS+, and has the following users defined:

```
jdoe:fWFuXyZ1S..Vk:1001:10:John Doe:/export/home/jdoe:/bin/sh
gevert:fWFuXyZ1S..Vk:1002:10:Gail Evert:/export/home/gevert:/bin/sh
```

To create directory entries for these users, complete the following steps:

1. **Log in as root.**

```
$ su
Password: <Enter your root password>
#
```

2. **Use the** `getent` **command to save the user entries in a file:**

```
# getent passwd > /tmp/passwd
```

3. **Use the** `niscat` **command to extract user information from the mail aliases file, and use the** `sed` **command to format the data:**

```
# niscat mail.aliases > /tmp/aliases.tmp
# sed 's/ /: /' /tmp/aliases.tmp > /tmp/aliases
```

---

**Note –** Refer to the *Release Notes* for details on the formatting rules for the `aliases` and `passwd` files.

---

4. **Change directories to the location shown and edit the** `imldifsync.conf` **file.**

```
# cd /etc/opt/SUNWmail/dir_svc
# vi imldifsync.conf
```

5. **Change the** `mail-server`**,** `passwd-file`**, and** `aliases-file` **values and uncomment the** `mode = users` **line as shown:**

```
mail-server = "<mailserverhostname>.<fully qualified domain name>"
passwd-file = "/tmp/passwd"
aliases-file = "/tmp/aliases"
mode = users
```

In the above example, your `mail-server` can be `testserver.eng.alpha.com`., where `testserver` is the hostname of the SIMS 3.2 mail server.

6. **Copy the** `imldifsync.conf` **files to** `users.conf`**.**

```
# cp imldifsync.conf users.conf
```

7. **If you want to set a user mail store quota edit** `users.conf`**.**

   The SIMS default setting is “no limit.” To set a space limit, modify the `mailQuota` attribute as follows:

```
add-val = { “mailQuota: <quota in bytes>” , “mailFolderMap: SUN-MS” }
```

   where `<quota in bytes>` would be 10000000 if you wanted to set a mail space quota of 10 megabytes. See “Message Store Quota Enforcement” on page 148 for detailed information on setting quotas.

8. **Change directories to the location shown and convert the user data to LDIF format.**

   Use the `imldifsync` command to generate formatted user data files (`LDIF` files).

```
# /opt/SUNWmail/sbin/imldifsync -c users.conf > /tmp/users.ldif
```

   You will see the following results on the screen:

```
==================Statistics=======================

Added DNs: 2
Modified DNs: 0
Delete DNs:   0

====================================================
```

---

**Note –** By default, the mail folder will be set to the Sun Message Store.

---

The file `users.ldif` contains the following:

**CODE EXAMPLE C-1** Contents of the `users.ldif` File *(1 of 2)*

```
dn: cn="John Doe (jdoe)",ou=People,o=Alpha,c=US
changetype: add
cn: John Doe (jdoe)
cn: John Doe
sn: Doe
initials: JD
givenName: John
freeFormName: John Doe
preferredRfc822Originator: john.doe@Engineering
```

**CODE EXAMPLE C-1** Contents of the `users.ldif` File *(2 of 2)*

```
preferredRfc822Recipient: jdoe@testserver.Alpha.com
rfc822Mailbox: john.doe@testserver.Alpha.COM
rfc822Mailbox: jdoe@testserver.Alpha.com
rfc822Mailbox: jdoe@testserver.Alpha.COM
rfc822Mailbox: john.doe@Engineering.Alpha.com
mailDeliveryOption: mailbox
mailHost: testserver.Alpha.com
userPassword: {crypt}fWFuXyZ1S..Vk
uid: jdoe
homeDirectory: /export/home/jdoe
dataSource: imldifsync 1.0
objectClass: top
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: emailPerson
objectClass: person
mailQuota: -1
mailFolderMap: SUN-MS

dn: cn="Gail Evert (gevert)",ou=People,o=Alpha,c=US
changetype: add
cn: Gail Evert (gevert)
cn: Gail Evert
sn: Evert
initials: GE
givenName: Gail
freeFormName: Gail Evert
preferredRfc822Originator: gail.evert@Engineering
mailForwardingAddress: gail.evert@testserver.Alpha.COM
mailDeliveryOption: mailbox
userPassword: {crypt}fWFuXyZ1S..Vk
uid: gevert
homeDirectory: /export/home/gevert
dataSource: imldifsync 1.0
objectClass: top
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: emailPerson
objectClass: person
mailQuota: -1
mailFolderMap: SUN-MS
```

9. **Change directories to the location shown and populate the directory with the user LDIF formatted data.**

   Use the `ldapmodify` command to add the new entries to the directory:

```
# cd /opt/SUNWconn/bin
# ldapmodify -D "cn=admin,o=Alpha,c=us" -w secret -f /tmp/users.ldif
```

   You will see the following output on the screen:

```
adding new entry cn="John Doe (jdoe)",ou=People,o=Alpha,c=US
adding new entry cn="Gail Evert (gevert)",ou=People,o=Alpha,c=US
```

10. **Verify that the entries are present in the directory, using the `ldapsearch` command.**

**CODE EXAMPLE C-2** Results of the `ldapsearch` Command for User Data

```
# ldapsearch -L -b "o=Alpha,c=us" "cn=*" cn
dn: CN=John Doe (jdoe),OU=People,O=Alpha,C=US
cn: John Doe (jdoe)
cn: John Doe
dn: CN=Gail Evert (gevert),OU=People,O=Alpha,C=US
cn: Gail Evert (gevert)
cn: Gail Evert
```

# Populating the Directory with User Aliases Data and Distribution Lists — Sample Session

The following example is a continuation from the user data population example shown in “Populating the Directory with User Data—Sample Session” on page 319. It also assumes that you have extracted the user mail-aliases information from NIS+ and are now attempting to populate the directory with user aliases data for Alpha Corporation, Inc. as shown below. The user mail-alias being created is called `testsubject` and it will have two people as its members, John Doe and Gail Evert. The owner of the alias is designated as `admin`. The distribution list mail-alias is

called `testsubject-list`, and it has owner `owner-testsubject-list` and automated request alias `testsubject-list-request`. The owner is user `jdoe` and the distribution list has two members, `gevert` and `jdoe`.

```
testsubject: gevert,jdoe
owner-testsubject: admin

testsubject-list: jdoe,gevert
testsubject-list-request: jdoe
owner-testsubject-list: jdoe
```

To create directory entries for these user aliases, complete the following steps:

1. **Log in as root.**

```
$ su
Password: <Enter your root password>
#
```

---

**Note –** Since you have to populate the directory with user data before you populate it with user aliases data, and since the process of extracting user, user aliases, and distribution list data is the same, you have already completed steps Step 3 to Step 5 as part of Populating the Directory with User Data—Sample Session." This section does not repeat these steps.

---

2. **Copy the `imldifsync.conf` file to `groups.conf` to keep the user population data distinct from the user aliases population data:**

```
# cp imldifsync.conf groups.conf
```

3. **Edit the `groups.conf` file, change the `add-val` line as shown below and change the `mode` token from `users` to `groups`.**

```
add-val = { "associatedDomain: <same value as mail-domain> }
mode = groups
```

`mail-domain` value should be lower down in the same file.

4. **Change directories to the location shown and convert the user aliases list data to LDIF format.**

   Use the `imldifsync` command to generate formatted user aliases data files (`LDIF` files).

```
# /opt/SUNWmail/sbin/imldifsync -c groups.conf > /tmp/user_aliases.ldif
```

You will see the following results on the screen:

```
==================Statistics=======================

Added DNs: 2
Modified DNs: 0
Delete DNs:   0

====================================================
```

**Note –** By default, the mail folder will be set to the Sun Message Store.

The file `user_aliases.ldif` contains the following:

**CODE EXAMPLE 10-1** Contents of the `user_aliases.ldif` File for User Aliases *(1 of 2)*

```
dn: cn="testsubject",ou=Groups,o=Alpha,c=US
changetype: add
cn: testsubject
rfc822MailMember: gevert@testserver.Alpha.COM
rfc822MailMember: jdoe@testserver.Alpha.COM
mailDeliveryOption: mailbox
rfc822Owner: admin@testserver.Alpha.COM
rfc822Owner: root@testserver.Alpha.COM
ownerDeliveryOption: mailbox
dataSource: imldifsync 1.0
objectClass: top
objectClass: rfc822MailGroup
objectClass: emailGroup
objectClass: groupOfNames
associateddoman: <the value you added to the .conf file>
-
dn: cn="testsubject-list",ou=Groups,o=Alpha,c=US
changetype: add
cn: testsubject-list
```

**CODE EXAMPLE 10-1** Contents of the `user_aliases.ldif` File for User Aliases *(2 of 2)*

```
rfc822MailMember: jdoe@testserver.Alpha.COM
rfc822MailMember: gevert@testserver.Alpha.COM
mailDeliveryOption: mailbox
rfc822Owner: jdoe@testserver.Alpha.COM
rfc822Owner: root@testserver.Alpha.COM
ownerDeliveryOption: mailbox
rfc822RequestsTo: jdoe@testserver.Alpha.COM
requestsToDeliveryOption: mailbox
dataSource: imldifsync 1.0
objectClass: top
objectClass: rfc822MailGroup
objectClass: emailGroup
```

5. **Change directories to the location shown and populate the directory with the user aliases LDIF formatted data.**

   Use the `ldapmodify` command to add the new entries to the directory:

```
# cd /opt/SUNWconn/bin
# ldapmodify -D "cn=admin,o=Alpha,c=us" -w secret -f /tmp/user_aliases.ldif
```

   You will see the following results on the screen:

```
adding new entry cn="testsubject",ou=Groups,o=Alpha,c=US
adding new entry cn="testsubject-list",ou=Groups,o=Alpha,c=US
```

6. **Verify that the entries are present in the directory, using the `ldapsearch` command.**

**CODE EXAMPLE C-3** Results of the `ldapsearch` Command for User Aliases Data

```
# ldapsearch -L -b "o=Alpha,c=us" "cn=*" cn
dn: CN=testsubject,OU=Groups,O=Alpha,C=US
cn: testsubject
# ldapsearch -L -b "o=Alpha,c=us" "cn=*" cn
dn: CN=testsubject-list,OU=Groups,O=Alpha,C=US
cn: testsubject-list
```

## Migrating /var/mail Mailboxes

`imimportmbox` is a utility that migrates `/var/mail` files into the message store. You need to determine which `/var/mail` files to transfer, since these files may be in a variety of places depending on the organization of the previous system.

`imimportmbox` currently uses a general purpose `/var/mail` driver to parse the complicated /var/mail source files. If the `/var/mail` driver finds an error in the source file, it may attempt to fix the error, thus rewriting the source file correctly. `imimportmbox` may also create standard `/var/mail` "dot" files during the normal course of parsing. Therefore, depending on the system layout, it may be useful to copy the target source files into a temporary area if the effects of the varmail driver are undesirable.

If a `/var/mail` file is too damaged to be parsed correctly, `imimportmbox` may report a failure. Therefore any major migration effort should have a means to track the success of each import.

The utility `imexportmbox` may be used to copy email back out from the message store to disk in `/var/mail` format. But an export may not inaccessibly be the same byte for byte as an import, since `imimportmbox` parses `/var/mail` into the message store format and `imexportmbox` recreates a valid `/var/mail` file, which are not inaccessibly exactly the same.

For more information on `imimportmbox` or`imexportmbox`, please see the *SIMS Reference Manual*.

APPENDIX **D**

# SIMS Directory Schema and Directory Information Tree



## Introduction

This appendix is for SIMS engineering groups and customers who wish to develop their own tools for populating the directory with data residing in other databases. SIMS provides a tool, `imldifsync`, which assists in migrating users and distribution lists from NIS and NIS+ into the directory (see “Populating the Directory” on page 167), however this appendix is for administrators who wish to create customized migration tools.

This chapter describes the following:

1. SIMS directory information tree (DIT) requirements,
2. The type and format of objects and attribute values for directory entries required by SIMS.
3. How to set up the DIT, create SIMS user and distribution list entries.
4. The attributes that are required to be indexed for optimal performance of the mail server.

This document assumes that the reader is familiar with, and is comfortable with installing and managing SIMS and has read the *SIMS Reference Manual*. Readers should also be familiar with LDAP Directory Interchange Format (`ldif(1)`). It is also assumed that the reader has read and understands the following documents.

- X.521 (1993)
- Definition of the inetOrgPerson ObjectClass (`ftp://ds.internic.net/internet-drafts/draft-smith-ldap-inetorgperson-00.txt`).

## Directory Basics

The directory server stores user and distribution list data, as well as configuration data for Sun Messaging Connectivity Services channels (SMCS). Data is stored in directory entries arranged in a tree structure called a directory information tree (DIT). A directory entry consists of a set of attributes and is described by its object classes and the required and optional attributes of these object classes. All the object classes and attributes which are supported by the directory service are defined in the *directory schema*.

# The Directory Information Tree

The role of a directory service is to support the storage and retrieval of data. The entries in an LDAP directory are often visualized as being organized in a tree-like structure. This mirrors the tree model used by most file systems. This is referred to as the directory information tree. Just as a file path uniquely identifies a file within a file system, a directory entry is uniquely identified within the DIT using a distinguished name (DN). A DN identifies the entry by using a comma-separated set of attribute and attribute values. The DN's left-most value is known as the Relative Distinguished Name (RDN). Following this value are subsequent attributes that represent a branch point above the entry. The final, or right-most, attribute represents the conceptual root point of the DIT.

SIMS requires that the data be represented in a combination of *primary/secondary* tree. The primary tree is the repository of all users and distribution list data and is patterned after an OSI DIT. The secondary tree is the Domain Component tree (DC tree) and mirrors the DNS hierarchy. The DC tree provides the mapping from the DNS name space to the primary namespace where all the users and distribution lists are defined. This mapping is used by message transfer agent for building routing tables and in making message routing decisions.

The root entry of the DIT is defined by the suffix value of the directory server. Hence, the LDAP directory server will have to support multiple suffixes in order for multiple DITs to be created. The Sun Directory Server and Netscape Directory Server support multiple DITs.

## Primary tree

As mentioned before, the primary tree for SIMS is patterned after an OSI tree and is rooted at `o=<orgnization-name>,c=<country-name>`. Thus, the suffix for the primary tree has two components. However, this DIT can be created with a single component suffix in which case the primary tree is rooted at `c=<country-name>`.

In the example illustrated below, we will use a two components suffix. The nodes in **bold** are the nodes that correspond to a site's organization structure. Each node in the DIT that mirrors the organization is required to have the following organization units:

- organization unit : people
- organization unit : groups
- organization unit : services

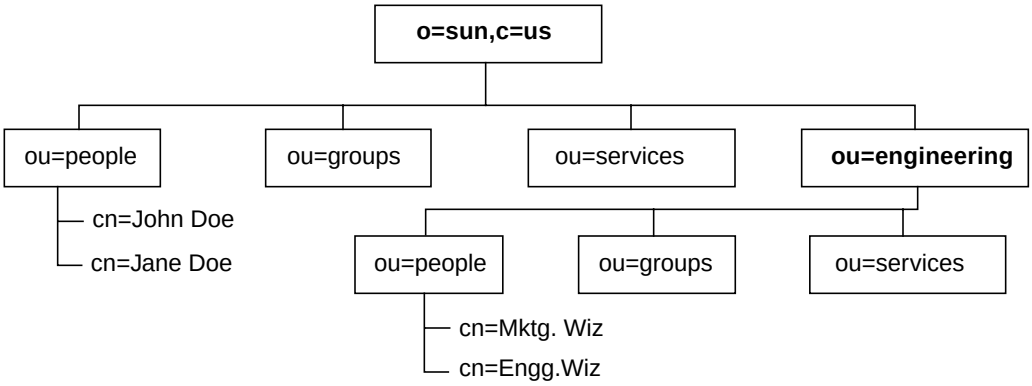



**FIGURE D-1** SIMS OSI (Primary) Directory Information Tree

User entries are defined so that they are contained within the *people* organization unit and distribution list entries are defined so that they are contained within the *groups* organization unit.

In the figure above, the DN for a user entry in engineering organizational unit will have a suffix of `ou=people,ou=engineering,o=sun,c=us`, preceded by the entries Relative Distinguished Name.

Each one of these containers are directory entries themselves and are comprised of `top` and `organizationalUnit` object class (these are defined in the section titled Miscellaneous Object Classes). The directory entry for *people* container is shown below (*groups* and *services* follow the same format):

```
dn: ou=People,o=sun,c=us
organizationalunit: people
objectclass: top
objectclass: organizationalUnit
```

In FIGURE D-1, the root of the DIT is defined by the suffix `o=sun,c=us`. This directory entry is comprised of `top`, `organization` and `domainRelatedObject` object classes. The directory entry for the root entry is shown below:

```
dn: o=sun,c=us
organization: sun
objectclass: top
objectclass: organization
objectclass: domainRelatedObject
associateddomain: sun.com
```

The value of `associatedDomain` attribute is the DNS suffix that corresponds to the node in the OSI tree. This is explained further when we talk about the Domain Component tree (secondary DIT).

## Secondary tree

This tree provides the mapping from DNS name space to the OSI name space and follows the recommendations of the *RFC2247* (*Using Domains in LDAP/X.500 Distinguished Names*). For the purposes of SIMS, domain component tree is rooted at the root DNS domain suffix for that organization. In the example of Sun Microsystems, Inc., the secondary suffix would be `dc=sun,dc=com`. As in the case of the primary tree, this tree can have a single component suffix.

In the example below, we use two components suffix. The tree is rooted at `dc=sun,dc=com` and each node is composed of `top`, `domain` and `labeledURIObject` object classes. The directory entry for people container is shown below:

```
dn: dc=eng,dc=sun,dc=com
dc: eng
objectclass: top
objectclass: domain
```

```
objectclass: labeledURIObject
labeleduri: ldap:///ou=eng,o=sun,c=US??sub
associatedname: ou=eng,o=sun,c=us
description: DNS to DN mapping for eng.sun.com
```

The value of `associatedName` is the DN in the OSI tree (also referred to as the *primary* tree) which contains the users in the DNS domain represented by this node (eng.sun.com). In our example, the DC nodes `associatedName` attribute points to the container `ou=eng,o=sun,c=us` in the OSI tree. This is required by SIMS message transfer agent. Mail routing tables are built for each sub-domain by searching for users / distribution lists in the associated OSI sub-tree.

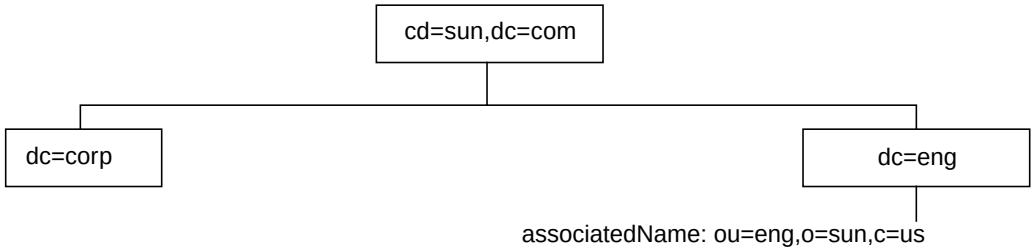



**FIGURE D-2** SIMS Domain Component (Secondary) Directory Information Tree

It is very important the associations between the domain component tree (secondary) and the OSI tree (primary) be setup correctly for desired mail routing to occur. Figure above shows that the node `dc=eng,dc=Sun,dc=COM` points to the `DN ou=eng,o=sun,c=us`. SIMS looks for users in the `eng.sun.com` domain in the sub-tree `ou=People,ou=eng,o=sun,c=us`.

# The SIMS Object Classes



This section describes the objects and attribute values used for creating SIMS directory entries. There are three classes of objects, and each object supports a number of attributes. Attributes are either *required*, *reserved*, or *optional*. Required attributes are used or the basic functionality of SIMS or used by one or more service

of SIMS when extended features of SIMS are used. Reserved attributes are reserved for future use by SIMS and should not be used. Optional attributes are not used nor planned to be used by SIMS.

A detailed description of the attributes is provided along with a parenthetical descriptive code. For example, a code may look like this:

**(cis, 1 - many, {mta, admin})**

The first part of the code (in this case **cis**) describes the attribute syntax. The second part (**1-many**) possible numbers of attribute values, and the third part (**{mta, admin}**) describes the SIMS services and functions which are affected by the attribute.

- **Attribute Syntax** is a directive to the Directory Service Agent (DSA). The possible syntaxes are:
  - **dn** - a string distinguished name (as defined in rfc1779)
  - **cis** - a case ignore string
  - **ces** - a case exact string (case is significant during comparisons)
  - **bin** - a binary value
  - **tel** - a string telephone number (blanks and dashes are ignored during comparisons)
  - **utctime** - UTC time stamp in the following format YYMMDDHHMMSS.
  - **protected** - an encrypted value. In Sun Directory Server 1.0, a value prefixed with `{crypt}` denotes that it has already been encrypted according to UNIX crypt. A value without the crypt prefix is assumed to be in the clear and is encrypted by the directory before it is stored. Attributes with *protected* syntax are not returned in searches unless the credentials that the client is using when binding to the directory has the access (see ACL for Sun Directory Server) over the attribute with a protected syntax.
- **Number of Attributes**. Attributes may appear more than once in a directory entry. The possible numbers of attribute values are:
  - **1** - one and only one value
  - **0-1** - zero or one value
  - **0-many** - zero or more values
  - **1-many** - more or more values
- **SIMS Services**. SIMS provides the following services:
  - **ms** - Sun Message Store
  - **ma** - IMAP/POP message access
  - **mta** - SMTP mail service
  - **smcs** - Connectivity services to legacy mail systems. Supported mail systems are cc:Mail, MS Mail and IBM PROFS
  - **admin** - SIMS administration service

- **SIMS Functions**. The SIMS services, individually or in combination, provide a set of functionality pertinent to a mail server. These are as follows:
  - **auth** - user authentication to mailbox and directory data.
  - **routing** - routing of messages. Includes routing to the correct mail server and to the correct channel.

Object class attributes, described in the sections below, are marked with a list of services that depend on that attribute. The format of the notation is described by the following BNF.

```
services      ::= "{" service-name [:service-name] "}"
service-name  ::= service [:service-name]
service       ::= "ms" | "ma" | "mta" | "smcs"
```

within parenthesis attribute syntax and list of services that depend on the attribute.

# User Object Classes

A SIMS e-mail user is represented by the entry in the directory. An entry which stores user information consists of attributes drawn from at a minimum the following directory object classes:

- `top` - attributes useful for describing the classifications of a directory object.
- `person` - attributes useful for describing a person
- `organizationalPerson` - attributes for describing a person belonging to an organization
- `inetOrgPerson` - same as `organizationalPerson` and also one that interacts with the internet
- `emailPerson` - attributes useful for describing an e-mail user.

X.521 (1993) defines the `top`, `person` and `organizationalPerson` object classes. The `inetOrgPerson` object class is defined in the IETF draft *Definition of the inetOrgPerson Object Class* (ftp://ds.internic.net/internet- drafts/draft-smith-ldap-inetorgperson-00.txt). The `emailPerson` and `gateway` object classes are defined by Sun Microsystems, Inc. and Wingra Technologies.

Optionally, the following object classes are used for a directory entry for those users who use legacy mail systems (like cc:Mail, MS Mail and IBM PROFS) and are connected to SIMS and to the internet mail using Sun Messaging Connectivity Services (SMCS).

- `gatewayCCMailUser` - attributes for describing a user of cc:Mail channel
- `gatewayMSMailUser` - attributes for describing a user of MS Mail channel
- `gatewayProfsUser` - attribute for describing a user of IBM PROFS channel

A SIMS user entry is extensible (as are all other directory entries) and may contain additional object classes / attributes once such schema extensions have been made in the directory. Care must be taken to ensure that the semantics of existing object classes are not changed by a schema extension.

## `emailPerson` Object Class

The `emailPerson` object class represents SIMS e-mail users. All SIMS users must have this object class. The object class is defined as follows:

```
( OID - TBD
NAME 'emailPerson'
MUST (
    commonname $ objectClass
 )
MAY (
    assistant $ channelName $ channelType $ dataSource $
    generationQualifier $ freeFormName $ homeDirectory $
    homeFacsimileTelephoneNumber $ mail $
    mailAutoReplyExpirationDate $ mailAutoReplyMode $
    mailAutoReplySubject $ mailAutoReplyText $
    mailAutoReplyTextInternal $ mailDeliveryFile $
    mailDeliveryOption $ mailFolderMap $ mailForwardingAddress
    $ mailHost $ mailMessageStore $ mailProgramDeliveryInfo $
    mailQuota $ objectStatus $ preferredRfc822Recipient $
    reportsTo $ rfc822MailBox $ userDefinedAttribute1 $
    userDefinedAttribute2 $ userDefinedAttribute3 $
    userDefinedAttribute4
    )
)
```

**TABLE D-1** Required emailPerson Attributes

| Attribute | Description |
|---|---|
| commonname | (cis, 1 - many, {mta, admin}) The user's full name. There can be more than one cn entry for a user though each entry is required to be unique. |
| objectClass | (cis, 1 - many, {mta, smcs, ma, admin}) the object classes used in defining a SIMS user entry |
| channelName | (cis, 0 - 1, {smcs, mta, admin}) The name of the users SMCS channel. The channel name is chosen by the administrator when SMCS is configured. |
| channelType | (cis, 0 - 1, {smcs, mta, admin}) one of the following values denoting the user's SMCS channel: 0 - cc:Mail; 1 - Microsoft Mail; 4 - SMTP; 8 - IBM PROFS |
| freeFormName | (cis, 0 - many, {smcs}) String to supplement a user's external e-mail address (used by SMCS channels today. May be used by MTA in the future). |

**TABLE D-1** Required emailPerson Attributes

| Attribute | Description |
|---|---|
| mail | (cis, 0-1, {mta, admin}) The user's advertised e-mail address (RFC 822 format). Also know as *preferredRfc822Originator*. |
| mailAutoReplyExpirationDate | (cis, 0-1, {mta, admin}) Disable auto-reply at this date. Date must be in UTC time format. e.g. YYMMDDHHSSZ. Please note that the date takes two digit years and is not Y2K compliant. |
| mailAutoReplySubject | (cis, 0 - I, {mta, admin}) The subject line of a auto- reply message. If it contains $SUBJECT then the token is replaced by the subject line of the incoming message. |
| mailAutoReplyText | (cis, 0 - 1 {admin, mta}) The body of the auto-reply message. If it contains the tokens $SUBJECT or $BODY then these are replaced by the subject or the body of the inbound message. Use "\n" as a line separator. |
| mailAutoReplyTextInternal | (cis, 0 - 1, {mta, admin}) The body of the auto- reply message for use within the organization. If it contains the tokens $SUBJECT or $BODY then these are replaced by the subject or the body of the inbound message. Use "\n" as a line separator. |
| mailDeliveryFile | (ces, 0 - many, {mta, admin}) Fully qualified path of a file name to which incoming messages are appended. |
| mailDeliveryOption | (cis, 1-many, {mta, admin}) One or delivery options. While inbound messages can be delivered into multiple message stores (mailbox and native) message access server can read messages only from one of them. The mail store where messages are read from are specified using the mailFolderMap attribute. The supported options are:<br>**mailbox** - deliver mail to the Sun Message Store mailbox. If *mailDeliveryOption* is set to *mailbox*, then in order for the user to read messages from the Sun Message Store using SIMS message access servers, *mailFolderMap* would have to be set to *Sun-MS*. Please refer to *mailFolderMap* below.<br>**shared** - deliver mail to a message store shared mailbox This is used for setting up shared mailbox for a distribution list. If a directory entry for a user has this attribute set to *shared*, it has no affect.<br>**native** - deliver mail to a UNIX file system mailbox. If *mailDeliveryOption* is set to *native*, then in order for the user to read messages from the native message store using SIMS message access servers, *mailFolderMap* would have to be set to *UNIX V7*. Please refer to *mailFolderMap* and *mailMessageStore* below.<br>**autoreply** - deliver mail to an auto-reply facility<br>**program** - deliver mail to a Solaris program. Must point to a valid Solaris program and the program has to be on a approved list of programs installed by the system administrator. See “To Make Delivery Programs Available to Users” on page 102.<br>**forward** - forward mail to another RFC 822 compliant address. Please see attribute *mailForwardingAddress* for related information. |

**TABLE D-1** Required emailPerson Attributes

| Attribute | Description |
|---|---|
| | **file** - append mail to a file. For this option to have any effect, *mailDeliveryFile* would have to point to a valid file over which the user id. under which SIMS is running has write privileges. Please refer to *mailDeliveryFile* in this section. |
| | **c:<custom channel name>** - route mail to a user define channel. SIMS software development kit allows users to develop channel programs. These channel programs are used to process the messages delivered to the channel queue by the message transfer agent. Please refer to the SIMS software development kit and the mta documentation for more details on the developing and using custom channel programs. Also note that the key word c followed by a : is required to precede the channel name. |
| mailFolderMap | (cis, 0 - 1, {ma, admin}) The message store for a user's mail folders.<br>**UNIX V7** - UNIX version 7 mail store. Also know as the Berkeley style `/var/mail` message store.<br>**Sun-MS** - Sun Message Store. This is the recommended message store. |
| mailForwardingAddress | (cis, 0 - many, {mta, admin}) Forward mail to the specified e-mail address (RFC 822 format). For the MTA to forward the e- mail to these addresses, *mailDeliveryOption* would have to be set to *forward* in addition to any other delivery options. For example, if a user wants to forward mail to another address, then the directory entry for the user has the first block of values for *mailForwardingAddress* and *mailDeliveryOption*. However if the user wishes to continue receiving mail on their default server and forward a copy of every message to another address then the directory entry would have the second block of values.<br>`mailDeliveryOption: forward`<br>`mailFolderMap: Sun-MS`<br>`mailForwardingAddress: abraham.lincoln@whitehouse.com`<br>`mailDeliveryOption: forward`<br>`mailDeliveryOption: mailbox`<br>`mailFolderMap: Sun-MS`<br>`mailForwardingAddress: abraham.lincoln@whitehouse.com` |
| mailHost | (cis, 0 - 1, {mta, ma, admin}) Hostname of the user's IMAP / POP mail server. This is the fully qualified official hostname of the mail server where the mail is read from. MTA will deliver the incoming mail to the message store on this mail server. |
| mailMessageStore | (ces, 0 - 1, {mta, admin}) File system location for user's INBOX. This applies only when a *mailDeliveryOption* is set to *native*. MTA will deliver incoming messages to this file. |
| mailProgramDeliveryInfo | (ces, 0 - many, {mta, admin}) Specifies one or more named commands, to use in program delivery. The valid named commands are defined by `imta-program(1m)`. Please refer to the man pages for details on how to define named commands. |

**TABLE D-1** Required emailPerson Attributes

| Attribute | Description |
|---|---|
| mailQuota | (cis, 0 - 1, {mta, ms, admin}) Maximum size (in bytes) of a user's message store. A value of minus one (-1) denotes no limit on the cumulative size of messages in users INBOX and folder collection. A value of minus 2 (-2) implies that the globally defined mail quota (`ims-default-quota` in `/etc/opt/SUNWmail/ims/ims.cnf`) is used for this user. This can be typically set in the Admin Console (see “Configuring Advanced Options” on page 152). |
| preferredRfc822Recipient | (cis, 0 - 1, {mta, admin}) The user's internal e- mail address (RFC-822 format). |
| rfc822MailBox | (cis, 0 - many, {mta, admin}) Stores all the e-mail addresses (RFC-822 format) defined for the user. It stores, at a minimum, a copy of the e-mail addresses in the `mail` and `preferredRfc822Recipient` attributes. |

**TABLE D-2** Reserved emailPerson Attribute

| Attribute | Description |
|---|---|
| `mailAutoReplyMode` | (cis, 0 - many, {}) Mode of operation for the auto-reply facility. Currently only vacation is supported, hence the only acceptable value for this field is vacation. |
| `objectStatus` | (cis, 0 - 1, {} ) Used during Legacy Mail directory synchronization to denote a deleted entry. |

**TABLE D-3** Optional emailPerson Attribute

| Attribute | Description |
|---|---|
| `homeDirectory` | (cis, 0 - 1, {admin}) File system location for user's home directory. |
| `homeFacsimileTelephoneNumber` | (tel, 0-many, {admin}) User's home fax number |
| `generationQualifier` | (cis, 0 - many, {admin}) Generation information to qualify a users name. e.g. Jr., III, etc. |
| `dataSource` | (cis, 0 - many, {}) Original data source or migration tool for data in the users entry. Free form text. |
| `assistant` | (dn, 0 - many, {}) Distinguished name of the user's assistant. |
| `reportsTo` | (cis, 0 - 1, {}) Distinguished name of the user's manager |
| `userDefinedAttribute1` | (cis, 0 - many, {}) Attribute for use by the user. |

**TABLE D-3** Optional emailPerson Attribute

| Attribute | Description |
|---|---|
| `userDefinedAttribute2` | (cis, 0 - many, {}) Attribute for use by the user. |
| `userDefinedAttribute3` | (cis, 0 - many, {}) Attribute for use by the user. |
| `userDefinedAttribute4` | (cis, 0 - many, {}) Attribute for use by the user. |

## `gatewayCCMailUser` Object Class

The `gatewayCCMailUser` object class only appears in a user's directory entry if the corresponding SMCS channel has been configured. The `gatewayCCMailUser` object class is defined as follows:

```
( OID - TBD
NAME 'gatewayCCMailUser'
MUST (
    objectClass
    )
MAY (
    cCMailAddresses $ preferredCCMailOriginator $
    preferredCCMailRecipient
    )
)
```

**TABLE D-4** gatewayCCMailUser Attributes

| Attribute | Definition |
|---|---|
| objectClass | (cis, 1- many) The object classes used in defining the user's entry. |
| cCMailAddresses | (cis, 0-many) Used to route e-mail messages through a Lotus cc:Mail channel. It stores a copy of the e-mail addresses in the *preferredCCMailOriginator* and *preferredCCMailRecipient* attributes. |
| preferredCCMailOriginator | (cis, 0 - 1) E-mail address for routing through a Lotus cc:Mail channel. |
| preferredCCMailRecipient | (cis, 0 - 1) The native Lotus cc:Mail address. |

## `gatewayMSMailUser` Object Class

The gatewayMSMailUser object class is defined as follows:

```
( OID - TBD
NAME 'gatewayMSMailUser'
MUST (
    objectClass
    )
```

```
MAY (
    mSMailAddresses $ preferredMSMailOriginator $
    preferredMSMailRecipient
    )
)
```

**TABLE D-5** gatewayMSMailUser Attributes

| Attribute | Definition |
|---|---|
| objectClass | (cis, 1- many) The object classes used in defining the user's entry. |
| mSMailAddresses | (cis, 0 - many) Used to route e-mail messages through a Microsoft Mail channel. It stores a copy of the e-mail addresses in the *preferredMSMailOriginator* and *preferredMSMailRecipient* attributes. |
| preferredMSMailOriginator | (cis, 0 - 1) E-mail address for routing through a Microsoft Mail channel. |
| preferredMSMailRecipient | (cis, 0 - 1) The native Microsoft Mail address. |

## `gatewayPROFSUser` Object Class

The gatewayMSMailUser object class is defined as follows:

```
( OID - TBD
NAME gatewayPROFSUser`
MUST (
    objectClass
    )
MAY (
    pROFSAddresses $ preferredPROFSOriginator $
    preferredPROFSRecipient
    )

)
```

**TABLE D-6** gatewayPROFSUser Attributes

| Attribute | Definition |
|---|---|
| objectClass | (cis, 1- many) The object classes used in defining the user's entry. |
| pROFSAddresses | (cis, 0 - many) Used to route e-mail messages through an IBM PROFS channel. It stores a copy of the e-mail addresses in the *preferredPROFSOriginator* and *preferredPROFSRecipient* attributes. |
| preferredPROFSOriginator | (cis, 0 - 1) E-mail address for routing through an IBM PROFS channel. |
| preferredPROFSRecipient | (cis, 0 - 1) The native IBM PROFS address. |

# Distribution List Object Classes

An e-mail distribution list is represented by an entry in the directory consisting of attributes drawn from these object classes:

- `top` - attributes useful for describing the classifications of a directory object.
- `groupOfNames` - attributes useful for describing a collection of user objects.
- `rfc822MailGroup` - attributes useful for describing an e-mail distribution list.
- `emailGroup` - attributes for further describing an e-mail distribution list, especially those needed to support Sendmail distribution lists (`aliases (4)` format).

The `top` and `groupOfNames` object classes are defined in X.521 (1993). The `rfc822MailGroup` object class is defined in the default schema of the University of Michigan LDAP implementation. The `emailGroup` object class is defined by Sun Microsystems, Inc.

A distribution lists entry is extensible and may contain attributes from additional object classes once such object classes have been defined in the directory schema.

## `emailGroup` object class

The emailGroup object class represents SIMS distribution list. All SIMS distribution lists must have this object class. The object class is defined as follows:

```
( OID - TBD
NAME 'emailGroup'
MUST (
commonname $ objectClass
)
MAY (
    authorizedDomain $ authorizedSubmitter $ dataSource $
    expandable $ mailDeliveryFile $ mailDeliveryOption $
    mailProgramDeliveryInfo $ mailHost $ ownerDeliveryFile $
    ownerDeliveryOption $ ownerProgramDeliveryInfo $
    requestsToDeliveryFile $ requestsToDeliveryOption $
    requestsToProgramDeliveryInfo $ rfc822AuthorizedSubmitter $
    rfc822Mailmember $ rfc822Owner $
    rfc822UnauthorizedSubmitter $ unauthorizedDomain $
    unauthorizedSubmitter
    )
)
```

**TABLE D-7** Required emailGroup Attributes

| Attribute | Definition |
|---|---|
| commonname | (cis, 1 - many, {mta, admin}) A distribution list's common name. The distribution list name is generated from the *commonname* attribute. Please refer to "Creating a Distribution List" on page 355 for details on how the mta generates a fully qualified rfc822 address from the commonname. |
| objectClass | (cis, 1 - many, {mta, admin}) The object class used in defining a SIMS distribution list. |
| authorizedDomain | (cis, 1 - many, {mta, admin}) Domain name from which users are authorized to post to the distribution list. Wildcard character is "*" and the value should conform to rfc822 specification. A distribution list entry with an empty *authorizedDomain* allows senders from all domains to post messages to the list, except if they are called out in the following attributes: *unauthorizedSubmitter, unauthorizedDomain, rfc822UnauthorizedSubmitter.* |
| authorizedSubmitter | (dn, 1 - many, {mta, admin}) Distinguished name of the user(s) authorized to submit messages to the distribution list. These users should be defined in the directory server. The mta determines the e-mail address of the *authorizedSubmitter* by looking up the e-mail address of the dn in the directory. |
| expandable | (cis, 0 - 1, {mta, admin}) Determines if the distribution list is expandable or not. Takes value of TRUE/FALSE. If set to TRUE, `expn <dl name>` returns the rfc822 address of the members of this distribution list. |
| mailDeliveryFile | (ces, 0 - many, {mta, admin}) Fully qualified path of a file name to which incoming messages to this distribution list are appended. |
| mailDeliveryOption | (cis, 0 - many, {mta, admin}) One or delivery options. While inbound messages can be delivered into multiple message stores (mailbox and native) message access server can read messages only from one of them. The mail store where messages are read from are specified using the *mailFolderMap* attribute. The supported options are:<br>**mailbox** - Does not apply to distribution lists.<br>**shared** - deliver mail to a shared mailbox in Sun Message Store. This is used for setting up shared mailbox for a distribution list. Shared mailbox works for members of the distribution list receiving mail on the same mail server. This implies that setting *mailDeliveryOption* to *shared* delivers messages only to those members specified in members attribute who receive mail on the same mail server (i.e. have the same mailHost). Members specified in *rfc822MailMember* (i.e. external members of this distribution list) do not receive mail messages.<br>**native** - Does not apply to distribution lists.<br>**autoreply** - Does not apply to distribution lists. |

**TABLE D-7** Required emailGroup Attributes

| Attribute | Definition |
|---|---|
| | **program** - deliver mail to a Solaris program. Must point to a valid Solaris program and the program has to be on a approved list of programs installed by the system administrator. See "To Make Delivery Programs Available to Users" on page 102.<br><br>**forward** - Does not apply to distribution lists.<br><br>**file** - append mail to a file. For this option to have any effect, *mailDeliveryFile* would have to point to a valid file over which the user id. under which SIMS is running has write privileges. Please refer to *mailDeliveryFile* in this section.<br><br>**c:<custom channel name>** - route mail to a user define channel. SIMS software development kit allows users to develop channel programs. These channel programs are used to process the messages delivered to the channel queue by the message transfer agent. Please refer to the SIMS software development kit and the mta documentation for more details on the developing and using custom channel programs. Also note that the key word *c* followed by *a :* is required to precede the channel name. |
| mailProgramDeliveryInfo | (ces, 0 - many, {mta, admin}) Specifies one or more named commands, to use in program delivery. The valid named commands are defined by `imta-program(1m)`. Please refer to the man pages for details on how to define named commands. |
| mailHost | (cis, 0 -1, {admin, mta}) Hostname of the distribution list's SMTP/MIME mail server. The host name should be fully qualified. |
| rfc822AuthorizedSubmitter | (cis, 0 - many, {admin, mta}) E-mail addresses of users authorized to post messages to the list. Distribution lists where this attribute is not set permits everybody to post messages to the list, as long as the sender is not called out in *rfc822UnauthorizedSubmitter*, *unauthorizedDomain* or *unauthorizedSubmitter* attributes. Setting this attribute has the effect of disallowing all users from sending messages to this list except those users specified in *rfc822AuthorizedSubmitter.* |
| rfc822MailMember | (cis, 0 - many, {mta, admin}) Stores the e-mail addresses (RFC-822 format) defined for the members of this list. |
| rfc822Owner | (cis, 0 - many, {mta, admin}) E-mail address of the owner(s) of the list. Must conform to RFC-822 format. |
| rfc822UnauthorizedSubmitter | (cis, 0 - many, {admin, mta}) E-mail addresses of users not permitted to post to the list. |
| unauthorizedDomain | (cis, 0 - many, {mta, admin}) Domain name for which users are not permitted to post to the list. |
| unauthorizedSubmitter | (dn, 0 - many, {admin, mta}) Distinguished name of users not permitted to post messages to the list. These users have to be defined in the directory. |

**TABLE D-8** Reserved emailGroup Attributes

| Attribute | Definition |
|---|---|
| ownerDeliveryFile | (ces, 0 - many, {mta, admin}) Mail addressed to `owner- <listname>` is sent to this file. This is a valid file over which the user id. under which SIMS is running has write privileges. |
| ownerDeliveryOption | (cis, 0 - many, {admin mta}) One or more options for mail addressed to `owner-<listname>`. |
| ownerProgramDeliveryInfo | (ces, 0 - many, {admin, mta}) Mail addressed to `owner-<listname>` are delivered to this name command. The valid named commands are defined by `imta-program(1m)`. Please refer to the man pages for details on how to define named commands. |
| requestsToDeliveryFile | (ces, 0 - many, {mta, admin}) Mail addressed to `<listname>-request` is sent to this file. This is a valid file over which the user identification under which SIMS is running has write privileges. |
| requestsTorDeliveryOption | (cis, 0 - many, {admin mta}) One or more options for mail addressed to `<listname>-request`. |
| requestsToProgramDeliveryInfo | (ces, 0 - many, {admin, mta}) Mail addressed to `<listname>-request` are delivered to this named command. The valid named commands are defined by `imta-program(1m)`. Please refer to the man pages for details on how to define named commands. |

## `rfc822MailGroup` Object Class

The `rfc822MailGroup` object class is described below for reference. This object class is defined by in the default schema of University of Michigan LDAP implementation. The object class is defined as follows:

```
( OID - TBD
NAME 'rfc822MailGroup'
MUST (
    commonname $ objectClass
)
MAY (
    associatedDomain $ autoMgt $ description $
    destinationIndicator $ errorsTo $ facsimileTelephoneNumber $
    internationaliSDNNumber $ joinable $ krbName $ labeledURI $
    mail $ member $ memberOfGroup $ moderator $
    multiLineDescription $ notice $ owner $
    physicalDeliveryOfficeName $ postOfficeBox $ postalAddress $
    postalCode $ preferredDeliveryMethod $ registeredAddress $
    requestsTo $ rfc822ErrorsTo $ rfc822RequestsTo $ seeAlso $
    streetAddress $ suppressNoEmailError $ telephoneNumber $
```

```
    teletexTerminalIdentifier $ telexNumber $ userPassword $
    x121Address $ xacl
    )
)
```

**TABLE D-9** Required rfc822MailGroup Attributes

| Attribute | Definition |
|---|---|
| commonname | (cis, 1 - many, {mta, admin}) a distribution lists common name. The distribution list name is generated from the commonname attribute. Please refer to “Creating a Distribution List” on page 355 for details on how the mta generates a fully qualified rfc822 address from the *commonname*. |
| objectClass | (cis, 1 - many, {mta, admin}) The object class used in defining a SIMS distribution list. |
| associatedDomain | (cis, 1 - many, {mta, admin}) Domain name associated with the list. |
| errorsTo | (dn, 0 - many, {mta, admin}) Distinguished name of an entry to which errors are sent. |
| member | (dn, 0 - many, {mta, admin}) Distinguished names of members of this list. These users have to be defined in the directory for them to receive e- mail messages sent to the list. |
| moderator | (dn, 0 - many, {mta, admin}) Distinguished name of the moderator of the list. If a distribution list entry has a valid *moderator* value, all messages submitted to the list are forwarded to the moderator. The moderator then re-submits messages to the list for them to be delivered to the list members. |
| requestsTo | (dn, 0 - many, {mta, admin}) Distinguished name to which requests to be added to this list, are sent. Request alias generated by the MTA is of the format `<listname>-request@<domain>`, where domain is the value of the *associatedDomain* attribute. |
| rfc822ErrorsTo | (cis, 0 - many, {mta, admin}) E-mail address (RFC 822 format) to which errors are sent. |
| rfc822RequestsTo | (cis, 0 - many, {mta, admin}) E-mail address (RFC 822 format) to which request to be added to the list are sent. Request alias generated by the MTA is of the format `<listname>-request@<domain>`, where `domain` is the value of the *associatedDomain* attribute. |

**TABLE D-10** Reserved rfc822MailGroup Attributes

| Attribute | Definition |
|---|---|
| joinable | (cis, 0 -1, {mta, admin}) Whether users may add themselves to a list or not. Legal values are TRUE or FALSE. |
| mail | (cis, 0 - many, {mta, admin}) |
| memberOfGroup | (dn, 0 - many, {mta, admin}) Distinguished name of the lists that have this list as a member. |

**TABLE D-10** Reserved rfc822MailGroup Attributes

| Attribute | Definition |
|---|---|
| owner | (dn, 1 - many, {admin, mta}) Distinguished name of a person responsible for maintaining the distribution list. |
| suppressNoEmailError | (cis, 0 - 1, {mta, admin}) Prevent delivery of “no e- mail” errors. Takes TRUE or FALSE as values. |
| userPassword | (protected, 0 - 1, {admin, mta, ma}) List password. |

**TABLE D-11** Optional rfc822MailGroup Attributes

| Attribute | Definition |
|---|---|
| description | (cis, 0 - many, {admin}) Description of the list. |
| destinationIndicator | (cis, 0 - many, {admin}) Country and city address information |
| facsimileTelephoneNumber | (tel, 0 - many, {admin}) Fax telephone number of the distribution list. |
| internationaliSDNNumber | (tel, 0 - many, {admin}) International ISDN number of the distribution list. |
| krbName | (cis, 0 - many, {}) Kerberos name for the list. |
| labeledURI | (cis, 0 - many, {}) Uniform resource identifier for the list. |
| multiLineDescription | (cis, 0 - many, {}) Multi line description of the list. |
| notice | (cis, 0 - many, {}) List notice. |
| physicalDeliveryOfficeName | (cis, 0 - many, {}) Mail stop |
| postOfficeBox | (cis, 0 - many, {}) Post office box. |
| postalAddress | (cis, 0 - many, {}) Postal address |
| postalCode | (cis, 0 - many, {}) Postal code |
| preferredDeliveryMethod | (cis, 0 - 1, {}) Preferred delivery method of communication. |
| registeredAddress | (cis, 0 - many, {}) Registered postal address |
| seeAlso | (dn, 0 - many, {}) Distinguished name of an entry to consult for further information. |
| telephoneNumber | (tel, 0 - many, {}) Telephone number in international format. |
| teletexTerminalIdentifier | (cis, 0 - many, {}) Teletex terminal ID and optional parameters for a teletex terminal. $ separated string. |
| telexNumber | (cis, 0 - many, {}) Telex number, country code and answer back code for a teletex terminal. |
| x121Address | (cis, 0 - many, {}) An address as defined by the ITU recommendation X.121. |
| xacl | (cis, 0 - many, {}) Extended access control list. |

## Miscellaneous Object Classes

The following object classes are used to create directory entries that create the DIT. The are:

- country
- organization
- domainRelatedObject
- organizationalUnit
- domain
- labeledURIObject

### `country` Object Class

The country object class may be used to create the root suffix entry of the primary DIT. If the primary tree has a single component suffix, that suffix is comprised of the `top` and `country` object classes. The object class is defined as follows:

```
( OID - TBD
NAME 'country'
MUST (
    countryName $ objectClass
)
MAY (
    description $ searchGuide
    )
)
```

### `organization` Object Class

The `organization` object class may be used to create the root suffix entry of the primary DIT. If the primary suffix has two components, that suffix is comprised of the `top`, `organization` and `domainRelatedObject` object classes. If the primary tree has single component suffix (e.g. `c=us`) then the second level nodes in the tree are created from `top`, `organization` and `domainRelatedObject` object classes. The object class is defined as follows:

```
( OID - TBD
NAME 'organization'
MUST (
    organizationName $ objectClass
)
MAY (
    businessCategory $ description $ destinationIndicator $
    facsimileTelephoneNumber $ internationaliSDNNumber $
```

```
    locality $ physicalDeliveryOfficeName $ postOfficeBox $
    postalAddress $ postalCode $ preferredDeliveryMethod $
    registeredAddress $ searchGuide $ seeAlso $ state $
    streetAddress $ telephoneNumber $
    teletexTerminalIdentifier $ telexNumber $ userPassword $
    x121Address
    )
)
```

## `domainRelatedObject` Object Class

The `domainRelatedObject` object class may be used to create the root suffix entry of the primary DIT. If the primary suffix has two components, that suffix is comprised of the `top`, `organization` and `domainRelatedObject` object classes. If the primary tree has single component suffix (e.g. `c=us`) then the second level nodes in the tree are created from `top`, `organization` and `domainRelatedObject` object classes. The object class is defined as follows:

```
( OID - TBD
NAME 'domainRelatedObject'
MUST (
associatedDomain $ objectClass
)
MAY (
    )
)
```

## `organizationalUnit` Object Class

The `organizationalUnit` object class is used to create the container entries of the primary DIT. These entries are the organizational unit containers corresponding to an OSI tree based on geography (east, west, UK, Russia, etc), functional units (engineering, marketing, etc). The `OU` entry is created by using `top` and `organizationalUnit` object classes. Each one of these organization units is required to have three more `OU` entries `people`, `groups`, and `services`. The object class is defined as follows:

```
( OID - TBD
NAME 'organizationalUnit'
MUST (
    objectClass $ organizationalUnitName
    )
MAY (
    businessCategory $ description $ destinationIndicator
    $ facsimileTelephoneNumber $ internationaliSDNNumber $
    locality $ physicalDeliveryOfficeName $ postOfficeBox $
```

```
    postalAddress $ postalCode $ preferredDeliveryMethod $
    registeredAddress $ searchGuide $ seeAlso $ state $
    streetAddress $ telephoneNumber $    teletexTerminalIdentifier $
    telexNumber $ userPassword $     x121Address
    )
)
```

## `domain` Object Class

The `domain` object class is used to create the container entries of the secondary DIT. These entries are the domain component containers corresponding to a DNS suffix. The `DC` entry is created by using `top`, `domain` and `labeledURIObject` object classes. The object class is defined as follows:

```
( OID - TBD
NAME 'domain'
MUST (
    dc $ objectClass
    )
MAY (
    associatedName $ businessCategory $ description $
    destinationIndicator $ facsimileTelephoneNumber $
    internationaliSDNNumber $ locality $ organizationName $
    physicalDeliveryOfficeName $ postOfficeBox $ postalAddress $
    postalCode $ preferredDeliveryMethod $ registeredAddress $
    searchGuide $ seeAlso $ state $ streetAddress $
    telephoneNumber $ teletexTerminalIdentifier $ telexNumber $
    userPassword $ x121Address
    )
)
```

## `labeledURIObject` Object Class

The `labeledURIObject` object class is used to create the container entries of the secondary DIT. These entries are the domain component containers corresponding to a DNS suffix. The `DC` entry is created by using `top`, `domain` and `labeledURIObject` object classes. The object class is defined as follows:

```
( OID - TBD
NAME 'labeledURIObject'
MUST (
    objectClass
    )
MAY (
```

```
    labeledURI
    )
)
```

# Client Data Objects

These LDAP objects and attributes were created to support specific applications.

## Calendar Data

Calendar data consists of the LDAP objects and attributes that will be generated to support the Web Access calendaring object and the license support for the calendar server. An entry which stores calendaring information consists of attributes drawn from the following directory object classes:

```
objectclass IMCalendarUser
MUST
    objectclass $ userID $ userPassword $ IMcalendarHost
MAY
    IMcalendarName
```

`userID` - defined in RFC 1274, *The COSINE and Internet X.500 Schema*, as part of object class `pilotPerson`.

`userPassword` - defined in RFC 1274, *The COSINE and Internet X.500 Schema*, as part of object class `simpleSecurityObject`.

`IMcalendarHost` and `IMcalendarName` are defined for the first time in this document. At this time, calendar data has only one object class, `IMCalendarUser`. It has the following attributes.

**TABLE D-12** `IMCalendarUser` Required Attributes

| Attribute | Definition |
|---|---|
| IMcalendarHost | (cis, 1, {}) The fully qualified calendar server hostname. |
| objectClass | (ces, 1-many, {}) The object classes used in defining the calendar entry. |
| userID | (cis, 1, {}) (Mandatory Attribute) User's identification shared with SIMS/IMAP login. |
| userPassword | (cis, 1, {}) (Mandatory Attribute) The calendar user's password shared with SIMS/IMAP login. |
| IMcalendarName | (cis, 1, {}) Name of calendar. |

# Creating a Directory Information Tree, Users and Distribution Lists

In this section, we will show how to use the previously described object classes to create the directory information tree and populate the directory with user and distribution list entries.

## Setting up the DIT

SIMS requires two directory information trees. The domain component tree, also referred to as the DC tree, provides the mapping from DNS name space to the primary tree. The primary tree, also referred to as the OSI tree, is the repository of all user, distribution list, and SMCS channel entries. The mapping that is provided by the DC tree is essential for SIMS message transfer agent.

We will first show how to setup the OSI tree followed by examples on setting up the DC tree, including the DC to OSI mapping. For the purposes of this example, we will assume that the mail server hostname is `mail.widget.com`, the organizations DNS suffix is `widget.com` and the organization name is Widget, Inc. As Widget, Inc. is small company, no organizational units are defined.

In our example, the root node's suffix has two components and is created by an entry that is defined by `top`, `organization` and `domainRelatedObject` object classes. The following directory entry will create the root node for the OSI tree.

```
dn: o=Widget,c=us
organization: Widget, Inc
objectclass: top
objectclass: organization
objectclass: domainRelatedObject
associateddomain: widget.com
```

SIMS requires that each branch in the OSI tree contain three organization units (OUs). These are `people`, `groups`, and `services`. SIMS will search for users in `OU=people` container and for distribution lists in `OU=groups` container. These nodes (or containers) are created by an entry that is defined by `top` and `organizationalUnit` object classes. The following directory entries create the three organizational units.

```
dn: ou=People,o=Widget,c=us
organizationalunit: people
objectclass: top
objectclass: organizationalUnit
```

```
dn: ou=Groups,o=Widget,c=us
organizationalunit: groups
objectclass: top
objectclass: organizationalUnit

dn: ou=Services,o=Widget,c=us
organizationalunit: services
objectclass: top
objectclass: organizationalUnit
```

This sets up the OSI tree rooted at `o=widget,c=us` with three organizational units under the root. Now we will show how to create the DC tree and setup the mapping from the DC to the OSI tree.

As is the case for OSI tree, the DC trees root suffix contains two components and is created by an entry that is defined by top, domain, and labeledURIObject object classes. The following entry will create the root node for the DC tree.

```
dn: dc=widget,dc=com
dc: widget
objectclass: top
objectclass: domain
objectclass: labeledURIObject
labelеduri: ldap:/// o=widget,c=US??sub
associatedname: o=widget,c=us
description: DNS to DN mapping for widget.com
```

Since Widget, Inc. does not have any DNS sub-domains, our example DC tree contains only one node. Please note that `associatedName` attribute has a value which is the DN for the branch in the OSI tree containing the people, group and services organization units. There is a strict one-to-one mapping between the DC tree and the corresponding branch in the OSI tree.

Extending our example, Widget, Inc. has grown and now has two divisions engineering and marketing. To represent them in the OSI tree, one would create `OU=engineering` and `OU=marketing` (these entries are exactly the same as the `OU=people` entry above with the only difference being the RDN—RDN would be marketing and engineering respectively). For each one of these branches, we will have to create three more organizational units one each for people, groups and services.

Because of the growth, Widget, Inc. also has two new DNS sub-domains `mktg.widget.com` and `eng.widget.com`. The following entries and `associatedName` mappings are created to extend and map the DC tree to the new organizational units in the OSI tree.

```
dn: dc=mktg,dc=widget,dc=com
dc: mktg
objectclass: top
objectclass: domain
objectclass: labeledURIObject
labeleduri: ldap:///ou=mktg,o=widget,c=US??sub
associatedname: ou=mktg,o=widget,c=us
description: DNS to DN mapping for mktg.widget.com

dn: dc=eng,dc=widget,dc=com
dc: eng
objectclass: top
objectclass: domain
objectclass: labeledURIObject
labeleduri: ldap:///ou=eng,o=widget,c=US??sub
associatedname: ou=eng,o=widget,c=us
description: DNS to DN mapping for eng.widget.com
```

---

**Note –** The SIMS install program creates a file containing LDIF representing the structure of the DIT. This file can be found at `/etc/opt/SUNWmail/slapd.ldif`. The contents of this file are added to the directory at install time using `ldapadd`.

---

## Creating a User Entry

To provision a user in SIMS, a user has to have, at the minimum, the following set of attributes/values. The example below assumes that the user is being added in the `widget.com` DNS domain, he receives and reads mail from Sun Message Store, and his mail server is `mail.widget.com`. There are no SMCS channels installed and this users reads and sends e-mail from SIMS. The data is in Lightweight Directory Interchange Format (LDIF).

In addition to the required attributes from `emailPerson` object class, a SIMS user definition uses two other attributes from the `inetOrgPerson` object class:

- **uid** (ces, 0 - 1, {mta, ms}) Unique identifier for a SIMS user. This is the login identifier of a SIMS user. It is required to be unique in the sub-tree of the directory which message access server searches while performing user authentication.
- **userPassword** (protected, 0 - 1, {ms}) Encrypted string representing the users password. In Sun Directory Server, the supported encryption scheme used is crypt.

```
dn: cn=John Doe(jdoe),ou=People,o=Widget,c=US
objectClass: top
objectClass: person
objectClass: organizationalPerson
```

```
objectClass: inetOrgPerson
objectClass: emailPerson
commonname:  John Doe(jdoe),
commonname: John Doe
surname: Doe
givenname: John
uid: jdoe
userPassword: {crypt}7h8g467x907gh
mailHost: mail.widget.com
preferredrfc822recipient: jdoe@mail.widget.com
rfc822mailbox: john.doe@widget.com
rfc822mailbox: jdoe@widget.com
mailquota: -1
mail: john.doe@widget.com
mailfoldermap: Sun-MS
maildeliveryoption: mailbox
```

## Creating a Distribution List

To provision a distribution list in SIMS, a distribution list entry has to have the following minimum set of attributes/values. The example below assumes that the list is being added in the `eng.widget.com` DNS domain, mail server is `mail.eng.widget.com`., and the list has one member. The data is in Lightweight Directory Interchange Format (LDIF).

```
dn: cn=all,ou=People,ou=eng,o=Widget,c=US
objectClass: top
objectClass: groupOfNames
objectClass: rfc822MailGroup
objectClass: emailGroup
commonname: all
member: cn=John Doe(jdoe),ou=People,ou=eng,o=Widget,c=US
mailhost: mail.eng.widget.com
associatedomain: eng.widget.com
```

If an external user has to be added to the distribution list, the external member addresses are specified in `rfc822MailMember` attribute. Other required attributes can be used to modify the behavior of the distribution list. For example one can setup a moderated list by adding one or more `moderator`s. Or one can disallow messages from DNS domain `singnet.sg` to be sent to the members of the list by adding `unauthorizedDomain: singnet.sg`.

# Indexed Attributes

Directory attributes must be indexed to optimize the access speed of directory information. Attributes that a directory client (message transfer agent, admin console, mail clients, and message access server) retrieves from the directory server must be indexed. However, the more indexes you support on your server, the more of a performance hit the server will take whenever entries are added or changed. Thus, it is important that only attributes used for searching by mail server be indexed. Attributes can be indexed using any of the following matching rules. The keywords inside parenthesis correspond to Sun Directory Server 1.0

- Equality (eq) - Optimizes direct access to entries where an exact attribute value is supplied
- Presence (pres) - Optimizes searches with filters specifying the presence of an attribute but no specific value (cn=*, for example)
- Substring (sub) - Optimizes searches with filters containing a partially-specified attribute value (cn=ada*, for example).
- Approximate (approx) - Optimizes searches with approximate match filters. The method used in approximate indexing is to discard vowels.

Below is a list of attribute names recommended for the optimal performance of SIMS. These are specified using the following BNF:

```
idx-list ::= attr-list:idx-syntax-list:dep-service-list
attr-list ::= attr [,attr-list]
attr ::= <name of attribute in schema>
idx-syntax-list ::= idx-syntax [, idx-syntax-list]
idx-syntax ::= eq | pres | sub |
approx dep-service-list ::= dep-service [, dep-service-list]
dep-service ::= mta | ma | ms | admin | smcs
```

Below, the first set of comma separated names is the attribute list. The second set of names (after the colon), is the index syntax rules. The third set of names, are the dependent services.

< commonname, surname, givenname, mail, mailHost : pres, eq, sub, approx : admin, mta, ma >

< preferredRfc822Recipient, rfc822Mailbox, ccmailaddresses, profsaddresses, msmailaddresses : pres, eq : smcs, mta, admin >

< userid : pres, eq : ma, mta, admin >

APPENDIX **E**

# Error Messages



## User Management Error Messages

`Fails to add/modify entry`

The connection to the directory server might be down. The Admin Server might need to be restarted.

`Cannot add the following entry ...`

The connection to the directory server might be down. The Admin Server might need to be restarted.

`Cannot access Content Manager`

Admin server might be down. Restart Admin Server.

`Failed to delete entry:`

The connection to the directory server might be down. The Admin Server might need to be restarted.

`... does not exists in the directory`

Directory structure might be inconsistent. Check the DN to make sure the node exists in the directory.

`Contains invalid input value.`

Check the indicated field to make sure it contains the valid information.

`Search failed`

The connection to the directory server might be down. The Admin Server might need to be restarted.

`Cannot find Administrative Server`

Admin server might be down. Restart Admin Server.

`Cannot find main Admin Console`

Admin Console is not connected. Try go back to the main login page and go through the login process.

`Create group failed`

The connection to the directory server might be down. The Admin Server might need to be restarted.

`Cannot find owner DN`

Inconsistent directory information. Manual inspection of the directory is required.

`Cannot access session`

Admin Console is not connected. Try go back to the main login page and go through the login process.

`Create group failed due to RMI error`

Transport error. Try restart the HotJava browser.

`Cannot find moderator DN`

Inconsistent directory information. Manual inspection of the directory is required.

`Parent node does not exist: ...`

Inconsistent directory information. Manual inspection of the directory is required.

`Cannot authenticate to LDAP server: ...`

The directory server might be down. Restart the directory server

`Failed to initialize LDAP library: ...`

The LDAP client library is not found in the library search path. Check installation components.

`Cannot connect to LDAP server: ...`

The directory server might be down. Restart the directory server

`Cannot delete entry because SMCS is using this context`

The delete operation is stopped because the entry is under SMCS control. Please verify the operation.

`Cannot delete root dn`

User tried to delete the root folder from the Admin Console. Attempt to do this will result in attempt to remove “all” of the entries in the directory.

# Log Manager Error Messages

`You need to search logs before saving them.`

You are attempting to save the current display before completing the search of log entries. Perform a search of desired log entries by selecting the desired search criteria then click the Search button. Log entries that match your specified criteria will display. Click on the Log Manager menu and select Save the current display.

# IMTA Error Messages

These messages may occur while using the IMTA and IMTA channel property books.

`IMTA is already running.`

You are attempting to start the IMTA when it is already running. If desired, you can restart the IMTA from the IMTA property book by clicking on the IMTA menu and selecting Restart IMTA, or Stop IMTA then Start IMTA.

`[Add channel] Channel name already exists.`

You are attempting to create a channel with a name that already exists. Specify a unique name for the channel you are attempting to create.

`Please select a channel`

From the Selected menu in the IMTA property book, you chose an option other than Monitor Queue but did not select a channel from the Channels section. Select a channel from the Channels section by clicking on it. Click on the Selected menu and choose the desired option.

`This channel is not configurable.`

You are attempting to configure an internal channel (reprocessing, conversion, and defragmentation channels). Internal channels cannot be configured using the Admin Console.

`Invalid entry in field: Pattern: <> Please change the value before continuing.`

In the Rewrite Rules section of the IMTA channels property book, you have entered an invalid entry for the pattern or a blank pattern and clicked the Add button. Refer to “To Add, Delete, or Modify A Rewrite Rule” on page 134 for more information on the correct syntax for entering a pattern. Try to add the rewrite rule again.

`Invalid entry in field: Template: <ddsdsd hhh> Please change the value before continuing.`

In the Rewrite Rules section of the IMTA channels property book, you have entered an invalid entry for the template and clicked the Add button. Refer to “To Add, Delete, or Modify A Rewrite Rule” on page 134 for more information on the correct syntax for entering a template. Try to add the rewrite rule again.

## Queue Monitor Error Messages

This section contains error messages that you may receive while using the Queue Monitor, an explanation of the problem, and instructions on how to resolve the problem, if applicable.

`qmonitorSvr.SelectMtaChannel(qChannel) of QMonitorPanel(): Channel created but yet to be configured.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`getQueMonitorRemoteObj() in QMonitorPanel: QUEUE MONITOR: Could not communicate with the server due to a network problem.`

The IMTA may be down. Check the System Status section on the Admin Console home page to determine the status of the IMTA. If the IMTA is down, start it from the IMTA property book by clicking on the IMTA menu and selecting Start IMTA. If taking this action does not resolve this problem, contact your authorized service provider.

`channelList.addElement in QMonitorPanel: Could not add channels to channel list.`

There may be a network problem or the server may be down. Check your admin server, if it is down, restart the server, and then restart console.

`Init() in QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`Error getting imageURL of QMonitorPanel:LoadImageURLException`

Any or all pieces of the Uniform Resource Locator (URL) are not in the proper format. Contact your authorized service provider.

`setChoiceBar() of QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`scaleCounters() of QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`updateCounterDisplay() of QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`constructMsgCount() of QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`constructMsgVolume() of QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`constructDualGauge() of QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`Notation() of QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Close extra windows or applications on your desktop and re-try again.

`getStoredMessages() of handleevent() in QMonitorPanel: QUEUE MONITOR: Could not communicate with the server due to a network problem.`

The IMTA may be down or the Admin server may be down. If the IMTA is down, start it from the IMTA property book by clicking on the IMTA menu and selecting Start IMTA. If the Admin Server is down, restart the server, and then restart console.

`Channel <channel name: No Messages>`

There are no stored messages in the selected channel. Select a channel which has stored messages and try again.

`ResetCounters() of handleevent() in QMonitorPanel: QUEUE MONITOR: Could not communicate with the server due to a network problem.`

The IMTA may be down. Check the System Status section.

`ResetCounters() of handleevent() in QMonitorPanel: Could not allocate memory for resource creation.`

The object classes involved may not yet be assigned memory. Contact your...

`Error in run() of QmonitorPanel Could not allocate memory for resource creation.`

The thread that should be running may have been interrupted by another thread. Contact your authorized service provider.

`Server Polling in run() of QMonitorPanel QUEUE MONITOR: Could not communicate with the server due to a network problem.`

The IMTA may be down. Check the System Status section...

# Message Access Protocols Error Messages

`Failed to start IMAP4/POP3.`

The IMAP4/POP3 server cannot start. Contact your authorized service provider.

`Failed to stop IMAP4/POP3.`

The IMAP4/POP3 server cannot stop. Contact your authorized service provider.

`IMAP4/POP3 are already started.`

You are attempting to start the IMAP4/POP3 server when it is already up.

`IMAP4/POP3 are already stopped.`

You are attempting to stop the IMAP4/POP3 server when it has already down.

`You should start IMAP4/POP3 in advance.`

You requested connection information when the IMAP4/POP3 server is not running. Start the IMAP4/POP3 server from the Internet Message Access Protocols property book by clicking on the Internet Message Access Protocols menu and selecting start message access protocols IMAP4/POP3.

# Directory Service Error Messages



“Directory Service Error Messages Returned by slapd and slurpd Daemons” lists the errors returned by the `slapd` and `slurpd` and all the directory service tools.

“Directory Service Error Messages Returned by the Admin Console” lists the errors returned by the Admin Console when configuring the directory service.

Some of the error messages that are returned by the directory service are self-explanatory; these errors are not included in this section.

Some error messages indicate internal errors in the directory service software. For these errors, the explanation recommends that you contact your authorized service provider. To enable your authorized service provider to diagnose the problem, make sure that you have the following information:

- The configuration file in use when the error was reported
- The log files
- Copies of any entries mentioned in the error message

## Directory Service Error Messages Returned by slapd and slurpd Daemons

`<= dn2id could not open dn2id`*fileext*

The specified file cannot be opened. Check that the root directory for the data store is defined correctly. Check that the directory exists and that you have access to it.

`<= dn2id NOID`

The `dn2id` file is invalid. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`<= dn2id_delete could not open dn2id`*fileext*

The specified file cannot be opened. Check that the root directory for the data store is defined correctly. Check that the directory exists and that you have access to it.

`<= has_children -1 could not open id2children`*fileext*

The specified file cannot be opened. Check that the root directory for the data store is defined correctly. Check that the directory exists and that you have access to it.

`<= id2children_add -1 could not open id2children`*fileext*

The specified file cannot be opened. Check that the root directory for the data store is defined correctly. Check that the directory exists and that you have access to it.

`<= id2entry (`*info*`) not found`

Database error. Use `idxgen` to regenerate the index files, as described in *SIMS Reference Manual*. Check whether any data has been lost.

`<= index_read NULL (could not open` *file*`)`
`<= index_add_values -1 (could not open/create` *file*`)`

Database error. Use `idxgen` to regenerate the index files, as described in *SIMS Reference Manual*. Check whether any data has been lost.

`<= ldbm_cache_open NULL` *location* `errno` *number* `reason` *reason*

The internal database cannot be opened. Use the information in *reason* to diagnose the problem.

`accept() failed errno` *number* `(`*error*`)`

Cannot accept connection. Stop and restart `slapd`, as described in "Maintaining the Directory Service" on page 241.

`Alias dereferencing problem`

An alias encountered during the operation could not be dereferenced. Check that the aliased entry exists. Check that you have access to the aliased entry. If the alias is in a replica naming context, synchronize the replica with the master naming context.

`Alias loop detected: (Entry:` *dn*`)`

A loop was detected while dereferencing an alias. Check that the alias entry is correctly defined.

`Alias problem`

The distinguished name in the aliasedObjectName is not valid, or the entry indicated does not exist.

`Already exists`

The attribute and value you are trying to add is already present in the entry.

`another slurpd is already running`

You have tried to start a second `slurpd` daemon when one is already running. The second daemon will not start.

`artc: msgid not in request list`

Internal error. Contact your authorized service provider.

`Bad Alias Entry (`*alias*`):no aliasedObjectName`

The alias entry does not contain a value for the aliasedObjectName attribute.

`Bad parameter to an ldap routine`

Internal error. Contact your authorized service provider.

`ber_alloc failed`

Insufficient memory available.

`ber_flush failed errno` *number* `msg` (*error*)

Cannot send LDAP PDU.

`ber_get_next on fd` *number* `failed errno` *number* (*error*)

Cannot receive PDU from the network. Check for errors in your networking software.

`ber_printf failed`

Error encoding result.

`ber_scanf failed`

An invalid LDAP PDU was received.

`cache_add_entry_lock failed`

The entry you are trying to add already exists in the database. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`Calloc error in reception`

Insufficient memory available

`Calloc error of appl table in reception`

Insufficient memory available.

`calloc of` *number* `elems of` *number* `bytes failed`

Insufficient memory available.

`Cannot initialize queue`

Insufficient memory available.

`Cannot initialize status data`

Cannot open `/usr/tmp/slurpd.status`. Check the file permissions.

`Cannot modify object class`

You cannot modify the objectClass attribute of an entry. Delete and re-create the entry, ensuring that you provide all the mandatory attribute of the new objectClass.

`Can't contact LDAP server`

The LDAP server cannot be reached. Retry the operation. Check for other errors indicating a shortage of resources required by the directory server.

`Can't find aliased entry (`*dn*`)`

The aliased entry does not exist or you do not have access to it. Check that the alias entry is defined correctly. If the alias entry is in a replica naming context, synchronize the replica with the master naming context.

`Can't find aliased entry (`*dn*`) for entry (`*aliasdn*`)`

Check that the aliased entry exists. Check that you have access to the aliased entry. If the alias is in a replica naming context, synchronize the replica with the master naming context.

`Can't getmsg from slapd errno=`*number*

Internal error in SNMP daemon. Contact your authorized service provider.

`Can't initialize connection to the license server, exiting`

The license server holding the directory server license cannot be reached. The directory server will not run without a license.

`Can't putmsg to slapd errno=`*number*

Internal error in SNMP daemon. Contact your authorized service provider.

`can't set file descriptor limit to` *number*

The maximum number concurrent of connections, specified with the `-n` option of `slapd`, is too large.

`check_naming failed`

An invalid distinguished name was supplied.

`Constraint violation`

An attribute value exceeds the maximum size permitted. Check for other error messages indicating the attribute for which the value is incorrect.

`could not delete` *number* `(`*string*`) from cache`

Internal error. Contact your authorized service provider.

`could not find attribute` *attribute*

Client error. A delete attribute operation is requested for an attribute that is not present.

`could not find value for attr` *attribute*

Client error. A delete attribute value operation is requested for an attribute value that is not present.

`Could not fork to run` *process*

Error running `idxgen`. Child processes could not be created. Reduce the number of processes running on your machine and rerun `idxgen`.

`could not open config file` *file* `- absolute path?`

Check that the configuration file exists in the specified location and is readable. Check that you have specified an absolute path to the file.

`Could not open id2entry`*fileext*

The specified file cannot be opened. Check that the root directory for the data store is defined correctly. Check that the directory exists and that you have access to it.

`could not open slapd.replog`

The replication log file cannot be opened. Check that the file exists and has suitable file permissions.

`Could not open/create dn2id`*fileext*

The specified file cannot be created or opened. Check that the root directory for the data store is defined correctly. Check that the directory exists and that you have access to it.

`Could not open/create id2entry` *file*

The `id2entry` file cannot be created or opened. Check that the root directory for the data store is defined correctly. Check that the directory exists and that you have access to it.

`Could not open/create id2entry`*attr*

The specified file cannot be opened. Check that the root directory for the data store is defined correctly. Check that the directory exists and that you have access to it.

`Could not write next id` *number*

Error running `idxgen`. Check that the database directory is configured correctly. Check that the database directory exists and has suitable permissions. Make sure that there is sufficient disk space available.

`db_errno` *number*

Database problem. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`Decoding error`

An invalid LDAP PDU was received.

`dn2id_add failed`

Database error. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`dn_normalize - unknown state` *number*

Internal error. The supplied distinguished name is invalid.

`DSA is busy`

Retry the operation. Check for other errors indicating a shortage of resources required by the directory server.

`DSA is unavailable`

Retry the operation. Check for other errors indicating a shortage of resources required by the directory server.

`DSA is unwilling to perform`

Retry the operation. Check for other errors indicating a shortage of resources required by the directory server.

`Encoding error`

An LDAP request or response could not be encoded. Check that there is sufficient memory available.

`Entry (`*dn*`), attr (`*attribute*`) not allowed`

The named entry contains an attribute that is not permitted for that objectClass.

`Entry (`*dn*`), required attr (`*attribute*`) missing`

The named entry is missing a value for a mandatory attribute.

`Error: cannot acquire lock on` *file* `for trimming`

Check the file permissions.

`Error: cannot create status file` *file*

Check the directory and file permissions.

`Error: cannot open status file` *file*`:` *error*

Use the information in *error* to diagnose and fix the problem.

`error: can't lock file` *file*`:` *error*

Check the file permissions.

`error: can't seek to offset` *number* `in file` *file*

Internal format error in replication log file.

`Error: copy_replog: Can't lock replog` *replogfile* `for read:` *error*

Check for errors in your operating environment. Use the information in *error* to diagnose and fix the problem.

`Error: copy_replog: Can't lock replog` *replogfile* `for write:` *error*

Check for errors in your operating environment. Use the information in *error* to diagnose and fix the problem.

`Error: copy_replog: Error closing` *replogfile*

Check for errors in your operating environment. Use the information in *error* to diagnose and fix the problem.

`Error: copy_replog (`*number*`): Directory /usr/tmp is not writable`

Check that `/usr/tmp` is accessible. Check for errors in your operating environment.

`Error: do_bind: ldap_unbind failed:` *error*

Use the information in *error* to diagnose and fix the problem.

`Error: do_bind: null ri ptr`

Internal error. Contact your authorized service provider.

`Error: do_bind: unknown auth type` *bindmethod* `for` *hostname*`:`*port*

The bind method is not correctly configured in `slapd.conf`.

`Error: do_ldap: bad op` *changetype*`, dn =` *dn*

The specified entry could not be modified. The modification requested is not appropriate for the entry.

`Error: do_unbind: ldap_unbind failed for` *hostname*`:`*port*`:` *error*

Use the information in *error* to diagnose and fix the problem.

`Error: `*file*`: file not readable`

You do not have read access to the specified file. Check that the file is specified correctly. Check that the file and directory permissions are set correctly.

`Error: `*file*`: file not writable`

You do not have write access to the specified file. Check that the file is specified correctly. Check that the file and directory permissions are set correctly.

`Error in poll errno=`*number*

Internal error in SNMP daemon. Contact your authorized service provider.

`Error: ldap operation failed, data written to` *file*

Check for other error messages indicating the operation that failed. Use the data in *file* to retry the operation.

`Error: ldap_add_s failed adding "`*dn*`":` *error*

The `ldapadd` operation failed. Use the information in *error* to diagnose the problem.

`Error: ldap_delete_s failed deleting` *entry*`:` *error*

Use the information in *error* to diagnose and fix the problem.

`Error: ldap_modify_s failed modifying` *entry*`:` *error*

Use the information in *error* to diagnose and fix the problem.

`Error: ldap_modrdn_s failed modifying` *entry*`:` *error*

Use the information in *error* to diagnose and fix the problem.

`Error: ldap_open(`*hostname*`,` *port*`) failed:` *error*

Use the information in *error* to diagnose and fix the problem.

`Error: ldap_simple_bind_s for` *hostname*`:`*port* `failed:` *error*

Use the information in *error* to diagnose and fix the problem.

`Error: malformed modify op,` *passedtype*`:` *value* `(expecting` *exptype*`:)`

The modification operation cannot be complete with the specified type.

`error: malformed replog entry (begins with` *string*`)`

There is an incorrect entry in the replication log file.

`Error: malformed replog line` *number*

There is an error in the specified line of the replication log file. Correct the line and resynchronize the replica. If you cannot correct the problem, make a copy of the `slapd.replog` file and contact your authorized service provider.

`Error: parse_replica_line: unknown keyword` *keyword*

There is an unrecognized keyword in the replication log file.

`Error:` *path*`: directory does not exist`

The specified directory does not exist. Check that the directory is specified correctly.

`Error:` *path*`: directory not readable`

You do not have read access to the specified directory. Check that the directory is specified correctly. Check that the directory permissions are set correctly.

`Error: re is null in Ri_process`

Internal error. Contact your authorized service provider.

`Error: Re_parse: bad type` *type*

Internal error. Make a copy of the `slapd.replog` file and contact your authorized service provider.

`Error: Re_parse: malformed replog file`

The replication log file has been corrupted. Make a copy of the `slapd.replog` file and contact your authorized service provider.

`Error: Ri_process: ri == NULL!`

Internal error. Contact your authorized service provider.

`Error truncating replication log:` *replogfile*

The *replogfile* cannot be re-initialized. Check for errors in your operating environment.

`Error while writing:` *error*

Error while copying the replication log file. Use the information in *error* to diagnose and fix the problem.

`Error: write_reject: Cannot create` *file*`:` *error*

Cannot write to the specified file. Use the information in *error* to diagnose and fix the problem.

`Errors encountered while processing config file "`*file*`"`

The configuration file cannot be processed. Check that the file exists and that you have access to it. Check for operating environment errors indicating the reason why the file cannot be processed.

`Failed to add status element structure`

Fatal error running `slurpd`. Check the log files for other information.

`Fatal error: no "replogfile" directive given`

No replication log file is specified in `slapd.conf`.

`Fatal error while copying replication log`

There is a format error in the replication log file.

*file*`: db_errno` *number*

Database problem. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

*file*`: line` *number*`: unknown hashing method "`*hash*`"`

The hashing method must be either *none* or *crypt*.

`FIONBIO ioctl failed on` *socket*

An error occurred when configuring the communications socket.

`FIONBIO ioctl on` *number* `failed`

Internal error. Contact your authorized service provider.

`id` *number* `already in next block`

Database error. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*. Check whether any data has been lost.

`id2children_add failed`

Database error. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`id2entry_add failed`

Database error. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`idl_fetch of (`*string*`) returns NULL`

Internal error. Contact your authorized service provider.

`idl_fetch_one (`*string*`) returns NULL`

Internal error. Contact your authorized service provider.

`idl_insert_key (`*string*`)` *number* `failed`

Database problem. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`Illegal aliasing: Entry` *aliasdn*`, aliasedObjectName` *dn*

An alias entry is not defined correctly, or the aliasedObject is itself an alias.

`Impossible case in dn2entry_deref`

Internal error. Contact your authorized service provider.

`Inappropriate matching`

A specified matching rule is not appropriate for the attributes involved in the operation. For example, an approximate match is not appropriate for an attribute whose values are in binary syntax.

`incoming connection refused, out of file descriptors`

Retry the operation. Check your operating environment to see whether more file descriptors could be configured.

`index_add_entry failed`

Database error. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`Insufficient access`

You do not have permission to access an entry required for this operation.

`Internal error: Re_write: NULL argument`

Internal error. Contact your authorized service provider.

`invalid base 64 encoding char (`*string*`)` *encodedstring*

The `ldif` file is invalid: binary data is not base64 encoded. Pass the binary data to the utility in a file specified by the `-f` option.

`Invalid credentials`

Either the distinguished name or the password specified are not valid.

`Invalid DN syntax`

A distinguished name is incorrectly specified.

`Invalid traversal type` *number*

Internal error. Contact your authorized service provider.

*ldapadd or ldapmodify*`: unknown` *item* `"`*name*`" (line` *number* `of entry:` *dn*`)`

`ldapadd` or `ldapmodify` could not complete the operation because of the unknown item reported. Check that the operation is specified correctly. Check that all attributes mentioned are present in the schema definition.

*ldapadd or ldapmodify*`: expecting "`*item1*`" but saw "`*item2*`" (line` *number* `of entry` *dn*`)`

`ldapadd` or `ldapmodify` could not complete the operation because of the unknown item reported. Check that the operation is specified correctly. Check that all attributes mentioned are present in the schema definition.

*ldapadd or ldapmodify*`: extra lines at end (line` *number* `of entry` *dn*`)`

The definition of the specified entry contains extra lines, so the operation cannot be completed. Check that the operation is specified correctly.

*ldapadd or ldapmodify*`: missing value on line` *number* `(attr is` *attribute*`)`

The operation cannot be completed because no value is provided for the specified attribute.

`ldapmodify: skipping change record for entry:` *dn*
`(LDAP host/port does not match replica: lines)`

There is a mismatch between the host and port defined for the replica in the master server configuration and the host and port used.

`listen() failed errno` *number* `(`*error*`)`

The port is already in use by another application. Check that the port number is specified correctly.

`listener pthread_create failed`

Cannot create thread for listening on incoming socket. Stop and restart `slapd`, as described in “Maintaining the Directory Service” on page 241.

`Local error`

Check for other errors, both in the directory service and in the operating environment, and make sure those problems are resolved. Retry the operation. If this error message is returned frequently, contact your authorized service provider.

`Loop detected`

Loop detected when dereferencing an alias. Check that the alias entry is defined correctly.

`Malformed slurpd status file` *file*

The file `/usr/tmp/slurpd.status` is invalid and should be deleted.

`malloc of` *number* `bytes failed`

Insufficient memory available.

`merged parent (id` *number*`) error info:`

An error occurred when merging the results returned after a referral.

`more than` *hopcount* `referral hops (dropping)`

The query has been forwarded more than the specified maximum number of time and will be returned to the user with a error message.

`Naming violation`

An invalid distinguished name was encountered.

`next_id` *number*`: cannot fclose`

Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`next_id` *number*`: cannot fgets nextid from` *file*

Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`next_id` *number*`: cannot fprintf`

Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`next_id` *number*`: could not fgets nextid from` *file*

Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`next_id` *number*`: could not open` *file*

Either the database is empty or the `NEXTID` file has been removed. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`next_id_return of` *number*`: cannot fclose` *file* `next id` *number*

Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`next_id_return of` *number*`: cannot fprintf` *string* `next id` *number*

Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`next_id_return of` *number*`: could not open` *file* `next id` *number*

Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*.

`no access to parent`

Access control error. Check that the access controls are defined correctly. Check your authentication to the directory.

`No License`

There is no license available for the directory server.

`No object class for entry (`*dn*`)`

There is no value for the objectClass attribute of the entry.

`no parent & not root`

You are attempting to create the root entry of a tree. Only the root user (cn=admin be default) is permitted to create this entry.

`No replicas in slapd config file` *file*`!`

No replicas are defined for this data store, so no synchronization is necessary.

`No such object`

A specified entry does not exist in the directory. Check that the operation is specified correctly.

`no values for type` *type*

There is no value for the specified attribute.

`nonexistent continuation block (`*string*`)`

Database error. Use `idxgen` to regenerate the index files, as described in the *SIMS Reference Manual*. Check whether any data has been lost.

`not enough pattern space`

An LDAP filter contains a regular expression that is too complicated.

`null ref in (`*file*`)`

No value is defined for the referral attribute.

`Object class violation`

An attribute has been specified that is not permitted in this object class.

`Object is a leaf`

The operation you are trying to perform is not appropriate for an entry that does not have child entry. Make sure you have specified the operation and entry correctly.

`op_delete: can't find op` *number*

Internal error. Contact your authorized service provider.

`open /dev/null failed errno` *error*

Use the information in the *error* to diagnose and fix the problem.

`Operation not allowed on nonleaf`

The operation you are trying to perform is not appropriate for an entry that has a child entry. Make sure you have specified the operation and entry correctly.

`Operation not allowed on RDN`

The operation you are trying to perform would change the RDN. If you want to change the value of an attribute used in an RDN, use `ldapmodrdn` and not `ldapmodify`.

`Operations error`

Check for other error messages explaining the specific problem.

`Out of memory`

Check your operating environment. If the operation you are performing is a search, try specifying more exact criteria so that a smaller number of results are returned.

`out of memory, add_replica`

Check for messages concerning memory in your operating environment. If possible, make more memory available to the directory server.

`Out of memory in get_repl_hosts`

Check for messages concerning memory in your operating environment. If possible, make more memory available to the directory server.

`Out of memory initializing globals`

Check for messages concerning memory in your operating environment. If possible, make more memory available to the directory server.

`out of memory, Ri_init`

Check for messages concerning memory in your operating environment. If possible, make more memory available to the directory server.

`parent does not exist`

The entry is not present in the DN list. Your database might be corrupt. Use `idxgen` to regenerate the index files, as described in *SIMS Reference Manual*.

`parse_line missing ':'`

There is a syntax error in the `ldif` file: the colon (:) is missing from a line.

`parse_line missing value`

There is a syntax error in the `ldif` file: an attribute has no value.

`Protocol error`

Check for other error messages explaining the specific problem.

`pthread_create failed`

Cannot create a new thread in the `slapd` server. Stop and restart `slapd`, as described in “Maintaining the Directory Service” on page 241.

`realloc of` *number* `bytes failed`

Insufficient memory available.

`Re_parse: error: re is NULL`

Internal error. Contact your authorized service provider.

`Re_parse: error: replbuf is NULL`

Internal error. Contact your authorized service provider.

`replica` *hostname*`:`*port* `pthread_create failed`

Cannot create replication thread. Stop and restart the directory server, as described in “Maintaining the Directory Service” on page 241.

`result errno` *number*`, error` *error*`, matched` *matchinginfo*

An error occurred when merging the results returned after a referral.

`Results too large`

The result set for the operation is too large. Partial results might be returned. If the operation you are performing is a search, try specifying more exact criteria so that a smaller number of results are returned.

`RL error`

Internal error. Contact your authorized service provider.

`Rq_dump: cannot open` *file* `for write`

Check for errors in your operating environment.

`RR error`

Internal error. Contact your authorized service provider.

`select failed errno` *number* (*error*)

Internal error. Contact your authorized service provider.

`setsockopt() failed errno` *number* (*error*)

Cannot create listening socket. Stop and restart `slapd`, as described in “Maintaining the Directory Service” on page 241. Check for errors in your operating environment or networking software.

`Slapd refuses registration`

The SNMP daemon cannot communicate with the `slapd` daemon. Stop and restart `slapd`, as described in “Maintaining the Directory Service” on page 241.

`Slapd refuses trap registration`

The SNMP daemon cannot communicate with the `slapd` daemon. Stop and restart `slapd`, as described in “Maintaining the Directory Service” on page 241.

`slapd.at.conf: line` *number*`: duplicate attribute`

The attribute defined at the specified line has already been defined in the file.

`snmp getmsg error errno=`*number*

Cannot communicate with SNMP daemon.

`snmp initmbx failed, aborting`

Cannot communicate with SNMP daemon.

`snmp malloc failed errno=`*number*

Insufficient memory available.

`snmp mkfifo` *string* `failed errno=`*number*

Cannot communicate with SNMP daemon.

`snmp open failed errno=`*number*

Cannot communicate with SNMP daemon.

`snmp open` *string* `failed errno=`*number*

Cannot communicate with SNMP daemon.

`snmp poll error errno=`*number*

Cannot communicate with SNMP daemon.

`snmp pthread_create failed`

Cannot create thread for SNMP management. Stop and restart `slapd`, as described in “Maintaining the Directory Service” on page 241.

`snmp putmsg failed errno=`*number*

Cannot communicate with SNMP daemon.

`snmp trap putmsg failed errno=`*number*

Cannot communicate with SNMP daemon.

`snmp Unknown command received` *string*

Internal error. Contact your authorized service provider.

`snmp unknown register command received` *string*

Internal error in SNMP daemon.

`socket() failed errno` *number* (*error*)

Cannot create listening socket. Stop and restart `slapd`, as described in “Maintaining the Directory Service” on page 241”. Check for errors in your operating environment or networking software.

`Timed out`

Check for other error messages indicating resource shortages in the directory service and in your operating environment. If the operation you are performing is a search, try specifying more exact criteria so that a smaller number of results are returned.

`Too many tokens (max 100)`

The configuration file contains a line that has more than 100 parameters.

`Type or value exists`

You have specified a value that already exists for an attribute, or you have specified a value for a single-valued attribute that is already present in the entry.

`Undefined attribute type`

An attribute type that is not defined in the schema has been specified.

`Unknown error`

Check for other errors, both in the directory service and in the operating environment, and make sure those problems are resolved. Retry the operation. If this error message is returned frequently, contact your authorized service provider.

`unknown filter type` *number*

An LDAP filter contains an invalid attribute type.

`unknown fmt` *string*

Internal error. Contact your authorized service provider.

`Unknown message type received on trap fd`

Internal error in SNMP daemon. Contact your authorized service provider.

`unknown request` *pdu*

Invalid LDAP PDU received.

`unknown version` *versionnumber*

The LDAP protocol version supplied in the bind request is not valid. Only LDAPV2 is supported.

`Warning: failed to add replica` *hostname*`:`*port* `- ignoring replica`

Error running `slurpd`. Check that the named host is reachable. Check the log files on both systems for other information.

`Warning: freeing re (dn:` *dn*`) with nonzero refcnt`

Internal error.

`Warning, license lost`

There is no license available for the directory server.

`Warning, license still lost`

There is no license available for the directory server.

# Directory Service Error Messages Returned by the Admin Console

*accessrights* `: Invalid access rights. Ignore by clause`

The access rights specified are not valid and are ignored. Correct the access control definition.

`Acl creation failed`
*failure reason*

The access control list could not be created because of the reason given. Use the information in *failure reason* to diagnose and fix the problem.

`ACL generation failed :` *failure reason*

The access control list could not be generated because of the reason given. Use the information in *failure reason* to diagnose and fix the problem. If the problem recurs, contact your authorized service provider.

`Acl update failed`
*failure reason*

The access control list could not be updated because of the reason given. Use the information in *failure reason* to diagnose and fix the problem.

`At least one access rule required`

You have not supplied any access control rules. At least one access control rule is required.

*attribute* `: duplicated attribute in class` *class*

The attribute *attribute* is already present in this object class.

*attribute* `: invalid index. Assume no index`

The index definition for this attribute is incorrect. No index will be generated.

`Attribute` *attribute* `not editable`

You cannot change the definition of this attribute.

`Attribute generation failed :` *failure reason*

The attribute could not be generated. Use the information in *failure reason* to diagnose and fix the problem. If the problem recurs, contact your authorized service provider.

`Attribute update failed`
*failure reason*

The attribute could not be updated. Use the information in *failure reason* to diagnose and fix the problem.

`Backend creation failed`
*failure reason*

The data store could not be created. Use the information in *failure reason* to diagnose and fix the problem.

`Backend generation failed :` *failure reason*

The data store could not be created. Use the information in *failure reason* to diagnose and fix the problem. If the problem recurs, contact your authorized service provider.

`Backend update failed`
*failure reason*

The data store could not be updated. Use the information in *failure reason* to diagnose and fix the problem.

`Backup of configuration failed`
(*failure reason*)

Your configuration cannot be backed up. Use the information in *failure reason* to diagnose and fix the problem.

`Backup of Data Store failed`
*failure reason*

Your data store cannot be backed up. Use the information in *failure reason* to diagnose and fix the problem.

`Blank characters not supported in alias`

The alias specified contains blank characters, which are not permitted. Enter a different value for the alias.

`Can't access directory` *directory*

The directory specified cannot be accessed. Check the path name is correct. Check the file protection.

`Can't backup current configuration to` *location* (*failure reason*)

You configuration cannot be backed up to the location specified. Use the information in *failure reason* to diagnose and fix the problem.

`Can't create directory` *directory*

The specified directory cannot be created. Check that the path name is correct. Check the file protection.

`Can't delete index file` *file*`:` *failure reason*

The specified index file cannot be deleted. Use the information in *failure reason* to diagnose and fix the problem.

`Can't instantiate AuditTrail object (`*failure reason*`)`

Internal error. Contact your authorized service provider.

`Can't load property file` *file* `(`*failure reason*`)`

The specified property file cannot be loaded. Use the information in *failure reason* to diagnose and fix the problem.

`Can't restore data store from` *location*`: Missing files`

The data store cannot be restored from the specified location because at least one file is missing. A data store directory must contain the following files:

- `NEXTID`
- `dn2id.dbb`
- `id2children.dbb`
- `id2entry.dbb`
- Optionally, one or more attribute index files, called *attribute*`.dbb`

`Can't save property file to` *location* `(`*failure reason*`)`

The specified property file cannot be saved. Use the information in *failure reason* to diagnose and fix the problem.

`Can't start LDAP/HTTP gateway (`*failure reason*`)`

The LDAP/HTTP gateway process cannot start. Use the information in *failure reason* to diagnose and fix the problem.

`Can't start slapd process (`*failure reason*`)`

The `slapd` process cannot start. Use the information in *failure reason* to diagnose and fix the problem.

`Can't start slurpd process (`*failure reason*`)`

The `slurpd` process cannot start. Use the information in *failure reason* to diagnose and fix the problem.

`Can't stop slurpd process (`*failure reason*`)`

The `slurpd` process cannot stop. Use the information in *failure reason* to diagnose and fix the problem.

*class* `: duplicated object class`

The specified object class is defined twice. Remove one of the definitions.

`Class generation failed :` *failure reason*

Use the information in *failure reason* to diagnose and fix the problem. If the problem recurs, contact your authorized service provider.

`Configuration update failed (`*failure reason*`)`

The configuration cannot be updated. Use the information in *failure reason* to diagnose and fix the problem.

`Congested limit must be smaller than Normal limit`

The value supplied for the congested limit must be smaller than the normal limit. Change the value of one of these limits. See "To Create a Data Store" on page 192 for information about setting congestion thresholds.

`Critical limit must be smaller than Congested limit`

The value supplied for the critical limit must be smaller than the congested limit. Change the value of one of these limits. See "To Create a Data Store" on page 192 for information about setting congestion thresholds.

`crontab file update failed (`*failure reason*`)`

The update to the `crontab` failed, so the replication schedule will not be applied. Use the information in *failure reason* to diagnose and fix the problem.

`Data Store could not be restored`
*failure reason*

The data store cannot be restored. Use the information in *failure reason* to diagnose and fix the problem.

`Delete Index failed (operation forbidden on default index)`

You cannot delete the default index. Check that you have selected the correct index.

*directory* `: invalid directory`

The directory you have specified does not exist or you do not have access to it.

`Duplicated naming context` *name*

The naming context you have specified already exists. Check that you have specified the naming context correctly.

`Error detected while starting directory services`
*error*

The directory service cannot start. Use the information in *error* to diagnose and fix the problem.

`Error detected while stopping directory services`
(*error*)

An error was detected while stopping the directory service. Use the information in *error* to diagnose and fix the problem.

`Error found in file` *file* `at line` *number* `:` *error*

The specified file has an error at the specified line. Use the information in *error* to diagnose and fix the problem.

`Error found while parsing line` *number* `: Extra token ignored`

There is an extra token in the specified line of the file being parsed. This token is ignored. This message is usually accompanied by another message that identifies the file being parsed.

`Error found while parsing line` *number* `: missing token`

There is a token missing in the specified line of the file being parsed. This message is usually accompanied by another message that identifies the file being parsed.

`Error while moving backend` *datastore* `to` *directory* `(file already exists)`

You cannot move the specified data store to the specified directory. A file that would be created already exists.

`Error while moving data store` *name* `to` *location* (*error*)

You cannot move the specified data store to the specified location. Use the information in *error* to diagnose and fix the problem.

`Error while saving configuration` (*error*)

The configuration cannot be saved. Use the information in *error* to diagnose and fix the problem.

`Error while saving configuration to` *location* `(`*error*`)`

The configuration cannot be saved to the specified location. Use the information in *error* to diagnose and fix the problem.

`Error while saving server configuration to` *location* `(`*error*`)`

The configuration cannot be saved to the specified location. Use the information in *error* to diagnose and fix the problem.

`Error while starting synchronization process (`*error*`)`

The replication synchronization cannot be started. The configuration cannot be saved to the specified location. Use the information in *error* to diagnose and fix the problem.

*file* `is not writable`

Information cannot be written to the specified file. Check the file and directory protections.

`Get class attribute failed`
*failure reason*

The attributes for an object class could not be read from the configuration file. Use the information in *failure reason* to diagnose and fix the problem.

`Impossible to backup the configuration to` *directory*`. Choose another directory for backup`

You cannot back up the configuration to the specified directory. Check that you have specified the directory correctly. Check the file and directory protections. Use a different directory for backing up the configuration.

`Inconsistent bind method`

The bind method you have specified is not consistent with the information already configured or stored in the directory.

`Inconsistent host value (equal to localhost)`

The host value you specified is the name of the local host. Check that this is appropriate.

`Index regeneration failed (`*command: error*`)`

An attribute index could not be regenerated automatically. Try regenerating the index using the command given and see whether the error is repeated.

`Index update failed`
*failure reason*

An attribute index could not be updated. Use the information in *failure reason* to diagnose and fix the problem.

`Invalid address. Ignore by clause`

The address you have specified in the “access by” clause is not valid. Check that you have entered the address correctly.

`Invalid administrative session`

You are not correctly authenticated to the Admin Console. Stop and restart the Admin Console. If the problem recurs, contact your authorized service provider.

`Invalid Administrator Name`

The administrator name you specified is not valid.

`Invalid Administrator Name. (Invalid characters)`

The administrator name you specified contains invalid characters.

`Invalid attribute`

You have specified an invalid attribute.

`Invalid attribute` *attribute*

The specified attribute is not valid.

`Invalid attribute list`

You have specified an invalid attribute list.

`Invalid attribute list (`*attribute*`: unknown attribute)`

You have specified an attribute list that contains an unknown attribute.

`Invalid attribute name`

You have specified an invalid or unrecognized attribute.

`Invalid attribute name (attribute does not exist)`

The attribute you specified does not exist. Check that you have specified the attribute name correctly.

`Invalid attribute syntax`

The attribute syntax you have specified is not valid. See the *SIMS Reference Manual* for a list of valid attribute syntaxes.

`Invalid bindderef value`

The setting for bindderef in the configuration file is not valid. The value should be True or False.

`Invalid binddn clause not in a replica clause`

The configuration file has been edited manually and a binddn is specified without being associated with a replica.

`Invalid bindmethod clause not in a replica clause`

The configuration file has been edited manually and a bindmethod is specified without being associated with a replica.

`Invalid by clause`

The configuration includes an invalid by clause in an access control definition.

`Invalid cache size (`*size*`)`

The cache size you have specified is not valid. Specify a value between 1 and 1,000,000.

`Invalid cachesize clause ignored`

The cache size you have specified is not valid. The default cache size is used.

`Invalid characters in attribute name`

An attribute name contains invalid characters.

`Invalid class name`

A class name you have specified is invalid.

`Invalid congestion management values (`*value*`)`

One of the congestion thresholds you have specified is not valid. See “To Create a Data Store” on page 192 for information about setting these values.

`Invalid credentials clause not in a replica clause`

The configuration file has been edited manually and a distinguished name and password are specified without being associated with a replica.

`Invalid DB directory value (not a directory)`

The database directory you specified is not valid. Check that the directory exists and that the file and directory protections are suitable.

`Invalid dbcachesize`

The cache size you have specified is not valid.

`Invalid dbcachesize clause ignored`

The cache size you have specified is not valid. The default cache size is used.

`Invalid directory` *directory*`. Use /tmp`

The data store directory you have specified is not valid.

`Invalid distinguished name (`*dn*`)`

The specified distinguished name is not valid. Check that you have entered the DN correctly.

`Invalid DN`

An invalid DN has been specified.

`Invalid DN (`*dn*`)`

The specified distinguished name is not valid. Check that you have entered the DN correctly.

`Invalid dn. Ignore by clause`

The DN specified in the *access by* clause is not valid. This clause is ignored.

`Invalid dnattr. Ignore by clause`

The DN attribute specified in the *access by* clause is not valid. This clause is ignored.

`Invalid domain. Ignore by clause`

The domain name specified in the *access by* clause is not valid. This clause is ignored.

`Invalid domain name`

An invalid domain name has been specified.

`Invalid exclattr clause`

The set of attributes to be excluded from the replica is not specified correctly in the configuration file.

`Invalid` *file* `command file`

The specified file does not exist or cannot be read. Check that the file exists and that suitable file protection is set.

`Invalid filter`

An invalid filter has been specified.

`Invalid host name`

An invalid host name has been specified.

`Invalid inclattr clause`

The set of attributes to be included in the replica is not specified correctly in the configuration file.

`Invalid index clause ignored`

An index clause is not valid and will be ignored.

`Invalid IP address`

An invalid IP address has been specified.

`Invalid LDAP port value (`*portnumber*`)`

The specified port number is not valid. Check that you have specified the port number correctly. Check that the port is not being used by another application.

`Invalid log file`

An invalid log file has been specified.

`Invalid log file size (`*value*`)`

The specified value for log file size is not valid.

`Invalid logfile size`

The logfile size is not valid.

`Invalid Maximum concurrent connection value (`*value*`)`

The specified value for the maximum number of concurrent connections is not valid.

`Invalid mode clause ignored`

The configuration file does not specify the mode of a data store correctly. A data store can be read-only or read-write.

`Invalid naming attribute length (`*length*`)`

The length of the specified naming attribute is not valid. Check that the attribute value is specified correctly.

`Invalid naming attribute size`

A naming attribute has an invalid size. Check that the attribute value is specified correctly.

`Invalid Naming Context (`*name*`)`

The naming context you have specified is not valid. Check that you have specified the naming context correctly.

`Invalid referral clause`

The referral clause in the configuration file is not correctly specified.

`Invalid replica port. Use 389`

The port you specified for `slurpd` to use is not valid. The default port, number 389, will be used.

`Invalid replica specification`

A replica is not specified correctly.

`Invalid replicadn clause`

The DN of a replica is not specified correctly.

`Invalid Search size limit (`*limit*`)`

The specified search size limit is not valid. Specify a value between 1 and 1,000,000.

`Invalid Search time limit (`*value*`)`

The specified search time limit is not valid. Specify a value between 1 and 1,000,000.

`Invalid subtree clause`

A subtree is specified incorrectly in the configuration file.

`Invalid user ID or password`

The login name or password specified for login are incorrect.

`Invalid Web gateway port value (`*port*`)`

The port that you specified for the LDAP/HTTP gateway is not valid.

`LDAP admin component initialization failed (`*failure reason*`)`

The LDAP Admin Console could not be initialized. Use the information in *failure reason* to diagnose and fix the problem.

`LdapConfigException in generate_crontab_file (`*exception*`)`

Use the information in *exception* to diagnose and fix the problem.

`LdapConfigException in load_slurpd_synchro (`*exception*`)`

Use the information in *exception* to diagnose and fix the problem.

*limit* `: invalid time limit value`

The specified time limit is not valid. Specify a value between 1 and 1,000,000.

*loglevel* `: invalid loglevel (integer expected)`

The specified log level is invalid. See `slapdcmd(1M)` for a list of available log levels.

`Missing access right`

The access control definition is incomplete. You have not specified the access right to be granted.

`Missing admin password`

You have not set a password for the administrator.

`Missing admin password mode`

You have not specified whether or not the administrator's password is encrypted.

`modify replica: internal error`

Internal error. Contact your authorized service provider.

`modify subtree: internal error`

Internal error. Contact your authorized service provider.

*name* `is not a directory`

The directory you have specified does not exist or cannot be accessed. Check that the directory exists and has suitable protection set.

`Naming Context` *name* `already in use`

There is already a naming context with the name you specified.

`Object Class creation failed`
*failure reason*

An object class cannot be created. Use the information in *failure reason* to diagnose and fix the problem.

`Object Class update failed`
*failure reason*

An object class cannot be updated. Use the information in *failure reason* to diagnose and fix the problem.

`Restore configuration failed (`*failure reason*`)`

You cannot restore the requested configuration. Use the information in *failure reason* to diagnose and fix the problem.

`rootdn:Invalid DN`

The DN you have specified is not valid.

`rootdn:Invalid DN (`*dn*`)`

The specified DN is not valid.

`Synchronization failed:` *failure reason*

A replica could not be synchronized. Use the information in *failure reason* to diagnose and fix the problem.

`The attribute` *attribute* `is not editable`

You are trying to modify an attribute that is frozen in the schema. You cannot change elements of the schema that are frozen.

`The attribute list has to contain at least one attribute`

You have specified an empty attribute list.

`The object class` *class* `is not editable`

You are trying to modify an object class that is frozen in the schema. You cannot change elements of the schema that are frozen.

`This attribute is not editable`

You are trying to modify an attribute that is frozen in the schema. You cannot change elements of the schema that are frozen.

`This naming context is not included in the Data Store naming context` *name*

The naming context is not a subtree of the specified naming context.

`This object class is not editable`

You are trying to modify an object class that is frozen in the schema. You cannot change elements of the schema that are frozen.

`This replica is not included in the Data Store naming context` *name*

The replica is not a subtree of the specified naming context.

`Unable to restore configuration`
`Missing` *file* `configuration file in directory` *directory*

The configuration cannot be restored. The specified file is missing from the specified directory.

`Unknown DN Error`

A distinguished name is specified incorrectly. Try using the DN editor to construct the name.

*url* `: invalid ldap url`

An invalid URL has been specified.

*value* `: invalid default access value`

The default access value is not valid.

*value* `: invalid lastmod value`

The setting for lastmod is not valid.

*value* `: invalid readonly value`

The setting for readonly is not valid.

*value* `: invalid schemacheck value`

The value for schemacheck is not valid.

*value* `: invalid size limit value`

The size limit value is not valid.

# Glossary

**abstract syntax** A description of a data structure that is independent of machine-oriented structures and encoding.

**ACAP** A protocol which enhances IMAP by allowing the user to set up address books, user options, and other data for universal access.

**access control rules** Rules that define which users are granted which permissions for a given set of directory entries or attributes.

**ACSE** Association Control Service Element. The method used in OSI for establishing a call between two applications. Checks the identities and contexts of the application entities, and could apply an authentication security check.

**Administration Console or Admin Console** A GUI (graphical user interface) which enables you to configure, monitor, maintain, and troubleshoot the SIMS components.

**address mapping** See forward address mapping or reverse address mapping.

**address resolution** A means for mapping Network Layer addresses onto media-specific addresses. See also *ARP*.

**address token** The address element of a rewrite rule pattern.

**ADMD** Administration Management Domain. An X.400 Message Handling System public service carrier. Examples: MCImail and ATTmail in the U.S., British Telecom Gold400mail in the U.K. The ADMDs in all countries worldwide together provide the X.400 backbone. See also *PRMD*.

**Administration Services** Administers all components of SIMS through a JMAPI-based GUI. See also *JMAPI*.

**agent** In the client-server model, the part of the system that performs information preparation and exchange on behalf of a client or server application. See also *NMS*, *DUA*, *MTA*.

**alias** An address which delivers messages to a specified group of users. A listserve.

**alias entry** Contains the name of the directory entry it represents in the directory information tree, and can also contain other attributes. It is identified by the distinguished name.

**alias file** A file used to set aliases not set in a directory, such as the postmaster alias.

**ANSI** American National Standards Institute. The U.S. standardization body. ANSI is a member of the International Organization for Standardization (ISO).

**API** Application Program Interface. A set of calling conventions defining how a service is invoked through a software package.

**Application Layer** The top-most layer in the OSI Reference Model providing such communication services as electronic mail and file transfer.

**ASM** Application Specific Module. An example of this is an external Solaris Backup.

**ASN.1** Abstract Syntax Notation One. The OSI language for describing abstract syntax. See also *BER*.

**attribute** The form of information items provided by the Directory Service. The directory information base consists of entries, each containing one or more attributes. Each attribute consists of a type identifier together with one or more values. Each directory Read operation can retrieve some or all attributes from a designated entry.

**attribute index** An index, or list, of entries which contains a given attribute or attribute value.

**autoreply option file** A file used for setting options for autoreply, such as vacation notices

**AVM** Admin View Module. An extension of Java's Abstract Window Toolkit that provides the Administration Console's graphical user interface.

**AWT** Abstract Window Toolkit. A Java development toolkit.

**backbone** The primary connectivity mechanism of a hierarchical distributed system. All systems that have connectivity to an intermediate system on the backbone are connected to each other. This does not prevent you from setting up systems to bypass the backbone for reasons of cost, performance, or security.

**backend** Stores directory information. There are two types of backends; LDBM which provides access to information stored in a database, and shell which provides access to information stored in any format.

**backup** The process of dumping the contents of folders from the Sun Message Store to a backup device. See also *purge* and *restore*.

**BER** Basic Encoding Rules. Standard rules for encoding data units described in ASN.1. Sometimes incorrectly lumped under the term ASN.1, which properly refers only to the abstract syntax description language, not the encoding technique.

**big-endian** A format for storage or transmission of binary data in which the most significant bit (or byte) comes first. The reverse convention is called little-endian.

**BOC** Bell Operating Company. More commonly referred to as RBOC for Regional Bell Operating Company. The local telephone company in each of the seven U.S. regions.

**CA** Certificate Authority. An organization that issues digital certificates (digital identification) and makes its public key widely available to its intended audience.

**cache** A temporary storage file of information that has been retrieved from the directory.

**CCITT** See also *ITU*.

**chaining** The directory server passes an information request to the Directory Service Agent (DSA) that can process the request. The second DSA returns the result to the first DSA, which then returns it to the client. See also *knowledge information*.

**channel** An interface with another Sun Internet Mail Server version component, another email system, or a mail user agent.

**character set labels** SIMS can be configured to process either 7 or 8 bit character sets, by using the menus in the Channel Property book. This configuration will affect encrypted and possibly garbled messages received from other systems. For more detailed instructions, see pg. 124 of the Adminstator's Guide.

**ciphertext** Data that has been coded (enciphered, encrypted, or encoded) for security purposes.

**client-server model** A common way to describe network services and the model user processes (programs) of those services. Examples include the name-server / name-resolver paradigm of the DNS and fileserver / file-client relationships such as NFS and diskless hosts.

**composition** The process of constructing a message by the Mail User Agent (MUA). See also *MUA*.

**congestion thresholds** A limit on disk space set by the system administrator which prevents the database from becoming overloaded by restricting new operations when system resources are insufficient.

**content** The content of a message provides the data that the originator of the message intends to transmit to the recipient.The content of a message can contain text as well as images, audio, video, and binary or application-specific files.

**content-transfer encoding** Specifies how data is encoded so the data can traverse Internet Mail Transport Agents (IMTAs) outside of the SIMS email system that may have data or character-set limitations.

**conversion channel** Converts body of messages from one form to another.

**cross reference** Any naming context that can be contacted directly. See also *knowledge information*.

**data store** A store that contains directory information, typically for an entire directory information tree.

**DCE** Distributed Computing Environment. An architecture of standard programming interfaces, conventions, and server functionalities (e.g., naming, distributed file system, remote procedure call) for distributing applications transparently across networks of heterogeneous computers. Promoted and controlled by the Open Software Foundation (OSF), a consortium led by HP, DEC, and IBM. See also *ONC*.

**defragmentation** The Multiple Internet Extensions (MIME) feature that enables a large message that has been broken down into smaller messages or fragments to be reassembled. A Message Partial Content-Type header field that appears in each of the fragments contains information that helps reassemble the fragments into one message. See also *fragmentation*.

**denial of service attack** A situation where an individual intentionally or inadvertently overwhelms your mail server by flooding it with messages. Your server's throughput could be significantly impacted or the server itself could become overloaded and nonfunctional.

**Departmental Edition** Also referred to as Sun Internet Mail Server-Departmental Edition, is the version of SIMS intended for use by local departmental environments. This package performs its own routing and delivery within a local office or department, but hands off interdepartmental mail to backbone or enterprise server.

**dereferencing an alias** Specifying, in a bind or search operation, that a directory translate an alias DN to the DN of an actual entry.

**destination channel** The last element of a host / domain rewrite rule, in whose queue a message should be placed in for delivery.

**directory context** The point in the directory tree at which a search is begun.

**directory entry** A set of directory attributes and their values identified by its distinguished name. Each entry contains an object class attribute that specifies the kind of object the entry describes and defines the set of attributes it contains.

**Directory Information Tree** Directory Information Tree is the tree-like hierarchical structure in which directory entries are organized.

**directory schema** The set of rules that defines the data that can be stored in the directory.

**Directory Service** A logically centralized repository of information. The component in SIMS that stores user, distribution list, and configuration data.

**dirsync option file** A file used to set options for the dirsync program which cannot be set through the command line.

**disconnected state** The mail client connects to the server, makes a cache copy of selected messages, then disconnects from the server.

**distinguished name** The sequence of attributes and values of an entry within the directory information tree.

**distribution list** A list of email addresses (users) that can be sent a message by specifying one email address. See also *expansion*, *member*, *moderator*, *owner*, and *alias*.

**DIT** Directory information tree. A hierarchical structure in which directory data or information (names, email addresses, and so on) is stored.

**DNS** Domain Name Service. The naming facilities of the Internet.

**domain** In the Internet, a part of a naming hierarchy. Syntactically, an Internet domain name consists of a sequence of names (labels) separated by periods (dots), for example, `tundra.mpk.ca.us`. In OSI, `domain` is generally used as an administrative partition of a complex distributed system, as in MHS Private Management Domain (PRMD), and Directory Management Domain (DMD).

**domain rewriting rules** See also *rewrite rules*.

**domain template** The part of a rewrite rule that defines how the host/domain portion of an address is rewritten. It can include either a full static host/domain address or a single field substitution string, or both.

**dotted decimal notation** The syntactic representation for a 32-bit integer that consists of four 8-bit numbers written in base 10 with periods (dots) separating them. Used to represent IP addresses in the Internet as in 192.67.67.20.

**DSA** Directory System Agent. The software that provides the X.500 Directory Service for a portion of the directory information base. Generally, each DSA is responsible for the directory information for a single organization or organizational unit.

**DUA** Directory User Agent. The software that accesses the X.500 Directory Service on behalf of the directory user. The directory user may be a person or another software element.

**EMAPI** Extended MAPI Service Provider. Transparently turns Microsoft Exchange client into an Internet standard IMAP/LDAP client. See also *IMAP*, *LDAP*.

**encapsulation** The technique used by layered protocols in which a layer adds header information to the protocol data unit (PDU) from the layer above.

**encryption** Scrambling the contents of a message so that its contents cannot be read without the encryption, or code key.

**end system** An OSI system which contains application processes capable of communicating through all seven layers of OSI protocols. Equivalent to Internet host.

**Enterprise Edition** Also referred to as Sun Internet Mail Server-Enterprise Edition, provides full-featured messaging server for large user communities.

**entity** OSI terminology for a layer protocol machine. An entity within a layer performs the functions of the layer within a single computer system, accessing the layer entity below and providing services to the layer entity above at local service access points.

**entries** User, group, or organizational data used to configure message accounts.

**envelope** The part of an Internet mail message that contains the delivery information. The envelope contains the originator and recipient information associated with a message.

**ESMTP** Extended Simple Mail Transfer Protocol. An Internet message transport.

**expander** Part of an electronic mail delivery system which allows a message to be delivered to a list of addressees. Mail exploders are used to implement mailing lists. Users send messages to a single address (e.g., hacks@somehost.edu) and the mail exploder takes care of delivery to the individual mailboxes in the list.

**expansion** This term applies to the Internet Mail Transport Agent (IMTA) processing of distribution lists. The act of converting a message addressed to a distribution list into enough copies for each distribution list member.

**expunge** The act of deleting a message then removing the deleted message via a mail client.

**external channel** An interface between the IMTA and either another SIMS component or another component outside the SIMS email system.

**File System** This can be either safe or unsafe. A safe file system performs logging such that if a system crashes it is possible to rollback the data to a pre-crash state and restore all data. An example of a safe file system is Veritas File System, VxFS. An unsafe file system does not perform logging. If the system crashes, the state cannot be recreated and some data may be lost. You must also perform imcheck before activating message access to these files.

**firewall** Router or mail-level hosts that are equipped with special codes to control access between the Internet and the internal network.

**folder** Named place where mail is stored. Also called a *mailbox*. Inbox is a folder that stores new mail. Users can also have folders where mail can be stored. A folder can contain other folders in a hierarchical tree. Folders owned by a user are called *private folders*. See also *shared folders.*

**Folder Check** A utility which checks the accessibility of messages and folders and verifies links. This utility is used as part of the regular maintenance of SIMS.

**forward address mapping** Message envelopes, TO:address, are processed to a mapping table. The result of the mapping is tested. If necessary, the exact form of the envelope is exchanged for another which can then be processed by a different, and perhaps non-compliant RFC 822, mail system.

**fragmentation** The Multiple Internet Extensions (MIME) feature that allows the breaking up of a large message into smaller messages. See also *defragmentation*.

**FTAM** File Transfer, Access, and Management. The OSI remote file service and protocol. See also *FTP*.

**FTP** File Transfer Protocol. The Internet protocol (and program) used to transfer files between hosts. See also *FTAM*.

**full static host/domain address** The portion of a host/domain address elements set off by decimals as part of the domain template. See also *domain template*.

**gateway** The terms *gateway* and *application gateway* refer to systems that do translation from one native format to another. Examples include X.400 to/from RFC 822 electronic mail gateways. A machine that connects two or more electronic mail systems (especially dissimilar mail systems on two different networks) and transfers messages between them. Sometimes the mapping and translation can be complex, and it generally requires a store-and-forward scheme whereby the message is received from one system completely before it is transmitted to the next system after suitable translations.

**global log manager** A utility that handles log information from each Sun Internet Mail Server component.

**GOSIP** Government OSI Profile. A U.S. Government procurement specification for OSI protocols.

**group entry** See distribution lists.

**group folders** Contain folders for shared and group folders.

**header** The part of an Internet mail message that is composed of a field name followed by a colon and then a value. Headers include delivery information, summaries of contents, tracing, and MIME information.

**HTML** Hypertext Markup Language.

**HTTP** Hypertext Transfer Protocol.

**IAB** Internet Activities Board. The technical body that oversees the development of the Internet suite of protocols (commonly referred to as “TCP/IP”). It has two task forces (the IRTF and the IETF) each charged with investigating a particular area.

**IESG Internet Engineering Steering Group** The executive committee of the IETF.

**IETF Internet Engineering Task Force** One of the task forces of the IAB. The IETF is responsible for solving short-term engineering needs of the Internet. It has over 40 Working Groups.

**IMAP4** Internet Message Access Protocol. IMAP4 provides advanced disconnected mode client access.

**IMS log** Sun Message Store log files.

**IMTA** Internet Message Transfer Agent. IMTA routes, transports, and delivers Internet Mail messages within the email system.

**intermediate system** An OSI system which is not an end system, but which serves instead to relay communications between end systems.

**internal channel** An interface between internal modules of the IMTA. Internal channels include the reprocessing, conversion, and defragmentation channels. These channels are not configurable.

**Internet address** A 32-bit address assigned to hosts using TCP/IP. See also *dotted decimal notation*.

**IP Internet Protocol** The network layer protocol for the Internet protocol suite.

**ISDN** Integrated Services Digital Network. An emerging technology which is beginning to be offered by the telephone carriers of the world. ISDN combines voice and digital network services in a single medium making it possible to offer customers digital data services as well as voice connections through a single "wire." The standards that define ISDN are specified by ITU-T.

**ISO** International Organization for Standardization. See also *OSI*.

**ITU** International Telecommunications Union. International Consultative Committee for Telegraphy and Telephony. A unit of the International Telecommunications Union (ITU) of the United Nations. An organization with representatives from the PTTs of the world. CCITT produces technical standards, known as "Recommendations," for all internationally controlled aspects of analog and digital communications. See also *X Recommendations*.

**Java** A programming language developed by Sun Microsystems.

**JMAPI** Java Management Application Programming Interface. JMAPI is a collection of programming language classes that enable a diverse set of autonomous applications to be brought together under a common look, feel, and behavior everywhere they run.

**job controller** The SIMS component which schedules message delivery or message submission tasks between various SIMS components. Job controller also controls channel queues and determines the order of processing. Requests are processed in the order in which they are received by the system.

**Kerberos** Client-to-server security package produced by MIT.

**key ring** A collection of public and private security keys.

**knowledge information** Part of the directory service infrastructure information. The directory server uses knowledge information to pass requests for information to other servers.

**LDAP** Lightweight Directory Access Protocol. LDAP is a protocol used for the storage, retrieval, and distribution of information, including user profiles, distribution lists, and configuration data.

**LDAP filter** A way of specifying a set of entries, based on the presence of a particular attribute or attribute value.

**LDBM** A type of backend that stores directory information that provides access to information stored in a database. See also *backend*.

**LDIF** LDAP Data Interchange Format. A data format used to represent LDAP entries in text form.

**Legacy Mail Services** Provides batch-mode connectivity to legacy proprietary message transfer systems.

**little-endian** A format for storage or transmission of binary data in which the least significant byte (bit) comes first. See also *big-endian*.

**local channel** A channel that allows you to determine delivery options of local users and delivers mail to Solaris Operating Environment mailboxes.

**lookup** Same as a search, using the specified parameters for sorting data.

**Mailbox** A place where messages are stored and viewed. See *folder*.

**Mailtool** A `/var/mail` client application that runs under the OpenWindows V3 desktop environment.

**man page** UNIX Reference manual pages.

**managed object** A collection of configurable attributes, for example, a collection of attributes for the directory service.

**mapping tables** Two column tables which transform, map, an input string into an output string.

**master directory server** The directory server that contains the data that will be replicated.

**master message catalog** Contains message catalogs for the SIMS components.

**master program** A channel program that initiates a message transfer to another interface on its own.

**member** A user or group who receives a copy of an email addressed to a distribution list. See also *distribution list*, *expansion*, *moderator*, and *owner*.

**Message Access and Store** These are the SIMS components which store user messages and allow for retrieval and processing of messages.

**Message Access Services** Consists of protocol servers, software drivers, and libraries which support client access to the message store.

**message catalogs** The log messages, command line responses, and graphical user interface screen text contained in the SIMS components.

**message databases** Contain messages and attachments.

**message hash** Contain hashing files.

**message indices** Contain message index files.

**message submission** The client Mail User Agent (MUA) transfers a message to the mail server and requests delivery.

**MHS Message Handling System** The system of message user agents, message transfer agents, message stores, and access units which together provide OSI electronic mail. MHS is specified in the ITU-T X.400 series of Recommendations.

**MIB** Management Information Base. A collection of objects that can be accessed via a network management protocol. See also *SMI*.

**MIME** Multipurpose Internet Mail Extensions. A format for defining email message content.

**moderator** If the moderator feature is enabled, a message addressed to a distribution list is initially sent to the moderator only. The moderator can take one of the following actions: forward the message to the distribution list, edit the message and then forward it to the distribution list, or not forward the message to the distribution list. See also *distribution list*, *expansion*, *member*, and *owner*.

**MTA** Message Transfer Agent. An OSI application process used to store and forward messages in the X.400 Message Handling System. Equivalent to Internet mail agent.

**MUA** Mail User Agent. The client applications invoked by end users to read, submit, and organize their electronic mail.

**multicasting** A process by which the directory server broadcasts an information request to all DSAs it knows about. Any DSA that can process the request does so and returns the result to the first DSA. See also *knowledge information*.

**multithreaded** The ability to handle multiple, simultaneous sessions in a single process.

**name resolution** The process of mapping a name into the corresponding address. See also *DNS*.

**naming attribute** The final attribute in a directory information tree distinguished name. See also *relative distinguished name.*

**naming context** A specific subtree of a directory information tree that is identified by its DN. In SIMS, specific types of directory information are stored in naming contexts. For example, a naming context which stores all entries for marketing employees in the XYZ Corporation at the Boston office might be called ou=mktg, ou=Boston, o=XYZ, c=US.

**network address** See also *Internet address* or *OSI Network Address*.

**Network Layer** The OSI layer that is responsible for routing, switching, and subnetwork access across the entire OSI environment.

**NFS®** Network File System. A distributed file system developed by Sun Microsystems which allows a set of computers to cooperatively access each other's files in a transparent manner.

**NIC** Network Information Center. Originally there was only one, located at SRI International and tasked to serve the ARPANET (and later DDN) community. Today, there are many NICs, operated by local, regional, and national networks all over the world. Such centers provide user assistance, document service, training, and much more.

**NIST** National Institute of Standards and Technology. (Formerly NBS, National Bureau of Standards). See also *OIW*.

**NMS Network Management Station** The system responsible for managing a (portion of a) network. The NMS talks to network management agents, which reside in the managed nodes, via a network management protocol. See also *agent*.

**nondelivery report** During message transmission, if the IMTA does not find a match between the address pattern and a rewrite rule, the IMTA sends a nondelivery report back to the sender with the original message, then deletes its copy of the message.

**non-specific subordinate reference** A naming context that is lower in the directory tree but not a child of the naming context held by your directory server. See also *knowledge information*.

**Notary Messages** Text messages sent by the MTA to an email sender indicating delivery or non-delivery status of a sent message.

**object class** A template specifying the kind of object the entry describes and the set of attributes it contains. For example, SIMS specifies an `emailPerson` object class which has attributes such as `commonname`, `mail` (email address), `mailHost`, and `mailQuota`.

**off-line state** The mail client fetches messages from a server system to a client system, which may be a desktop or portable system and may delete them from the server. The mail client downloads the messages where they can be viewed and answered.

**on-line state** A state in which messages remain on the server and are remotely responded to by the mail client.

**ONC™ Open Network Computing** A distributed applications architecture promoted and controlled by a consortium led by Sun Microsystems.

**option files** IMTA option files contain global parameters used to override default values of parameters which apply to IMTA as a whole, such as sizes for various tables into which various configuration and alias files are read.

**organizational unit** A layer in the directory information tree.

**OSI Open Systems Interconnection** An international standardization program to facilitate communications among computers from different manufacturers. See also *ISO*.

**OSI Network Address** The address, consisting of up to 20 octets, used to locate an OSI Transport entity. The address is formatted into an Initial Domain Part which is standardized for each of several addressing domains, and a Domain Specific Part which is the responsibility of the addressing authority for that domain.

**owner** An individual who is responsible for a distribution list. An owner can add or delete distribution list members. See also *distribution list*, *expansion*, *member*, and *moderator*.

**PGP** Pretty Good Privacy. PGP provides client-to-client security, encrypting or scrambling the text of a message so that only the receiving message server can decrypt or unscramble the text.

**permanent failure** An error condition that occurs during message handling. When this occurs, the message store deletes its copy of an email message. The Internet Message Transport Agent (IMTA) bounces the message back to the sender and deletes its copy of the message.

**pipe channel** A channel which performs delivery of messages via a per-user-site-supplied program. These programs must be registered in SIMS by the system administrator, and thus do not pose a security risk.

**POP** Post Office Protocol. POP provides remote access support for older mail clients.

**populating the directory** Entering information for users and distribution lists to the SIMS directory service.

**PPP** Point-to-Point Protocol. The successor to SLIP, PPP provides router-to-router and host-to-network connections over both synchronous and asynchronous circuits. See also *SLIP*.

**PRMD** Private Management Domain. An X.400 Message Handling System private organization mail system. Example NASAmail. See also *ADMD*.

**protocol** A formal description of messages to be exchanged and rules to be followed for two or more systems to exchange information.

**proxy** The mechanism whereby one system "fronts for" another system in responding to protocol requests. Proxy systems are used in network management to avoid having to implement full protocol stacks in simple devices, such as modems.

**public key encryption** A cryptographic method that uses a two-part key (code) that is made up of public and private components. To encrypt messages, the published public keys of the recipients are used. To decrypt the messages, the recipients use their unpublished private keys known only to them.

**purge** The process of removing messages that are no longer referenced in user and group folders and returning the space to the Sun Message Store file system. See also *backup* and *restore*.

**PTT** Post Telephone Telegraph.

**Qualcomm Eudora** A mail client produced by Qualcomm Corporation that supports MIME, POP3, and MAIL protocols.

**quota** See user quota.

**RBOC** Regional Bell Operating Company. See also *BOC*.

**referral** A process by which the directory server returns an information request to the client that submitted it, with information about the Directory Service Agent (DSA) that the client should contact with the request. See also *knowledge information*.

**relaying** A message is passed from one mail server to another mail server.

**relative distinguished name** The final attribute and its value in the attribute and value sequence of the distinguished name. See also *distinguished name*.

**replica directory server** The directory that will receive a copy of all or part of the data.

**reprocessing channel** Performs deferred processing. The reprocessing channel is the intersection of all other channel programs. It performs only the operations that are shared with other channels.

**restore** The process of restoring the contents of folders from a backup device to the Sun Message Store. See also *backup* and *purge*.

**reverse address mapping** Addresses are processed to a mapping table, with a reversal database, generally substituting a generic address, possibly on a central machine, for an address on a remote or transitory system.

**rewrite rules** Also known as domain rewriting rules. A tool that the Internet Mail Transport Agent (IMTA) uses to route messages to the correct host for delivery. Rewrite rules perform the following functions: (1) extract the host/domain specification from an address of an incoming message, (2) match the host/domain specification with a rewrite rule pattern, (3) rewrite the host/domain specification based on the domain template, and (4) decide which IMTA channel queue the message should be placed in.

**RFC** Request For Comments. The document series, begun in 1969, describes the Internet suite of protocols and related experiments. Not all (in fact very few) RFCs describe Internet standards, but all Internet standards are published as RFCs. See http://www.imc.org/rfcs.html.

**RMI** Remote Method Invocation. A Java-based programming language that enables the Administration Console and the server to communicate.

**Roam** The Sun Internet Mail Client. Roam is a disconnected mode mail user agent (MUA) that supports the low-bandwidth IMAP protocol extensions of the Sun Internet Mail Server, IMAP.

**root entry** The first entry of the directory information tree (DIT) hierarchy.

**router** A system responsible for determining which of several paths network traffic will follow. It uses a routing protocol to gain information about the network, and algorithms to choose the best route based on several criteria known as “routing metrics.” In OSI terminology, a router is a Network Layer intermediate system. See also *gateway*.

**routability scope** Specifications which enable the IMTA to send messages by the most direct route, either to a specific user's folder, a group of folders, or to a mail host.

**routing** In an email system, the act of delivering a message based on addressing information extracted from the body of the message. The Internet Message Transfer Agent (IMTA) is the component responsible for routing messages.

**RTSE** Reliable Transfer Service Element. A lightweight OSI application service used above X.25 networks to handshake application PDUs across the Session Service and TP0. Not needed with TP4, and not recommended for use in the U.S. except when talking to X.400 ADMDs.

**S/Key** Client-to-server security package produced by Bell Labs

**SASL** Server-to-server security.

**schema** A set of rules which sets the parameters of the data stored in a directory. It defines the type of entries, their structure and their syntax.

**shared folder** *A* mailbox that can be viewed by members of a *distribution list*. Shared folders have an *owner* who can add or delete members to the group and can delete messages from a the shared folder. The can also have a moderator who can edit, block, or forward incoming messages.

**shell backend** A type of backend that stores directory information. This type provides access to information stored in any format, using shell scripts.

**single field substitution string** Part of the domain template that dynamically rewrites the specified address token of the host/domain address. See also *domain template*.

**SIMS Host** Name of host on which Sun Message Store is installed.

**SIMS initialization duration in days** Number of days to initialize the Sun Message Store

**SIMS Owner** Person in control of Sun Message Store files.

**SKIP** Simple Key management for IP. A security system that encrypts or scrambles the text of a message so only the receiving mail client or message server can decrypt or unscramble the text.

**SLAPD** A daemon that operates that accesses the database files that hold the directory information, and communicates with directory clients using the LDAP protocol.

**slave program** A channel program that accepts transfers initiated by another interface.

**SLIP** Serial Line IP. An Internet protocol used to run IP over serial lines such as telephone circuits or RS-232 cables interconnecting two systems. SLIP is now being replaced by PPP. See also *PPP*.

**SLURPD** A replication daemon that runs on demand or schedule and ensures that any directory information changes are propagated to systems that hold replicas of that information.

**smart host** The Internet Message Transfer Agent (IMTA) in a particular domain to which other IMTAs acting as routers forward messages if they do not recognize the recipients.

**SMI** Structure of Management Information. The rules used to define the objects that can be accessed via a network management protocol. See also *MIB*.

**SMTP** Simple Mail Transfer Protocol. The Internet electronic mail protocol. Defined in RFC 821, with associated message format descriptions in RFC 822.

**SMPT Dispatcher** A multithreaded connection dispatching agent which allows multiple multithreaded servers to share responsibility for a given service, thus allowing several multithreaded SMTP servers to run concurrently and handle one or more active connections.

**SMTP intranet or internet channel** A channel dedicated to relaying messages between the IMTA and a group of SMTP hosts within, or outside of, your mail network.

**SMTP router channel** SMTP channel that handles messages between the IMTA and firewall host.

**SNMP** Simple Network Management Protocol. The network management protocol of choice for TCP/IP-based internets.

**subordinate reference** The naming context that is a child of the naming context held by your directory server. See also *knowledge information*.

**Sun Directory Services** SIMS component which provides access and maintenance of user profiles, distribution lists, and other system resources.

**Sun Internet Mail Client** The client end of the SIMS solution that supports online, offline, and disconnected states. See also *Roam*.

**Sun Internet Mail Server** An enterprise-wide, open-standards based, scalable electronic message-handling system.

**Sun Messaging Connectivity Services** This component of SIMS provides batch-mode connectivity to proprietary message transfer systems, including: “LAN mail” systems, Lotus cc:Mail, Microsoft Mail and mainframe-based IBM OfficeVision.

**Sun Message Store** The server from which mail clients retrieve and submit messages.

**SQL** Structured Query Language. The international standard language for defining and accessing relational databases.

**Sun Messaging Connectivity Services** This component of SIMS provides batch-mode connectivity to proprietary message transfer systems, including: “LAN mail” systems, Lotus cc:Mail, Microsoft Mail and mainframe-based IBM OfficeVision.

**SSL** Secure Sockets Layer is an open, non-proprietary security protocol.

**synchronization** The update of data by a master directory server to a replica directory server.

**table lookup** With a table consisting of two columns of data, an input string is compared with the data within the table and transformed to an output string.

**tailor file** An option file used to set the location of various IMTA components.

**TCP** Transmission Control Protocol. The major transport protocol in the Internet suite of protocols providing reliable, connection-oriented, full-duplex streams. Uses IP for delivery. See also *TP4*.

**transient failure** An error condition that occurs during message handling. The remote Internet Message Transport Agent (IMTA) is unable to handle the message when it's delivered, but may be able to later. The local IMTA returns the message to the channel queue and schedules it for retransmission at a later time.

**transport protocols** Provides the means to transfer messages between message stores.

**UA** User Agent. An OSI application process that represents a human user or organization in the X.400 Message Handling System. Creates, submits, and takes delivery of messages on the user's behalf.

**user entry or user profile** Fields that describe information about each user, required and optional, examples are: distinguished name, full name, title, telephone number, pager number, login name, password, home directory, etc.

**user folders** Contain user's email folders.

**user quota** The finite amount of space, configured by the system administrator, allocated to each user for incoming or stored messages.

**user redirection** The remote Internet Message Transport Agent (IMTA) cannot accept mail for the recipient, but can reroute the mail to a mail server that can accept it.

**upper reference** Indicates the directory server that holds the naming context above your directory server's naming context in the directory information tree (DIT).

**UUCP** UNIX to UNIX Copy Program. A protocol used for communication between consenting UNIX systems.

**UUCP Channel** Unix to Unix Copy System is provided only in the SIMS Enterprise Edition. It is a asynchronous terminal line-based system used to provide support for file transfer and remote execution between different computer systems.

**`/var/mail`** The UNIX version 7 "`From`" delimited mailbox as implemented in the Solaris operating system.

**workgroup** Local workgroup environment, where the server performs its own routing and delivery within a local office or workgroup. Interdepartmental mail is routed to a backbone server. See also *backbone*.

**X.400** A message handling system standard.

**XFN** X-Open Federated Naming technology.

**X Recommendations** The CCITT documents that describe data communication network standards. Well-known ones include X.25 Packet Switching standard, X.400 Message Handling System, and X.500 Directory Services.

# Index

## SYMBOLS



## A

## B



## C

## D

## E



## F



## G

## H



## I

## J



## K



## L

## M

## N



## O

## P



## Q



## R

## S

## T

## U



## V



## W