Oracle® Communications Delegated Administrator

System Administrator's Guide

Release 7.0

E54914-02

February 2016

ORACLE®

Oracle Communications Delegated Administrator System Administrator's Guide, Release 7.0

E54914-02

Contents

7 Consolidating ACIs for Directory Server Performance

8 Customizing Delegated Administrator

9 Troubleshooting Delegated Administrator

10 Support For Additional Values of mailuserstatus

A Service Package Details

E Delegated Administrator Reference

Preface

This guide explains how to configure and administer Oracle Communications Delegated Administrator. Delegated Administrator provisions users, groups, organizations/domains, and resources in an LDAP directory used by Communications Suite applications such as Oracle Communications Messaging Server, Oracle Communications Calendar Server, and Oracle Communications Instant Messaging Server.

Audience

This document is intended for system administrators whose responsibility includes Delegated Administrator. This guide assumes you are familiar with the following topics:

- Messaging Server and Calendar Server protocols
- Oracle Directory Server Enterprise Edition and LDAP
- System administration and networking
- General deployment architecture

Related Documents

For more information, see the following documents in the Delegated Administrator documentation set:

- *Delegated Administrator Installation and Configuration Guide*: Provides instructions for installing and configuring Delegated Administrator.
- *Delegated Administrator Release Notes*: Describes the features, fixes, known issues, troubleshooting tips, and required third-party products and licensing.
- *Delegated Administrator Security Guide*: Provides guidelines and recommendations for setting up Delegated Administrator in a secure configuration.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit

http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Document Revision History

The following table lists the revision history for this guide:

Version	Date	Description
E54914-02	February 2016	Minor formatting and text changes.
E54914-01	August 2014	Initial release.

1

Delegated Administrator Overview

This chapter provides an overview of Oracle Communications Delegated Administrator.

Introduction to Delegated Administrator

The Delegated Administrator utility and console let you provision users, groups, domains, and resources in a LDAP directory used by Communications Suite applications such as Oracle Communications Messaging Server, Oracle Communications Calendar Server, Oracle Communications Instant Messaging Server, and Oracle Communications Contacts Server.

Delegated Administrator provisions the directory to support Messaging Server, Calendar Server, Instant Messaging, and Contacts Server.

With Delegated Administrator, you can distribute provisioning tasks to lower-level administrators who have the authority to manage specified organizations in the LDAP directory. The power to delegate user administration offers the following advantages:

- Allows you to create organizations in the directory structure that can be managed and provisioned as distinct (or unique) units. These organizations can contain users belonging to customer businesses, corporate departments, or other groups.
- Distributes among many administrators the potentially time-consuming responsibility for provisioning a large directory. Tens or hundreds of administrators can manage organizations within a directory that may include thousands or millions of users.

Delegated Administrator provides two interfaces for provisioning users and organizations in the directory.

The Delegated Administrator utility is a set of command-line tools for provisioning Messaging Server, Calendar Server, Instant Messaging, and Contacts Server organizations, users, groups, and Calendar resources.

> **Note:** The Delegated Administrator utility does not offer commands for creating the Service Provider roles and organizations described in this book. To create and manage these roles and organizations, you must use the Delegated Administrator console.

You invoke the utility with the **commadmin** command.

For information about the syntax and options available with the **commadmin** command, see "Delegated Administrator Reference".

The Delegated Administrator console is a graphical user interface (GUI) for provisioning Messaging Server, Calendar Server, Instant Messaging, and Contacts Server organizations, users, groups, and Calendar resources.

For information on how to use the console, see the Delegated Administrator console online help.

Delegated Administrator LDAP Attributes

Delegated Administrator enables you to provision users by modifying the LDAP directory. You do not need to modify the directory directly. However, it can be useful to understand attributes added to user entries and higher-level nodes in the directory.

For information about the LDAP schema object classes and attributes that support Delegated Administrator, see the discussion about Delegated Administrator LDAP object classes and attributes in *Communications Suite Schema Reference*.

Communications Suite Schema Reference also defines the object classes and attributes that support the other Communications Suite components: Messaging Server, Calendar Server, Instant Messaging, Contacts Server, Address Book, and Communications Express.

Note: When using the Delegated Administrator utility or the Delegated Administrator console, you should also run the Referential Integrity plug-in because that plug-in ensures that when entries are removed, all attributes that contain their DN are also removed. See the Maintaining Referential Integrity documentation in *Oracle Fusion Middleware Administration Guide for Oracle Directory Server Enterprise Edition 11* for more information.

Configuration Choices for LDAP Directory Access, Schema, and Access Manager

This section discusses the configuration choices you have for configuring direct LDAP access, schema, and Access Manager.

Direct LDAP Access to the Directory

By default, Delegated Administrator accesses the directory through direct LDAP calls. By using direct LDAP access, Delegated Administrator allows the following configuration choices:

- You can provision objects in a Schema 1 or Schema 2 directory.
- You can use Access Manager (Realm mode) with the Communications Suite products (including Delegated Administrator).
- You can run Delegated Administrator, and any other Communications Suites product, without installing or using Access Manager.

To take advantage of these choices, you must:

1. Choose whether to use Schema 1 or Schema 2 when you run the Directory Server Preparation Tool, **comm_dssetup.pl**.
2. Select **Direct LDAP access to the directory (DL)** when you run the Delegated Administrator configuration program, **config-commda**.

Directory Access Through Access Manager (Legacy Mode)

You can configure Delegated Administrator to access the directory using Access Manager in Legacy mode.

This access method is intended for users of previous releases of Delegated Administrator who are upgrading to the current release and want to continue to use Access Manager in Legacy mode. Support for Access Manager 7.x and Sun OpenSSO 8.x has been deprecated. Delegated Administrator optionally uses Access Manager or Sun OpenSSO for single sign-on, authentication, and policy options.

You configure this access method by selecting **Access Manager LDAP access (AM)** when you run the Delegated Administrator configuration program, **config-commda**.

In the Access Manager (Legacy Mode) access method:

- Access Manager (Legacy mode) must be installed. If you choose this access method, Delegated Administrator cannot be configured or run without Access Manager.
- Access Manager (Realm mode) cannot be installed. With this access method, Delegated Administrator is not compatible with Access Manager in Realm mode.
- The Delegated Administrator server must use the same Web container as Access Manager. The Delegated Administrator configuration program asks for Web container information after it asks for the Access Manager base directory.

LDAP Directory Access, Schema, and Access Manager Configuration Summary

Table 1–1 shows the configurations permitted by Delegated Administrator for LDAP directory access, schema choice, and Access Manager.

Table 1–1 LDAP Directory Access, Schema, and Access Manager Configurations

LDAP Directory Access	Schema	Access Manager Choice
Direct LDAP access	Schema 2	Access Manager (Realm mode)
Direct LDAP access	Schema 2	Access Manager not installed
Direct LDAP access	Schema 1	Access Manager (Realm mode)
Direct LDAP access	Schema 1	Access Manager not installed
Access Manager (Legacy mode)	Schema 2	Access Manager (Legacy mode)

Scenarios for Provisioning Users

Depending on your business needs you can create a simple directory structure managed by a single administrator, or a multi-tier directory hierarchy in which provisioning and management tasks are delegated to lower-level administrators.

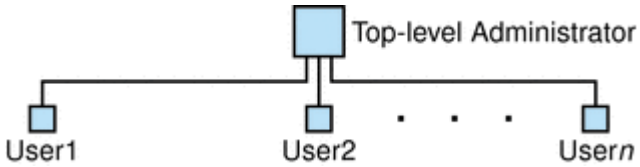
This section summarizes three scenarios of increasing complexity. It then describes the administrator roles and directory structures Delegated Administrator provides to support the requirements of these scenarios.

One-Tier Hierarchy

In this scenario, a company or organization might support hundreds or thousands of employees or users. All users are grouped in a single organization. A single administrator role views and manages the entire group. There is no delegation of administrative tasks.

Figure 1–1 shows an example of the administrator role in a single-organization, one-tier hierarchy.

Figure 1–1 Administrator Role in a One-Tier Hierarchy



In this one-tier hierarchy, the administrator is called the Top-Level Administrator (TLA).

In the example shown in Figure 1–1, the TLA directly manages and provisions the users (User1, User2, up to User*n*).

If you have one organization in your directory, the TLA is the only administrator you need.

For more information, see:

- Directory Structure Supporting a One-Tier Hierarchy
- Top-Level Administrator

Two-Tier Hierarchy

In this scenario, a large company such as an Internet Service Provider (ISP) provides services to businesses. Each business has its own unique domain, which may contain thousands or tens of thousands of users.

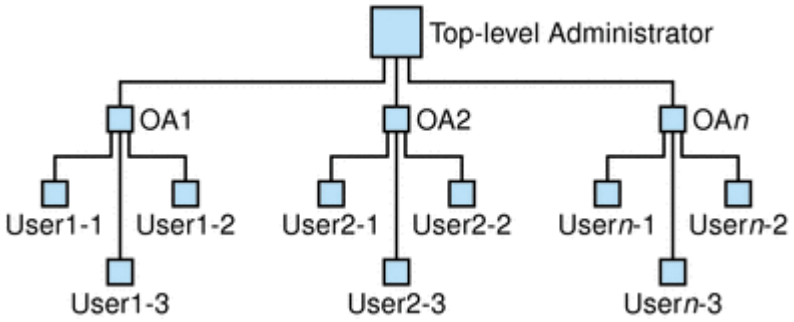
Instead of relying on a single Top-Level Administrator (TLA) to manage and provision all the domains, this scenario supports the delegation of tasks to lower-level administrators.

In a two-tier hierarchy, the directory contains multiple organizations. A separate organization is created for each hosted domain.

Each organization is assigned to an Organization Administrator (OA). The OA is responsible for the users in that organization. An OA cannot view or modify directory information outside the OA's own organization.

Figure 1–2 shows an example of the administrator roles in a two-tier hierarchy.

Figure 1–2 Administrator Roles in a Two-Tier Hierarchy



In the example shown in Figure 1–2, the TLA creates and manages OA1, OA2, up to OA*n*. Each OA manages the users in one organization.

If you need multiple organizations in your directory, you should create the OAs to administer the organizations and their users.

For more information, see:

- Directory Structure Supporting a Two-Tier Hierarchy
- Top-Level Administrator
- Organization Administrator Role

Three-Tier Hierarchy

In this scenario, a company, such as an ISP, offers services to hundreds or thousands of small businesses, each of which requires its own organization.

The ISP may support millions of end-users requiring mail services. Moreover, the ISP may work with third-party resellers who manage the end-user businesses.

Each day, dozens of organizations might have to be added to the directory.

In a two-tier hierarchy, the TLA would have to create all these organizations.

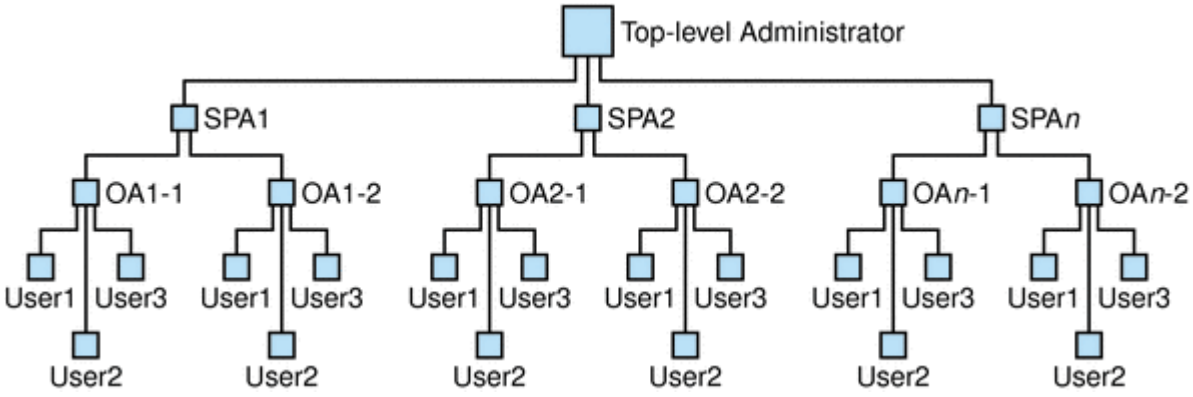
In a three-tier hierarchy, management tasks are delegated to a second level of administrators. This second level of delegation can ease the management of a large customer base supported by a large LDAP directory.

To support this hierarchy, Delegated Administrator introduces a role, the Service Provider Administrator (SPA).

The SPA's scope of authority lies between that of the Top-Level Administrator (TLA) and the Organization Administrator (OA).

Figure 1–3 shows an example of the administrator roles in a three-tier hierarchy.

Figure 1–3 Administrator Roles in a Three-Tier Hierarchy



In a three-tier hierarchy, the TLA delegates administrative authority to Service Provider Administrators (SPAs). The SPAs can create subordinate organizations for customers and assign Organization Administrators (OAs) to manage users in those organizations.

If you need multiple organizations that are themselves divided into subgroups or organizations, you can use a three-tier hierarchy that implements the TLA, SPA, and OA roles.

See "Service Provider Administrator and Service Provider Organizations" for information about the SPA role.

Administrator Roles and the Directory Hierarchy

This section shows sample Directory Information Trees that implement one- and two-tier hierarchies. It then describes the tasks that can be performed by the Top-Level Administrator and Organization Administrator.

Directory Structure Supporting a One-Tier Hierarchy

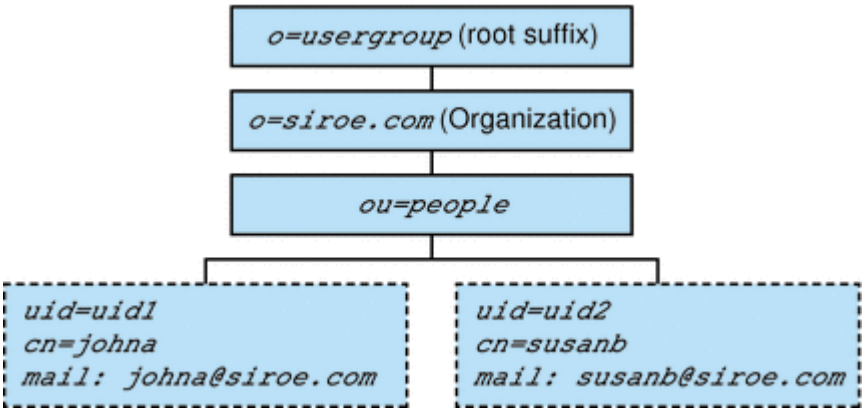
When you configure Delegated Administrator by running the configuration program, **config-commda**, you create a Top-Level Administrator (TLA) and a default organization.

One-Tier Hierarchy: Default Organization Under the Root Suffix

By default, the configuration program places the default organization under the root suffix.

Figure 1–4 shows a sample Directory Information Tree organized in a one-tier hierarchy (default configuration).

Figure 1–4 One-Tier Hierarchy: Sample Directory Information Tree (default)



One-Tier Hierarchy: Default Organization at the Root Suffix

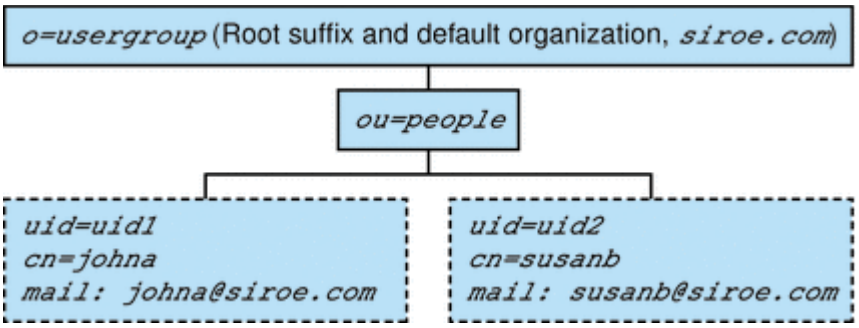
When you run the configuration program, **config-commda**, you can choose to create the default organization at the root suffix instead of under it. For configuration details, see *Delegated Administrator Installation and Configuration Guide*.

In this situation, the Directory Information Tree will look similar to the one shown in Figure 1–5.

However, if you create the default organization at the root suffix, this configuration of the LDAP directory cannot support multiple hosted domains. To support hosted domains, the default organization must be under the root suffix.

Figure 1–5 shows a sample one-tier hierarchy in which the default organization is created at the root suffix.

Figure 1–5 One-Tier Hierarchy: Default Organization at Root Suffix



Top-Level Administrator

The TLA has the authority to perform the following tasks:

- Create, delete, and modify organizations.

  In the example shown in Figure 1–6, the TLA can modify or delete **siroe.com** or **sesta.com** and can create additional organizations.

  In this example, the two organizations are also unique (hosted) domains.

- Create, delete, and modify users.
- Create, delete, and modify groups.
- Create, delete, and modify Calendar resources.
- Assign OA roles to users. For example, the TLA could assign an OA role to the user **johna** in the **siroe.com** organization.

  The TLA also can remove the OA role from a user.

- Assign TLA roles to other users. The TLA also can remove the TLA role from a user.

  The TLA can perform the preceding tasks by using the Delegated Administrator console or by executing Delegated Administrator utility (**commadmin**) commands. For a description of the **commadmin** commands, see "Delegated Administrator Reference".

- Assign service packages to organizations.

  See "Service Packages" for information on service packages.

  The TLA can assign specified types of service packages to an organization and determine the maximum number of each package that can be used in that organization.

  For example, the TLA could assign the following service packages:

  – In the **siroe.com** organization:

    1,000 gold packages

    500 platinum packages

  – In the **sesta.com** organization:

    2,000 silver packages

    1,500 gold packages

    100 platinum packages

Service Provider Administrator Role

The SPA can perform the following tasks:

- Create, delete, and modify shared and full organizations in the provider organization in which the SPA has administrative authority.
  - Modify or delete the DEF, HIJ, and SESTA organizations.
  - Create additional organizations under the VIS provider organization.
- Create, delete, and modify users in any organization under the provider organization.
- Create, delete, and modify groups in any organization under the provider organization.
- Create, delete, and modify Calendar resources in any organization under the provider organization.
- Assign OA roles to users.

  The SPA also can remove the OA role from a user.
- Assign the SPA role to other legitimate users under the provider organization (and remove the SPA role).
- Allocate service packages to organizations.

  See "Service Packages" for information about service packages.

  The SPA can assign specified types of service packages to an organization and determine the maximum number of each package that can be used in that organization.

  For example, the SPA could assign the following service packages:
  - In the DEF organization:

    1,000 gold packages

    500 platinum packages
  - In the HIJ organization:

    2,500 topaz packages

    500 platinum packages

    500 emerald packages

    1,000 ruby packages
  - In the SESTA organization:

    2,000 silver packages

    1,500 gold packages

    100 platinum packages

The SPA can use the Delegated Administrator console to perform these tasks. In this release, the Delegated Administrator utility does not include command options to perform these tasks.

Note: The TLA can modify or delete any existing shared organization or full organization. The TLA also can manage users in those organizations.

The TLA can remove the SPA role from a user but cannot assign the SPA role through the console.

See "Administrator Roles and the Directory Hierarchy" for a complete description of the administrative tasks performed by the TLA.

Organization Administrator Role

The OA has the authority to perform the following tasks within the OA's organization:

- Create, delete, and modify users

  In the example shown in Figure 1–6, if the user **johna** is assigned the OA role in the **siroe.com** organization, **johna** can manage users in **siroe.com**
- Create, delete, and modify groups
- Create, delete, and modify Calendar resources
- Assign the OA role to other users
- Assign and remove service packages for users

The OA cannot perform any of these tasks for users, groups, or resources outside the OA's organization.

For example, if **johna** is the OA for **siroe.com** in Figure 1–6, **johna** cannot manage users, groups, or resources in **sesta.com**.

The OA can perform the preceding tasks by using the Delegated Administrator console or by executing Delegated Administrator utility (**commadmin**) commands.

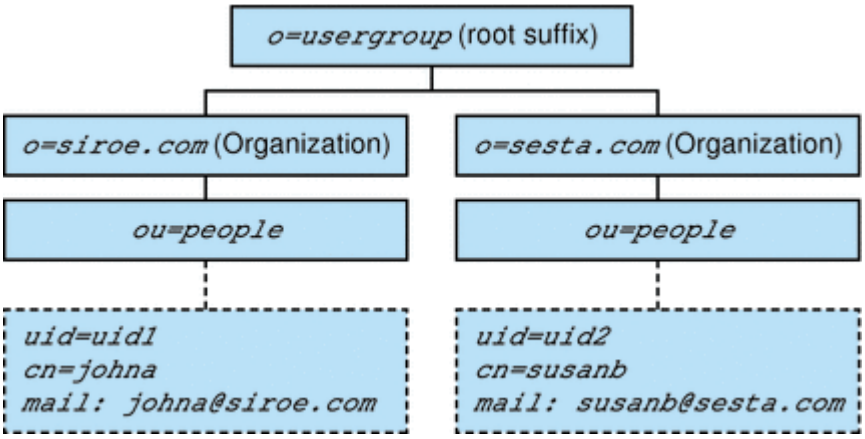
For a list of the **commadmin** commands available to the OA, see "Delegated Administrator Reference".

Directory Structure Supporting a Two-Tier Hierarchy

After Delegated Administrator has been configured with the **config-commda** program, the TLA can create additional organizations, as shown in Figure 1–6.

Figure 1–6 shows a sample Directory Information Tree organized in a two-tier hierarchy.

Figure 1–6 Two-Tier Hierarchy: Sample Directory Information Tree



For Former Users of iPlanet Delegated Administrator

Communications Suite Delegated Administrator is designed for provisioning users in an LDAP Schema 2 directory.

Users of previous versions of Messaging Server who have an LDAP Schema 1 directory may have used iPlanet Delegated Administrator, a deprecated tool. If you still have a Schema 1 directory, you should use iPlanet Delegated Administrator to provision users.

iPlanet Delegated Administrator uses different terms for the administrator roles than those currently used by Oracle Communications Delegated Administrator.

Table 1–2 lists and defines the administrator roles in each version of Delegated Administrator.

Note: In this release of Delegated Administrator, the TLA cannot create provider organizations or business organizations under a provider organization.

Table 1–2 Administrator Roles in iPlanet Delegated Administrator and Communications Suite Delegated Administrator

iPlanet Delegated Administrator	Communications Suite Delegated Administrator Utility	Communications Suite Delegated Administrator Console	Definition
Site Administrator	Top-Level Administrator (TLA)	Top-Level Administrator (TLA)	Manages the entire directory supported by Delegated Administrator, including the organizations and users.
None	None in this release	Service Provider Administrator (SPA)	Manages a provider organization, the shared and full business organizations under the provider organization, and users in those business organizations.
Domain Administrator	Organization Administrator (OA)	Organization Administrator (OA)	Manages one organization and the users in that organization.

2

Service Provider Administrator and Service Provider Organizations

This chapter describes the Service Provider Administrator role and the organization types and explains how to create them in Oracle Communications Delegated Administrator.

Service Provider Administrator

The Delegated Administrator console lets you delegate administrative tasks to a Service Provider Administrator (SPA), who can create and manage subordinate organizations.

The SPA's scope of authority lies between that of the Top-Level Administrator (TLA) and the Organization Administrator (OA).

With the SPA, you can create a three-tiered administrative hierarchy. See "Three-Tier Hierarchy" for more information.

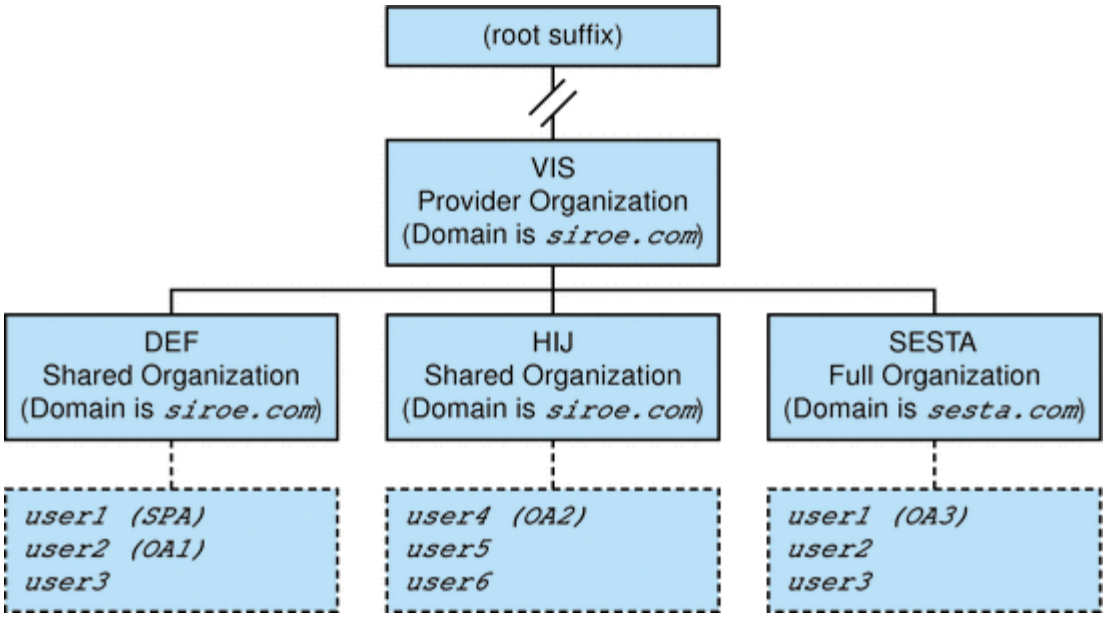
This second level of delegation can ease the management of a large customer base supported by a large LDAP directory. For example, an ISP may offer services to hundreds or thousands of small businesses, each of which requires its own organization. Each day, dozens of organizations might have to be added to the directory.

If you used a two-tiered hierarchy, the TLA would have to create all these organizations. Now the TLA can delegate these tasks to SPAs.

The SPAs can create subordinate organizations for customers and assign OAs to manage users in those organizations.

Figure 2–1 shows a logical view of a sample three-tiered organizational hierarchy.

Figure 2–1 Directory Using a Service Provider Administrator: Logical View



The example in Figure 2–1 shows one provider organization. However, a directory can contain multiple provider organizations.

In this example, administrative tasks are delegated as follows:

- The SPA has the authority to manage the VIS provider organization and all organizations under it. The SPA role is assigned to **user1** in the DEF organization.
- The Organization Administrator named OA1 manages DEF, a shared organization. This OA role is assigned to **user2** in the DEF organization.
- OA2 manages HIJ, a shared organization. This OA role is assigned to **user4** in the HIJ organization.
- OA3 manages SESTA, a full organization. This OA role is assigned to **user1** in the SESTA organization.

  SESTA is a full organization and has its own unique namespace. **user1** in SESTA (in the **sesta.com** domain) has a unique user ID.

See "Organizations Managed by the Service Provider Administrator" for definitions of provider and subordinate organizations.

Service Provider Administrator Role

The SPA can perform the following tasks:

- Create, delete, and modify shared and full organizations in the provider organization in which the SPA has administrative authority. In the example shown in Figure 2–1, the SPA for the VIS provider organization can:
  - Modify or delete the DEF, HIJ, and SESTA organizations.
  - Create additional organizations under the VIS provider organization.
- Create, delete, and modify users in any organization under the provider organization.
- Create, delete, and modify groups in any organization under the provider organization.

- Create, delete, and modify Calendar resources in any organization under the provider organization.
- Assign OA roles to users.

  For example, in the sample organization shown in Figure 2–1, the SPA could assign an OA role to **user2** in the SESTA organization. **user2** could then manage users in the SESTA organization.

  The SPA also can remove the OA role from a user.
- Assign the SPA role to other legitimate users under the provider organization (and remove the SPA role).
- Allocate service packages to organizations.

  See "Service Packages" for information about service packages.

  The SPA can assign specified types of service packages to an organization and determine the maximum number of each package that can be used in that organization.

  For example, the SPA could assign the following service packages:

  – In the DEF organization:

    1,000 gold packages

    500 platinum packages

  – In the HIJ organization:

    2,500 topaz packages

    500 platinum packages

    500 emerald packages

    1,000 ruby packages

  – In the SESTA organization:

    2,000 silver packages

    1,500 gold packages

    100 platinum packages

The SPA can use the Delegated Administrator console to perform these tasks. In this release, the Delegated Administrator utility does not include command options to perform these tasks.

Note: The TLA can modify or delete any existing shared organization or full organization. The TLA also can manage users in those organizations.

The TLA can remove the SPA role from a user but cannot assign the SPA role through the console.

See "Administrator Roles and the Directory Hierarchy" for a complete description of the administrative tasks performed by the TLA.

Assigning the SPA Role to a User

The SPA role must be assigned to a user in an organization designated for SPAs and subordinate to the provider organization that the SPA will manage.

In the example shown in Figure 2–1, assume you need to create an SPA for the provider organization named VIS. You could assign the SPA role to **user1** in the organization DEF.

The SPA must reside in a subordinate organization because a provider organization node does not contain any users.

Thus, before a provider organization can be managed by an SPA, at least one organization must be created under it. This organization should be designated to hold users who are assigned the SPA role. See "Creating a Provider Organization and Service Provider Administrator" for more information.

You cannot use the Delegated Administrator console or utility to create an SPA or a provider organization.

To create an SPA or provider organization, you must manually modify the custom service-provider template, **da.provider.skeleton.ldif**.

See "Creating a Provider Organization and Service Provider Administrator" for instructions on using the custom service-provider template to perform these tasks.

Organizations Managed by the Service Provider Administrator

The SPA can create, modify, and delete the following types of organizations that are subordinate to the SPA's provider organization:

- Provider Organization
- Full Organization
- Shared Organization

The provider organization, full organization, and shared organization are described in the sections that follow.

Provider Organization

A provider organization is a node in the LDAP directory that logically contains full organizations and shared organizations. The provider organization node has attributes that allow the SPA to manage subordinate organizations.

In the LDAP directory, a provider organization must be located under a mail domain. See "Sample Service-Provider Organization Data" for an example.

A provider organization cannot contain user entries. Instead, users are provisioned in the organizations created under the provider organization.

A provider organization stores directory information about the organizations created under it. For example:

- Whether the provider organization can contain shared organizations, full organizations, or both
- Domain names that can be used by the shared organizations created under this provider organization
- The types and number of Class-of-Services packages available to the organizations created under this provider organization

- The organization designated to be the home of the SPA for the provider organization.

Full Organization

A full organization has the following characteristics:

- It is subordinate to the provider organization and is created by the SPA.
- Users can be provisioned in a full organization.

  In the example shown in Figure 2–1, **user2** belongs to the **sesta.com** domain and has a mail address of **user2@sesta.com**.
- As a full organization, it has its own domain that no other organization can share, and it has its own unique namespace.

  In the example shown in Figure 2–1, the full organization, SESTA, has the domain name **sesta.com**.

Shared Organization

A shared organization has the following characteristics:

- It is subordinate to the provider organization and is created by the SPA.
- Users can be provisioned in a shared organization.

  In the example shown in Figure 2–1, **user5** belongs to the **siroe.com** domain and has a mail address of **user5@siroe.com**.
- It uses one or more of the shared domain names from the list provided by the provider organization.

  In the example shown in Figure 2–1, the shared organization DEF uses the domain name **siroe.com**.
- Other shared organizations can share the domain name used by this organization.

  In the example shown in Figure 2–1, both the DEF and HIJ organizations belong to the **siroe.com** domain.
- A shared organization does not have a unique namespace.

Creating a Provider Organization and Service Provider Administrator

You use the custom service-provider template (**da.provider.skeleton.ldif**) provided by Delegated Administrator to create your own provider organizations and SPAs.

Note: You also can install a sample provider organization (with subordinate organizations) and a sample SPA in your directory when you run the Delegated Administrator configuration program. You do this by selecting **Load Sample Organizations** in the configuration program. However, the sample organization template (**da.sample.data.ldif**) is meant to be used as an example, not as a template for creating your own provider organizations. See "Sample Service-Provider Organization Data" for details about this example.

Once you have created a provider organization and an SPA, the SPA can log into the Delegated Administrator console, create and manage subordinate organizations, and

assign the SPA role to other users in the SPA's organization. However, these SPAs can only manage the same provider organization.

To create another provider organization and an SPA to manage it, you should use the custom service-provider template again.

This section contains the following topics:

- "Entries Created by the Template" shows an example of the organizations created when an edited copy of the template is installed in the directory.
- "Information Needed to Create Organizations and SPA" defines the parameters in the template required to create a provider organization, a subordinate shared organization, and an SPA.
- "Steps for Creating a Provider Organization and Service Provider Administrator" explains how to edit the template and install the information in your directory.
- "Modifying Custom Service-Provider Template" is a listing of the template.

Entries Created by the Template

When you install your edited copy of the custom service-provider template in the directory, the following entries are created:

- A provider organization
- A subordinate shared organization designated to hold the SPA user
- One user in the subordinate organization to whom the SPA role is assigned
- A placeholder node under which full organizations can be created. These full organizations will be managed by the SPA for this provider organization.

Example 2–1 shows an example of the entries created by installing the template. It is a Directory Information Tree (DIT) view of the organizations.

Example 2–1 Custom Service-Provider Template: Directory Information Tree View

```
o=usergroup
   o=varrius.com
   o=siroe.com
     o=MyProviderOrg
       o=MySPAUserOrg
       ou=People
          uid=user1
   o=MyProviderOrgDomainsRoot
```

Nodes in the Sample Installed Custom Service-Provider Template

The nodes in the example shown in Example 2–1 are as follows:

- **o=usergroup**: The root suffix for user/group data.
- **o=varrius.com**: The default mail domain.
- **o=siroe.com**: The mail domain used by the provider organization.
- **o=MyProviderOrg**: The provider organization node.
- **o=MySPAUserOrg**: The subordinate shared organization designated to hold the provider organization users, including the user assigned the SPA role.
- **ou=people**: The standard LDAP organization unit required for containing users.

- **uid=user1**: The uid of the user in the MySPAUserOrg organization who is assigned to be the SPA.
- **o=MyProviderOrgDomainsRoot**: The placeholder node for holding full organizations subordinate to the MyProviderOrg provider organization.

Information Needed to Create Organizations and SPA

To create a provider organization, one subordinate organization, and an SPA, you need to replace parameters in the custom service-provider template with information specific to your installation.

As you read about these parameters, you can look at a listing of the **da.provider.skeleton.ldif** shown in "Modifying Custom Service-Provider Template". Or open the actual LDIF file, located in the following directory:

*DelegatedAdmin_home***/lib/config-templates**

For definitions of the attributes associated with these parameters, see *Communications Suite Schema Reference*.

Parameters Defining the Provider and Subordinate Organization

To create a provider organization and subordinate organization, edit the following parameters:

- *ugldapbasedn*

  Root suffix of user/group data in your directory.

  Examples:

  o=usergroup

  o=isp

  dc=red,dc=iplanet,dc=com

- *maildomain_dn*

  Complete DN of the mail domain underneath which the provider organization will be created.

  Examples:

  o=siroe.com,o=usergroup

  o=siroe.com,o=isp

- *maildomain_dn_str*

  The mail domain DN with all commas (,) replaced by underscores (_).

  For example, if the mail domain DN is:

  o=siroe.com,o=isp

  The mail domain DN string will be:

  o=siroe.com_o=isp

- *providerorg*

  Name of the provider organization. The directory node where the provider organization resides will be given this name.

  This parameter is used multiple times in the **da.provider.skeleton.ldif** template.

Examples:

MyProviderOrg

- *servicepackage*

  Name of a service package that can be assigned to users in the organizations subordinate to the provider organization. This is a multivalued parameter. In the "Provider Organization" section of the **da.provider.skeleton.ldif** file, you will see the following attribute:

  sunIncludeServices: *servicepackage*

  For each service package you want to include in the provider organization, add one instance of the **sunIncludeServices** attribute and *servicepackage* parameter. Only those service packages listed here can be assigned to users in subordinate organizations.

  Example:

  sunIncludeServices: gold

  sunIncludeServices: platinum

  sunIncludeServices: ruby

  sunIncludeServices: silver

  sunIncludeServices: mailcalendarimsample

  If you do not use the **sunIncludeServices** attribute (if you delete the line containing the *servicepackage* parameter), all service packages in the directory can be assigned.

- *dns_name*

  Domain name that can be assigned to subordinate organizations in the provider organization. This is a multivalued parameter.

  In the "Provider Organization" section of the **da.provider.skeleton.ldif** file, you will see the following attribute:

  sunAssignableDomains: *dns_name*

  The domain names in the **sunAssignableDomains** attribute are a subset (some or all) of the names listed in the mail domain organization's **sunPreferredDomain** and **associatedDomain** attributes. (The mail domain is the organization under which this provider organization is created.)

  For each domain name you want to include in the provider organization, add one instance of the **sunAssignableDomains** attribute and *dns_name* parameter. Only the domain names listed here can be assigned to subordinate organizations.

  Example:

  sunAssignableDomains: *siroe.com*

  sunAssignableDomains: *siroe.net*

  sunAssignableDomains: *varrius.com*

  sunAssignableDomains: *sesta.com*

  sunAssignableDomains: *sesta.net*

- *provider_business_org*

Name of the shared organization in which the SPA user resides. When you install the edited LDIF information in the directory, this organization is created as shared and subordinate to the provider organization. It is designated as the organization that contains the SPA user. Other users who are assigned the SPA role for this provider organization must reside in this subordinate shared organization.

In the "Provider Organization" section of the **da.provider.skeleton.ldif** file, you will see the following attribute:

sunProviderOrgDN: o=*provider_business_org*,o=*providerorg*,*maildomain_dn*

The **sunProviderOrgDN** attribute identifies the organization designated for provider organization users, particularly the SPA user.

Example:

sunProviderOrgDN: o=MyBusinessOrg,o=MyProviderOrg,o=siroe.com,o=isp

- *preferredmailhost*

  Machine name of the preferred mail host for the provider organization's subordinate organization (in which the SPA user resides). You must use a fully qualified domain name (FQDN).

  In the "Shared Subordinate Organization" section of the **da.provider.skeleton.ldif** file, you will see the following attribute:

  preferredMailHost: *preferredmailhost*

  Example:

  preferredMailHost: mail.siroe.com

- *available_domain_name*

  Domain name that can be assigned to a user in a particular subordinate organization. This is a multivalued parameter.

  The values for *available_domain_name* are a proper subset of the values given for the **sunAssignableDomains***: dns_name* attribute and parameter. Whereas *dns_name* applies to the entire provider organization, *available_domain_name* applies to a single subordinate organization.

  In the "Shared Subordinate Organization" section of the **da.provider.skeleton.ldif** file, you will see the following attribute:

  sunAvailableDomainNames: *available_domain_name*

  For each domain name you want this subordinate organization to inherit from the list of domain names in the provider organization's **sunAssignableDomains** attribute, add one instance of the **sunAvailableDomains** attribute and *available_domain_name* parameter. Only the domain names listed here can be assigned to the subordinate organization.

  Example:

  sunAvailableDomainNames: siroe.com

  sunAvailableDomainNames: siroe.net

  sunAvailableDomainNames: varrius.com

- *available_services*

  Service package available to a particular subordinate organization. This is a multivalued parameter.

The service packages assigned to the subordinate organization are a subset of those assigned to the entire provider organization with the **sunIncludeServices** attribute.

In the "Shared Subordinate Organization" section of the **da.provider.skeleton.ldif** file, you will see the following attribute:

sunAvailableServices: *available_services*

The format of the *available_services* parameter is

servicepackage name: count

where *count* is an integer. If count is absent, the default value is an unlimited number.

For each service package you want this subordinate organization to inherit from the service packages available in the provider organization's **sunIncludeServices** attribute, add one instance of the **sunAvailableServices** attribute and *available_services* parameter.

Example:

sunAvailableServices: gold:1500

sunAvailableServices: platinum:2000

sunAvailableServices: silver:5000

sunAvailableServices: mailcalendarimsample:100

- *is_bind_pwd*

  Bind password for Access Manager found is in the following entry and attribute:

  entry:
  ou=default,ou=OrganizationConfig,ou=1.0,ou=iPlanetAMAuthLDAPService,ou=services,*ugldapbasedn*

  attribute: **sunkeyvalue: iplanet-am-auth-ldap-bind-passwd**

  Example:

  sunkeyvalue:
  iplanet-am-auth-ldap-bind-passwd=AQICoJg2Cvs8tZ+u7pxq1Yp3jKfcxnn5nJ3O

  This parameter is not used if Delegated Administrator is configured in Direct LDAP mode.

Parameters Defining the SPA

To create an SPA, edit the following parameters:

- *spa_uid*

  The user ID for the SPA user.

  Example:

  uid: user1

- *spa_password*

  The password for the SPA user.

  Example:

  userPassword: x12P3&qrS

- *spa_firstname*

  The first name of the SPA user.

  Example:

  givenname: John

- *spa_lastname*

  The last name of the SPA user.

  Example:

  sn: Smith

- *spa_servicepackage*

  The service package assigned to the SPA user. See "Service Packages" for information about service packages.

  Example:

  inetCos: platinum

- *spa_mailaddress*

  The mail address of the SPA user. The domain part of the mail address must be one of the domain values that replace the *available_domain_name* parameter. That is, it must be a domain that has been made available for use in the subordinate organization in which the SPA user resides. See "Parameters Defining the Provider and Subordinate Organization" for more information.

  Example:

  mail: user1@siroe.com

See "Steps for Creating a Provider Organization and Service Provider Administrator" for instructions on how to edit the custom service-provider template and install the information in your directory,

Steps for Creating a Provider Organization and Service Provider Administrator

To create a provider organization and Service Provider Administrator:

This procedure assumes that you have already installed a root suffix and a default mail domain in the directory, as shown in the following example:

o=usergroup

o=varrius.com

1. If you have not already done so, create a mail domain in your directory. The provider organization and its subordinate shared organizations will use this mail domain.

   Example:

   In the following example, **siroe.com** is a mail domain under which the **da.provider.skeleton.ldif** file will install the provider organization and Service Provider Administrator.

   o=usergroup

   o=varrius.com

   o=siroe.com

2. Copy and rename the **da.provider.skeleton.ldif** file.

   When you install Delegated Administrator, the **da.provider.skeleton.ldif** file is installed in the following directory:

   *DelegatedAdmin_home***/lib/config-templates**

3. Edit the following parameters in your copy of the **da.provider.skeleton.ldif** file. Replace the parameters with the correct values for your installation.

   See "Information Needed to Create Organizations and SPA" for definitions of the parameters.

   Some parameters are used more than once in the LDIF file. You must search for and replace all instances of each parameter.

   A few parameters represent values for multivalued attributes. You can copy and edit these parameters (and their associated attribute names) to allow multiple instances of these attributes in your LDIF file. Multivalued parameters are noted below.

   - **ugldapbasedn**
   - **maildomain_dn**
   - **maildomain_dn_str**
   - **providerorg**
   - **servicepackage** (multivalued)
   - **dns_name** (multivalued)
   - **provider_sub_org**
   - **preferredmailhost**
   - **is_bind_pwd**
   - **available_domain_name** (multivalued)
   - **available_services** (multivalued)
   - **spa_uid**
   - **spa_password**
   - **spa_firstname**
   - **spa_lastname**
   - **spa_servicepackage**
   - **spa_mailaddress**

     For definitions of the attributes associated with these parameters, see *Communications Suite Schema Reference*.

4. Use the LDAP directory tool **ldapmodify** to install the provider organization and SPA in the directory. For example, you could run the following command:

   ```
   ldapmodify -D directory manager -w password -f da.provider.finished.ldif
   ```

   where

   - *directory manager* is the bind DN name of the Directory Server administrator.
   - *password* is the password of the Directory Service administrator.

- *da.provider.finished.ldif* is the name of the edited LDIF file to be installed as a provider organization and SPA in the directory.

Example:

The following example shows organization nodes and a Service Provider Administrator user installed under the **siroe.com** mail domain:

o=usergroup

o=varrius.com

o=siroe.com

o=MyProviderOrg

o=MySPAUserOrg

ou=People

uid=user1

o=MyProviderOrgDomainsRoot

The **MyProviderOrgDomainsRoot** organization is located under the root suffix, **usergroup**. **MyProviderOrgDomainsRoot** is the placeholder node created by the ldif; it holds full organizations subordinate to the **MyProviderOrg** organization.

Modifying Custom Service-Provider Template

The template (**da.provider.skeleton.ldif**) contains parameters that you must modify to create a provider organization and SPA.

The listing below shows the sections of the ldif file that have parameters. The listing does not include the entire file. Entries and ACIs required to support Access Manager are not included here.

You should only modify the parameters in the ldif file. Do not modify the sections of the file related to Access Manager.

da.provider.skeleton.ldif File (Relevant Sections)

```
#
# The following parameterized values must be replaced.
#
# <ugldapbasedn>          :: Root suffix for user/group data
# <maildomain_dn>         :: Complete dn of the mail domain underneath
#                            which the provider organization will be
#                            created.
# <maildomain_dn_str>     :: The maildomain dn with all ',' replaced
#                                                    by '_'. E.g.
#                            dn --\> o=siroe.com,o=SharedDomainsRoot,
#                            o=Business,dc=red,dc=iplanet,dc=com
#                            dn_str
--> o=siroe.com_o=SharedDomainsRoot_
#                            o=Business_dc=red_dc=iplanet_dc=com
# <providerorg>            : Organization value for provider node.
# <servicepackage>        :: One for each service package to include.
#                            All service packages in the system
#                            may be assigned by leaving this value empty.
# <dns_name>           :: One for each DNS name which may be assigned
#                            to a subordinate organization.
#                            These names form a proper subset (some or
#                            all) of the names listed in the <maildomain>
```

```
#                           organization's sunpreferreddomain
#                           and associateddomain attributes.
# <provider_sub_org>      :: Organization value for the shared subordinate
#                           organization in which the Provider
#                           Administrator resides.
# <preferredmailhost>     :: Name of the preferred mail host for the
#                           provider's subordinate organization.
# <available_dns_name> :: one for each DNS name that an organization
#                           allows an organization admin to use when
#                           creating a user's mail address.  This is
#                           a proper subset of the values given for
#                           <dns_name> (sunAssignableDomains attribute).
# <available_services>    :: One for each service packags available to an
#                           organization (sunAvailableServices attribute).
#                           These service packages form a proper subset
#                           of the ones assigned to a provider organization
#                           - <servicepackage> (sunIncludeServices
#                           attribute).  Form is
#                           <service package name>:<count>
#                           where count is an integer.  If count is absent
#                           then default is unlimited.
# <spa_uid>               :: The uid for the service provider administrator.
# <spa_password>          :: The password for the service provider
#                           administrator.
# <spa_firstname>         :: First name of the service provider
#                           administrator.
# <spa_lastname>          :: Last name of the service provider
#                           administrator.
# <spa_servicepackage>    :: Service package assigned to the service
#                           provider administrator.
# <spa_mailaddress>       :: The spa's mail address. The domain part of the
#                           mail address must be one of the values used for
#                           <available_dns_name>.
#
#
# Provider Organization
#
dn: o=<providerorg>,<maildomain_dn>
changetype: add
o: <providerorg>
objectClass: top
objectClass: sunismanagedorganization
objectClass: sunmanagedorganization
objectClass: organization
objectClass: sunManagedProvider
sunAllowBusinessOrgType: full
sunAllowBusinessOrgType: shared
sunBusinessOrgBase: o=<providerorg>domainsroot,<ugldapbasedn>
sunIncludeServices: <servicepackage>
sunAssignableDomains: <dns_name>
sunAllowMultipleDomains: true
sunAllowOutsideAdmins: false
sunProviderOrgDN: o=<provider_sub_org>,o=<providerorg>,<maildomain_dn>
# .
# .
# [Entries and ACIs required by Access Manager]
# .
# .
#
# Full Organizations node
```

```
#
dn: o=<providerorg>DomainsRoot,<ugldapbasedn>
changetype: add
o: <providerorg>DomainsRoot
objectClass: top
objectClass: organization
objectClass: sunmanagedorganization
# .
# .
# [Entries and ACIs required by Access Manager]
# .
# .
#
# Provider Admin Role shared organizations
#
dn: cn=Provider Admin Role,o=<providerorg>,<maildomain_dn>
changetype: add
cn: Provider Admin Role
objectClass: ldapsubentry
objectClass: nssimpleroledefinition
objectClass: nsroledefinition
objectClass: nsmanagedroledefinition
objectClass: iplanet-am-managed-role
objectClass: top
iplanet-am-role-description: Provider Admin
#
# Provider Admin Role full organizations
#
dn: cn=Provider Admin Role,o=<providerorg>DomainsRoot,<ugldapbasedn>
changetype: add
cn: Provider Admin Role
objectClass: ldapsubentry
objectClass: nssimpleroledefinition
objectClass: nsroledefinition
objectClass: nsmanagedroledefinition
objectClass: iplanet-am-managed-role
objectClass: top
iplanet-am-role-description: Provider Admin
#
# Shared Subordinate Organization. Includes 1 user who is
# the Provider Administrator.
#
dn: o=<provider_sub_org>,=<providerorg>,<maildomain_dn>
changetype: add
preferredMailHost: <preferredmailhost>
sunNameSpaceUniqueAttrs: uid
o: <provider_sub_org>
objectClass: inetdomainauthinfo
objectClass: top
objectClass: sunismanagedorganization
objectClass: sunnamespace
objectClass: sunmanagedorganization
objectClass: organization
objectClass: sunDelegatedOrganization
objectClass: sunMailOrganization
sunAvailableDomainNames: <available_domain_name>
sunAvailableServices: <available_services>
sunOrgType: shared
sunMaxUsers: -1
sunNumUsers: 1
```

```
sunMaxGroups: -1
sunNumGroups: 0
sunEnableGAB: true
sunAllowMultipleServices: true
inetDomainStatus: active
sunRegisteredServiceName: GroupMailService
sunRegisteredServiceName: DomainMailService
sunRegisteredServiceName: UserMailService
sunRegisteredServiceName: iPlanetAMAuthService
sunRegisteredServiceName: UserCalendarService
sunRegisteredServiceName: iPlanetAMAuthLDAPService
sunRegisteredServiceName: DomainCalendarService
# .
# .
# [Entries and ACIs required by Access Manager]
# .
# .
dn: ou=People,o=<provider_sub_org>,o=<providerorg>,<maildomain_dn>
changetype: add
ou: People
objectClass: iplanet-am-managed-people-container
objectClass: organizationalUnit
objectClass: top
dn: ou=Groups,o=<provider_sub_org>,o=<providerorg>,<maildomain_dn>
changetype: add
ou: Groups
objectClass: iplanet-am-managed-group-container
objectClass: organizationalUnit
objectClass: top
# .
# .
# [Entries and ACIs required by Access Manager]
# .
# .
#
# User - provider administrator
#
dn: uid=<spa_uid>,ou=People,o=<provider_sub_org>,o=<providerorg>, \
        <maildomain_dn>
changetype: add
sn: <spa_lastname>
givenname: <spa_firstname>
cn:  <spa_firstname> <spa_lastname>
uid: <spa_uid>
iplanet-am-modifiable-by: cn=Top-level Admin Role,<ugldapbasedn>
objectClass: inetAdmin
objectClass: top
objectClass: iplanet-am-managed-person
objectClass: iplanet-am-user-service
objectClass: iPlanetPreferences
objectClass: person
objectClass: organizationalPerson
objectClass: inetuser
objectClass: inetOrgPerson
objectClass: ipUser
objectClass: inetMailUser
objectClass: inetLocalMailRecipient
objectClass: inetSubscriber
objectClass: userPresenceProfile
objectClass: icsCalendarUser
```

```
mailhost: <preferredmailhost>
mail: <spa_mailaddress>
maildeliveryoption: mailbox
mailuserstatus: active
inetCos: <spa_servicepackage>
inetUserStatus: Active
nsroledn: cn=Provider Admin Role,o=<providerorg>,<maildomain_dn>
userPassword: <spa_password>
```

Creating Shared and Full Subordinate Organizations

Once you have created a provider organization and an SPA, the SPA can create and manage both shared and full organizations subordinate to the provider organization. The SPA uses the Delegated Administrator console to accomplish these tasks.

The following task outlines the key steps in creating a shared organization or a full organization. This task does not describe how to enter all the information displayed when you create an organization with the Create New Organization wizard. For detailed descriptions of the Create New Organization wizard, see the Delegated Administrator console online help.

To create a shared or full subordinate organization

1. Launch the Delegated Administrator console.

   Go to the following url:

   http://*host*:*port*/da/

   where

   host is the Web container host machine

   port is the Web container port

   For example:

   http://siroe.com:8080/da

   The Delegated Administrator console log-in window appears.

2. Log in to the Delegated Administrator console using the SPA login ID and password.

   "Creating a Provider Organization and Service Provider Administrator" describes how to create an SPA.

   The Service Provider Administrator page appears. The Organizations tab is selected by default. The page displays the organizations subordinate to the SPA's provider organization.

3. Click **New Organization**.

   The Create New Organization wizard appears. For details about entering and selecting information in the Create New Organization wizard, see the Delegated Administrator console online help.

4. Enter information in the Organization Information panel and click **Next**.

   The Contact Information panel appears.

5. Enter information in the Contact Information panel and click **Next**.

   The Account Information panel appears.

6. Determine whether to create a shared organization or full organization.

In the Account Information panel, you determine whether the organization will be shared or full.

A shared organization uses an existing domain shared with other organizations.

A full organization has its own unique domain.

- To create a shared organization, click the **Select from available domains** radio button.

  From the drop-down list, select a domain.

  Note: When you create a shared organization, the Calendar service details are inherited from the existing parent domain. Therefore, you will not enter Calendar service information for the organization. The Calendar Service Details panel will not appear in the Create New Organization wizard. Furthermore, after the shared organization is created, Calendar Service Details do not appear in the organization's Properties page.

- To create a full organization, click **Newdomain**.

  In the text box, enter a mail domain name. For example: **siroe.com.**

  If you want, enter alias names for the domain in the **Alias Names for the New Domain** text box.

7. Enter information in the remaining panels of the Create New Organization wizard.

   For details about these panels, see the Delegated Administrator console online help.

Sample Service-Provider Organization Data

You can choose to install sample organization data (defined in an LDIF file) in your directory when you run the Delegated Administrator configuration program, **config-commda**. (When you run the configuration program, select **Load sample organizations** in the **Service Package and Organization Samples** panel.) The configuration program adds the **da.sample.data.ldif** file to the LDAP directory tree.

This LDIF file is meant to be used as an example, not as a template for creating your own provider organizations. See "Information Needed to Create Organizations and SPA" for information on how to create a provider organization.

Figure 2–1 shows a logical view of the organizational structure provided by the sample LDIF file. (Figure 2–1 adds a shared organization, HIJ, that does not exist in the file.)

The sample LDIF file contains the following organizations under the root-suffix nodes:

- VIS provider organization. The following organizations are managed by the SPA for the VIS provider organization:
  - SESTA, a full organization. The SESTA organization has its own domain, **sesta.com.**
  - DEF, a shared organization. The DEF organization uses the shared domain, **siroe.com**.
- ESG provider organization. No subordinate organizations are defined for this provider organization.

The LDIF file defines the following administrator roles for these organizations:

- An SPA for the VIS provider organization (**user2@abc.com**)
- An SPA for the ESG provider organization (**user2_def**)
- An OA for the SESTA organization (**user1@abc.com**)
- An OA for the DEF organization (**user1_def**)

Logical Hierarchy and the Directory Information Tree

In a three-tiered directory hierarchy, a Directory Information Tree (DIT) does not look exactly like the logical view shown in Figure 2–1. Organizations are implemented in the DIT in a somewhat different hierarchy.

For example, in a DIT, full domains must reside directly under the root suffix. Therefore, domain nodes are added under the root suffix to store LDAP information for shared domains (used by shared organizations) and for full organizations (which have their own domains).

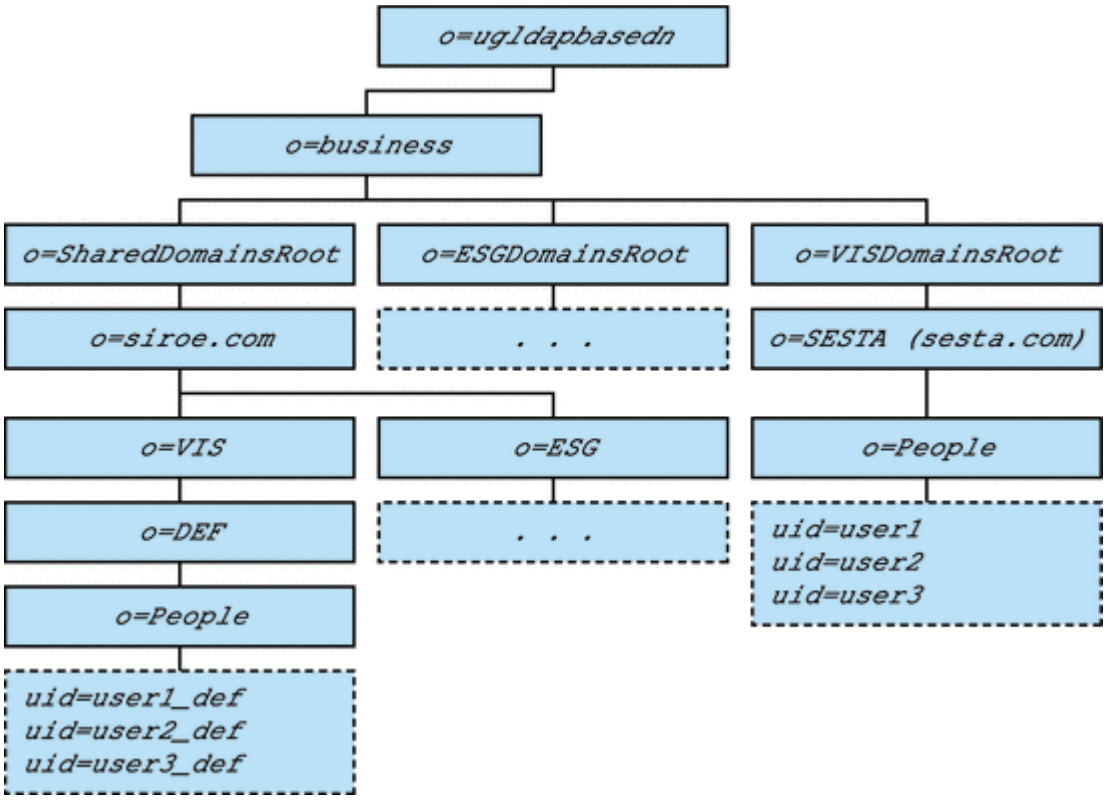
Sample Organization Data: Directory Information Tree View

Figure 2–2 shows a Directory Information Tree (DIT) view of the sample organization data.

The example shown in Figure 2–2, like the logical view shown in Figure 2–1, contains the following organizations:

- VIS and ESG (provider organizations)
- DEF, a shared organization subordinate to the VIS provider organization
- SESTA, a full organization subordinate to the VIS provider organization

Figure 2–2 Sample Organization Data: Directory Information Tree View



Nodes in the Sample Directory Information Tree

The nodes in the sample organization file (**da.sample.data.ldif**) are as follows:

- *ugldapbasedn*: This parameter represents the root suffix.
- o=business: A node that contains all businesses in the directory.
- o=SharedDomainsRoot: A node needed to contain the domains used by shared organizations.

  In this Directory Information Tree, shared organizations subordinate to different service provider organizations can use the same shared domain. This can be done because both the provider organizations have nodes under the **SharedDomainsRoot** node.
- o=ESGDomainsRoot and o=VISDomainsRoot: These nodes contain any full organizations that are subordinate to the ESG and VIS provider organizations.

  Each provider organization that manages full organizations must have a node at this level (under the root suffix).

  Multiple full organizations, each with its own domain, can exist under **ESGDomainsRoot** or **VISDomainsRoot**.
- o=siroe.com: The shared domain. It is used by the shared organization, DEF.
- o=VIS and o=ESG: These provider organization nodes contain any shared organizations subordinate to the VIS and ESG provider organizations.

  For example, the shared organization, DEF, is subordinate to the VIS provider organization.

- o=SESTA: The full organization. It has its own domain, **sesta.com**.
- o=DEF: The shared organization. It uses the domain **siroe.com**.
- ou=people: The standard LDAP organization unit required for containing users.

User DNs in the Sample Directory Information Tree

Some user DNs in the sample organization file shown in Figure 2–2 are as follows:

- For the user named **user1_def**, who belongs to the DEF organization:

  dn: uid=user1_def,ou=People,o=DEF,o=VIS,o=siroe.com, o=SharedDomainsRoot,o=Business,*ugldapbasedn*

- For the user named **user1**, who belongs to the SESTA organization:

  dn: uid=user1,ou=People,o=SESTA,o=VISDomainsRoot, o=Business,*ugldapbasedn*

3

Service Packages

This chapter contains information on service packages in Oracle Communications Delegated Administrator. The chapter reviews what a service package is, samples of those service packages and information on creating your own service packages.

What Is A Service Package?

A service package is implemented by the Class-of-Service mechanism in the LDAP directory. This mechanism lets you set values for predefined attributes that are installed in the directory when you configure Delegated Administrator. A service package adds the characteristics of the service to user or group entries.

When you configure Delegated Administrator (with **config-commda**), a sample set of Class-of-Service templates is available. You can also create your own service packages, by hand. In the Delegated Administrator console, you can assign the sample packages and your own packages to users or groups.

> **Note:** Delegated Administrator utility (**commadmin**) has no Service Package integration.

For additional information about all the sample service packages, the class-of-service definitions, and the locations of the packages and definitions in the LDAP directory, see "Service Package Details."

A service package coordinates the usage of the following components:

- Access Manager service (like "UserMailService")
- Service bundle (one or more of these services: mail service, calendar service, instant messaging service, contacts service)
- A type of LDAP object (users or groups)

Delegated Administrator automatically provides Access Manager service with every service definition. When you assign a service package to a user or group, Delegated Administrator takes the Access Manager object classes and attributes from the service definition and adds them to the LDAP entry.

Do not change or delete the Access Manager portion of any service package.

Service Bundles

Delegated Administrator provides bundles for these services: mail service, calendar service, instant messaging service, and contacts service.

A service package bundles together one or more services and a set of attributes associated with those services. Thus, an individual service package can contain the following combinations of services:

- Mail service only
- Calendar service only
- Instant messaging service only
- Contacts service only
- Some combinations of two or three services for example, mail and calendar services, mail and instant messaging services, mail, calendar, and instant messaging services.
- All four services

Note: Only mail service package templates, instant messaging service package templates, and contacts service package templates have LDAP attributes associated with their Class-of-Service definitions. The Calendar service package templates do not include attributes associated with the Calendar service definition.

Packages Defined for Particular LDAP Objects

A service package is defined either for users or for groups. You cannot assign the same service package to a user and a group.

Delegated Administrator provides sample service packages with the following service bundles and LDAP objects:

- User mail service
- User calendar service
- User instant messaging service
- User contacts service
- User mail and calendar service
- User mail and instant messaging service
- User mail and contacts service
- User calendar and instant messaging service
- User calendar and contacts service
- User mail, calendar, and instant messaging service
- User mail, calendar, instant messaging, and contacts service
- Group mail service
- Group calendar service
- Group mail and calendar service

Assigning Service Packages to Groups

In Communications Suite, a group is an entry in the LDAP directory that comprises a list of users. Characteristics of the group are not passed on to the users who are members of the group. Thus, when you assign a service package to a group, the

service package attributes modify the group, not all the users in the group. The user entries in the directory are not subordinate to (do not belong to) the group entry.

When a mail service package is assigned to a group, the group becomes a mailing list, which is used by Messaging Server.

When calendar service is assigned to a group, the members of the group share group invitations and other calendar information managed by Calendar Server.

A mail group does not have its own mailbox; a message sent to the group address is delivered to the mailboxes of the individual members of the group.

However, a calendar group does have its own calendar; an invitation sent to the group is displayed on the group calendar and on the calendars of the individual members of the group.

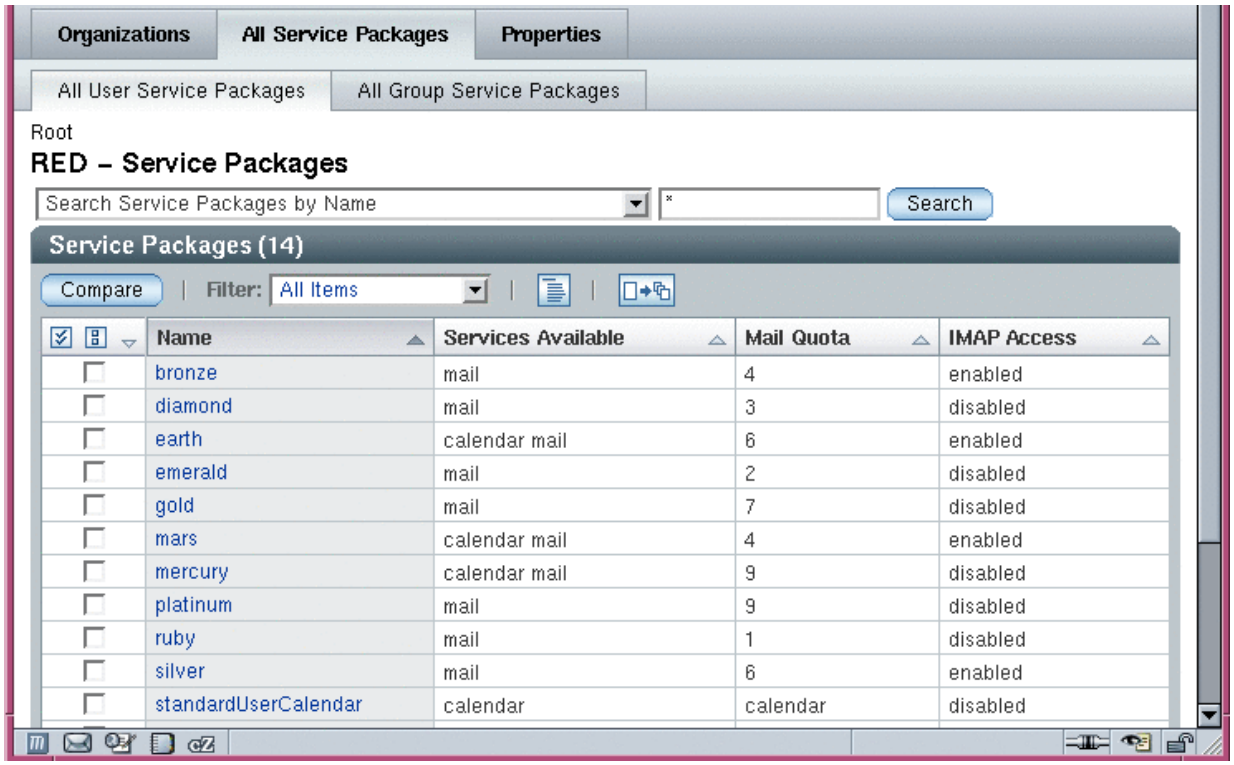
Sample Service Packages Provided by Delegated Administrator

When you configure Delegated Administrator, you can choose to install a set of predefined, sample Class-of-Service templates. (When you run the configuration program, select **Load sample service packages** in the **Service Package and Organization Samples** panel.) The configuration program adds the **cos.sample.ldif** file to the LDAP directory.

You can use the sample templates to provide services and mail attributes to users and groups. See "Service Package Details" for a list of the templates with their attribute values.

Figure 3–1 shows the user service package templates.

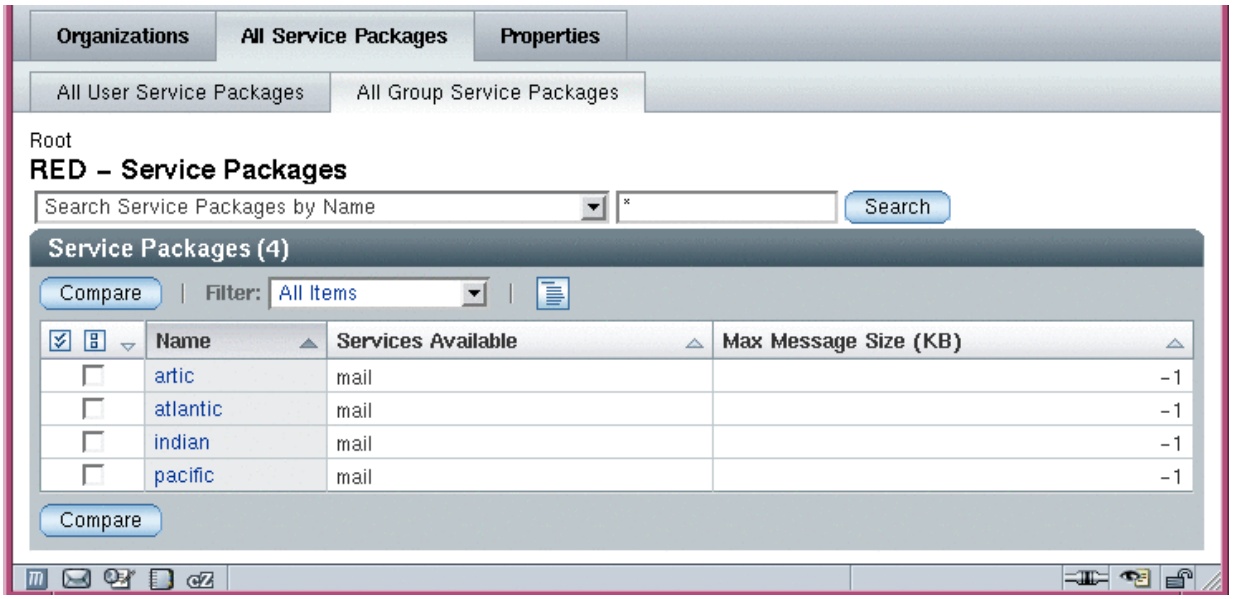
Figure 3–1 All User Service Packages Page - Sample Templates Displayed



Name	Services Available	Mail Quota	IMAP Access
bronze	mail	4	enabled
diamond	mail	3	disabled
earth	calendar mail	6	enabled
emerald	mail	2	disabled
gold	mail	7	disabled
mars	calendar mail	4	enabled
mercury	calendar mail	9	disabled
platinum	mail	9	disabled
ruby	mail	1	disabled
silver	mail	6	enabled
standardUserCalendar	calendar	calendar	disabled

Figure 3–2 shows the group service package templates.

Figure 3–2 All Group Service Packages Page - Sample Templates Displayed



Service-Package Tasks

In the Delegated Administrator console, you perform the following service-package tasks:

- Allocate service packages to organizations. By allocating some (or all) packages to an organization, you make those packages available to users or groups in the organization.

  For each package, you allocate a specified number of packages.

  For example, for the ABC organization, you might allocate 5,000 gold service packages, 10,000 venus service packages, and 500 atlantic service packages.
- Assign service packages to users.
- Assign service packages to groups.

> **Note:** Service package tasks cannot be performed with the Delegated Administrator utility (**commadmin**).

Guidelines for Assigning Service Packages

- The service packages allocated to an organization form the pool from which service packages can be assigned to users or groups in the organization.
- You can assign multiple service packages to a user or group.
- When you assign a service package to a user or group, all the attributes and values in the service package are automatically assigned to the user or group.
- To assign only calendar service to a user, use the standardUserCalendar service package. Calendar service does not have any associated attributes.

  Assigning the standardUserCalendar service package is equivalent to using the **-s cal** option in the **commadmin user create** or **commadmin user modify** command.

For instructions on how to allocate and assign service packages, see the Delegated Administrator console online help.

Creating Your Own Service Packages

If you want to create your own service packages, you add a Class of Service template to the DIT. You can use a Class-of-Service template stored in the **da.cos.skeleton.ldif** file. This file was created specifically for use as a template for writing service packages. See "Service Package Details" for more information.

Limitations in Viewing an Extended Service Package

By default, the service packages apply to a list of attributes in the COS definition.

You can extend a service package by adding any attribute to the definition entry. However, the console allows you to view only the predefined attributes provided when Delegated Administrator is configured. The Delegated Administrator console does not display any additional attributes you add to a service package definition.

Note: You should not remove the predefined attribute definitions from the Class-of-Service definitions provided by Delegated Administrator.

Example Service Package Assigned to an LDAP Entry

When you use Delegated Administrator to assign a service package to a user or group, a single attribute (**inetCOS**) is added to the target for every service package. The value of the **inetCOS** attribute assigns the entire service package to the user or group, including the service and any attributes associated with that service (**inetCOS** is a multi-valued attribute).

For example, suppose you assign the platinum package to a user. The following attribute is added to the user entry:

```
inetCOS: platinum
```

The platinum package provides mail service to the user. The package also contains the following values for mail attributes. Thus, assigning the platinum package has the effect of adding these attributes to the user entry:

```
mailMsgMaxBlocks: 800
mailQuota: 10000000
mailMsgQuota: 6000
mailAllowedServiceAccess:
+imap:ALL$+imaps:ALL$+pop:ALL$+pops:ALL$+smtp:ALL$+http:ALL
```

This change takes effect dynamically, through the LDAP server's Class-of-Service feature.

The Access Manager service definition provides the object classes and attributes required for the mail and calendar service. When you assign the service package, Delegated Administrator adds the object classes and attributes required for the service to the user or group entry.

See "Service Package Details" for more information about all the sample service packages, the class-of-service definitions, and the locations of the packages and definitions in the LDAP directory.

4

Removing Users, Groups, and Services from a Domain

This chapter describes how to remove users, groups, and Calendar resources from a domain, and how to remove services (such as mail service) from users in a domain.

Overview of Removing Users and Services from a Domain

This procedure requires three main tasks:

1. Mark users or services for deletion.

   You can use the Delegated Administrator console or **commadmin** command to perform this task.

2. Remove the application resources for that user or group.

   You use the application's utility to perform this task.

   After an LDAP entry or service has been marked as deleted in the directory, you must run a utility that removes resources such as mailboxes or calendars before the entry or service can be purged from the directory:

   - For mail services, the command is called **msuserpurge**.
   - In Calendar Server 7, for calendar services, the command is **davadmin** and in Calendar Server 6, for calendar services, the command is **csclean**.
   - If instant messaging (Instant Messaging Server) service has been marked as deleted, the **commadmin domain purge** command purges Instant Messaging Server service from the directory. You do not have to run an Instant Messaging Server-specific utility to remove Instant Messaging Server resources, as you do for mail or calendar service.
   - If contacts service has been marked as deleted, the **commadmin domain purge** command purges contacts service from the directory. You do not have to run a contacts server-specific utility to remove contacts resources, as you do for mail or calendar service.

   Note: This step removes the resource from the application, not from the LDAP directory. For example, **msuserpurge** removes the user's mailbox from the message store

3. Remove the entry or service from the LDAP directory.

You use the **commadmin domain purge** command to perform this task. For details about this command, see "Delegated Administrator Reference".

To Remove Users, Groups, and Calendar Resources from a Domain

This procedure permanently removes users, groups, and Calendar resources from a domain. The domain itself remains intact in the LDAP directory. Only the LDAP entries selected for deletion are removed.

Note: The **commadmin domain purge** command does not remove the user as a member from any groups of which the user is a member. To completely remove a user s entry from the directory you must enable the Referential Integrity plug-in, see the discussion about maintaining referential integrity in *Oracle Fusion Middleware Administration Guide for Oracle Directory Server Enterprise Edition 11.*

---

1. Mark the users, groups, and resources as deleted.

   For example, to mark selected entries as deleted in the **florizel.com** domain:

   ```
   commadmin user delete  chris -d florizel.com -n sesta.com -i deletedusers

   commadmin group delete -D chris -d florizel.com -n sesta.com -i deletedgroups

   commadmin resource delete -D chris -d florizel.com -n sesta.com -i
   deletedresources
   ```

   In the preceding examples, **deletedusers**, **deletedgroups**, and **deletedresources** are input files listing the entries marked for deletion.

   You also can use the Delegated Administrator console to delete entries:

   a. Navigate to the specified organization.

   b. Click the **Users** tab (if it is not already displayed), select the users to be deleted, and click **Delete**.

   c. Click the **Groups** tab, select the groups to be deleted, and click **Delete**.

   d. Click the **Resources** tab, select the resources to be deleted, and click **Delete**.

2. Remove resources from the selected users, groups, and calendars in the domain.

   A resource can be a mailbox or a calendar. (Instant Messaging Server and Contacts Server do not have associated resources that need to be removed.)

   For mail services, run the **msuserpurge** command.

   For information about the command see *Messaging Server Reference*.

   For calendar services, run the **davadmin** command for Calendar Server 7. Run the **csclean** command for Calendar Server 6.

   For information about the **davadmin** command see *Calendar Server System Administrator's Guide*. For information about the **csclean** command see *Administration Reference* for Calendar Server 6.3.

3. Permanently remove the selected entries from the domain by invoking the **commadmin domain purge** command. For details about this command, see "Delegated Administrator Reference".

For example, to remove selected users, groups, and resources from the florizel.com domain:

```
commadmin domain purge -D chris -d florizel.com -n sesta.com
```

In the preceding command, the **florizel.com** domain remains intact. Only the entries specified in the **deletedusers**, **deletedgroups**, and **deletedresources** input files are removed.

To Remove Services from a Domain

This procedure permanently removes mail, calendar, instant messaging (Instant Messaging Server), and contacts services from a domain and from each user, group, and resource in the domain. The domain itself, including its subordinate LDAP entries, remains intact in the directory.

1. Mark the service(s) in the domain as deleted by running the **commadmin domain delete** command.

   For example, to mark mail, calendar, Instant Messaging Server, and contacts services as deleted in the **florizel.com** domain:

   ```
   commadmin domain delete -D chris -d florizel.com -n sesta.com -S
   mail,cal,im,contacts
   ```

2. Remove resources from all users, groups, and resources in the domain.

   A resource can be a mailbox or a calendar. (Instant Messaging Server and Contacts Server do not have associated resources that need to be removed.)

   For mail services, run the **msuserpurge** command.

   For information about the command see *Messaging Server Reference*.

   For calendar services, run the **davadmin** command for Calendar Server 7. Run the **csclean** command for Calendar Server 6.

   For information about the **davadmin** command see *Calendar Server System Administrator's Guide*. For information about the **csclean** command see *Calendar Server 6.3 Administration Reference*.

   Note: If the mailbox or calendar of any user in the domain is *not* removed, the service cannot be purged from the domain. For example, for mail service, ensure the grace period has been reached and that the **msuserpurge** command has been run on all mail message stores encompassed by the domain.

3. Permanently remove the service(s) from the domain by invoking the **commadmin domain purge** command. For details about this command, see "Delegated Administrator Reference."

   For example, to remove mail, calendar, Instant Messaging Server, and contacts services from the florizel.com domain:

   ```
   commadmin domain purge -D chris -d florizel.com -n sesta.com -S
   mail,cal,im,contacts
   ```

To Permanently Remove an Entire Domain

This procedure permanently removes a domain from the directory. All user, group, and resource entries in the domain are also removed from the directory.

1. Mark the domain as deleted by running the **commadmin domain delete** command.

   For example, to mark the florizel.com domain as deleted:

   ```
   commadmin domain delete -D chris -d florizel.com -n sesta.com
   ```

   You also can use the Delegated Administrator console to mark the domain for deletion by selecting the organization on the Organizations page and clicking **Delete**.

2. Remove resources from all users, groups, and resources in the domain.

   A resource can be a mailbox or a calendar. (Instant Messaging Server and Contacts Server do not have associated resources that need to be removed.)

   For mail services, run the **msuserpurge** command.

   For information about the command see *Messaging Server Reference*.

   For calendar services, run the **davadmin** command for Calendar Server 7. Run the **csclean** command for Calendar Server 6.

   For information about the **davadmin** command see *Calendar Server System Administrator's Guide*. For information about the **csclean** command see *Calendar Server 6.3 Administration Reference*.

   Note: If the mailbox or calendar of any user in the domain is *not* removed, the domain cannot be removed. For example, for mail service, ensure the grace period has been reached and that the **msuserpurge** command has been run on all mail message stores encompassed by the domain.

3. Permanently remove the domain by invoking the **commadmin domain purge** command. For details about this command, see "Delegated Administrator Reference".

   For example, to remove the **florizel.com** domain

   ```
   commadmin domain purge -D chris -d florizel.com -n sesta.com
   ```

5

Setting Calendar Server Advanced Rights

This chapter describes how to set Oracle Communications Calendar Server's advanced rights by using Oracle Communications Delegated Administrator.

Setting Calendar-Service Advanced Rights in the Delegated Administrator Console

You can set Calendar Service attributes (including Advanced Rights) for an organization by editing the Organization Properties page in the Delegated Administrator console.

To set an organization's properties:

1. In the Organizations page, click the name of the organization you want to modify. The organization's Users page appears.
2. Click the **Properties** tab in the second (lower) row of tabs. Do not click the upper Properties tab; that will display properties for the root suffix of the directory.
3. Navigate to the **Calendar Service** section.
4. When you have finished viewing or modifying information for the organization, take one of these steps:
   - Click **Save** to save the changes you have made.
   - Click **Reset** to restore the organization's properties to the values they had when you last saved them.

You can use the Delegated Administrator console to set Calendar Service advanced rights for a specified organization.

Delegated Administrator stores these domain-specific rights in the LDAP directory by setting values for the **icsAllowRights** attribute. For information about this attribute, see the discussion about Messaging Server and Calendar Server LDAP Object Classes and Attributes in *Communications Suite Schema Reference*.

Relationship Between the Organization-Level Settings and Application-Level Default Settings

The current Delegated Administrator per-domain calendar service advanced rights interface can be misinterpreted.

For each allowed rights option, you can select a **yes** or **no** option that affects the current organization (domain). For example:

```
Allow double booking for user calendar: [yes] [no]
```

If you set the "Allow double booking for user calendar" right to **yes**, new user calendars in this domain will have the double-booking setting enabled.

Note: You can modify the double-booking setting for existing user, resource and group calendars by using the **cscal -k** command.

However, if you set this right to **no**, the Calendar Server-wide default for this option is used. In other words, a value of **no** means the following: "Do not set any value for this domain. Defer to the Calendar Server-wide default setting."

Table 5–1 lists the relationships between domain-specific rights and Calendar-Server default values.

Table 5–1 Relationships between domain-specific rights and Calendar Server default values

Delegated Administrator Advanced Rights Setting	Calendar Server-Wide Default Setting	What Happens?
no	**yes**	Advanced right is *allowed* for this domain
no	**no**	Advanced right is *disallowed* for this domain
yes	**yes**	Advanced right is *allowed* for this domain
yes	**no**	Advanced right is *allowed* for this domain

The following guidelines apply to setting advanced rights:

- By default, all of the advanced rights options are set to **no**, which represents the **icsAllowRights** attribute being unset or set to **0**.
- When you set the option to **yes**, you override the Calendar Server-wide default for this option. (The default is set in the **ics.conf** file.) That is, **yes** *allows* the right for the specified domain, which is the expected behavior.
- However, when you set the option to **no**, the Calendar Server-wide default for this option is used. A **no** value does *not* disallow the right.
- For example, if you set the "Allow double booking for user calendar" right Calendar Service advanced rights option to **no**, and the calendar server default value is **yes**, the option is *allowed* for this domain.
- The Delegated Administrator console interface and the online help state that setting an option to **no** disallows the right. This is incorrect.

6

Improving Delegated Administrator Performance

The following topics describe how you can tune Oracle Communications Delegated Administrator and related software to improve Delegated Administrator performance.

In addition to the guidelines described in this chapter, you can improve Directory Server performance by consolidating and reducing the number of default ACIs in the directory. See "Consolidating ACIs for Directory Server Performance" for information.

Speeding Up Display of Users, Groups, and Organizations

If an organization contains many users, the Delegated Administrator console may take time to display the User list page. If you try to create or edit a user while the page is still loading the existing users, an error occurs. Do not click any buttons or links until the page is ready.

Similarly, it can take time to open the Organization page or Group page if your directory contains many organizations or groups.

If these pages take too long to load, you can set wild-card search properties to a sufficiently low value to allow the pages to load quickly.

The properties are:

- **jdapi-wildusersearchmaxresults**: Search property for users.
- **jdapi-groupsmaxsearchresults**: Search property for groups
- **jdapi-wildorgsearchmaxresults**: Search property for organizations.

The wild-card search property limits are as follows:

- **-1**: Return all results. (Display all users, groups, or organizations.) -1 is the default value.
- **0**: Do not search. (Display no users, groups, or organizations.)
- *n* (**>0**): Return *n* (the specified number of results).

Displaying the User Page More Quickly

To display the user page more quickly:

1. Open the **resource.properties** file.

   The **resource.properties** file is located in the following directory:

   *DelegatedAdmin_home***/data/WEB INF/classes/sun/comm/cli/server/servlet**

2. Set the value of **jdapi-wildusersearchmaxresults** to a low value. For example:

   ```
   jdapi-wildusersearchmaxresults=50
   ```

   Alternatively, you can set the value to **0** to display no users. In the Delegated Administrator console, use the **Search** drop-down list to search for specified users.

3. Redeploy the edited **resource.properties** file to the Web container used by the Delegated Administrator server.

   Before the change can take effect, you must run the script that deploys the customized **resource.properties** file to your Web container.

   See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

Displaying the Group Page More Quickly

1. Open the **resource.properties** file.

   The **resource.properties** file is located in the following directory:

   *DelegatedAdmin_home***/data/WEB INF/classes/sun/comm/cli/server/servlet**

2. Set the value of **jdapi-groupsmaxsearchresults** to a low value. For example:

   ```
   jdapi-groupsmaxsearchresults=50
   ```

   Alternatively, you can set the value to **0** to display no groups. In the Delegated Administrator console, use the **Search** drop-down list to search for specified groups.

3. Redeploy the edited **resource.properties** file to the Web container used by the Delegated Administrator server.

   Before the change can take effect, you must run the script that deploys the customized **resource.properties** file to your Web container.

   See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

Displaying the Organization Page More Quickly

1. Open the **resource.properties** file.

   The **resource.properties** file is located in the following directory:

   *DelegatedAdmin_home***/data/WEB INF/classes/sun/comm/cli/server/servlet**

2. Set the value of **jdapi-wildorgsearchmaxresults** to a low value. For example:

   ```
   jdapi-wildorgsearchmaxresults=10
   ```

   Alternatively, you can set the value to **0** to display no organizations. In the Delegated Administrator console, use the **Search** drop-down list to search for specified organizations.

3. Redeploy the edited **resource.properties** file to the Web container used by the Delegated Administrator server.

   Before the change can take effect, you must run the script that deploys the customized **resource.properties** file to your Web container.

   See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

Increase JVM Heap Size

To improve the performance of common Delegated Administrator functions such as displaying pages and performing searches, you can increase the Java Virtual Machine (JVM) heap size used by the Web container to which Delegated Administrator is deployed. When the Web container's JVM heap size is too small, performance can be affected.

The JVM heap size is set by the following JVM option:

-Xmx*n*m

where <*n*> is the heap size in megabytes.

Typically, <*n*> is set to **256m**.

The following tasks outline how to set a higher JVM heap size for Web Server and Application Server.

Increasing the Web Server 6.*x* JVM Heap Size

1. Log in to the Web Server Administration Server.
2. Under the Java tab, select **JVM Options**.
3. Edit the **-Xmx256m** option.

   This option sets the JVM heap size.
4. Set the **-Xmx256m** option to a higher value, such as **Xmx1024m**.
5. Save the updated setting.

Web Server Documentation

See the *Sun Java System Web Server Administration Guide* and *Web Server Performance Tuning, Sizing, and Scaling Guide* for more information about using the Web Server Administration Server and setting JVM options.

Increasing the Web Server 7.*x* JVM Heap Size

1. Log in to the Web Server Administration Server.
2. Under the Configuration Tasks section, select **Edit Java Settings**.
3. Click the **JVM Settings** tab to display the JVM options.
4. Edit the **-Xmx256m** option.

   This option sets the JVM heap size.
5. Set the **-Xmx256m** option to a higher value, such as **Xmx1024m**.
6. Save the updated setting.

Web Server Documentation

See the *Sun Java System Web Server Administration Guide* and *Web Server Performance Tuning, Sizing, and Scaling Guide* for more information about using the Web Server Administration Server and setting JVM options.

Increasing the Application Server JVM Heap Size

1. Log in to the Application Server Administration Server.
2. Navigate to the JVM options.

3. Edit the **-Xmx256m** option.

   This option sets the JVM heap size.

4. Set the **-Xmx256m** option to a higher value, such as **Xmx1024m**.

5. Save the updated setting.

Application Server Documentation

For more information about using the Application Server Administration Server and setting JVM options, go to the *Sun Java System Application Server Documentation Center* and select **JVM Advanced Settings**. Alternatively, see Tuning the Java Runtime System in *Sun Java System Application Server Enterprise Edition 8.1 2005Q4 Performance Tuning Guide*.

Raise Directory Server Indexing Threshold

To improve performance of Delegated Administrator functions such as searching and displaying users, you can increase the threshold for indexes used by Directory Server to search the directory.

When Directory Server searches several LDAP objects, if the threshold is set to a low value, the index might run out of space before the search is completed. The remainder of the search is performed without indexing, which slows down the search operation.

> **Caution:** Perform this operation only if you are an experienced Directory Server administrator.

To set the index threshold to a higher value, change the value of the **nssldap-allidsthreshold** option in the **dse.ldif** file.

This option might be set to a value such as the following:

```
nssldap allidsthreshold: 4000
```

Set **nssldap-allidsthreshold** to a higher value. For example:

```
nssldap allidsthreshold: 200000
```

For more information about the All IDs Threshold, see Managing Indexes in Indexing Directory Data in the *Sun Java System Directory Server Administration Guide*. For a definition of the **nssldap-allidsthreshold** option, see "Database Configuration Attributes" in *Sun Java System Directory Server Administration Reference*.

In some cases, when you are logged in as the Top Level Administrator and you click on the Advanced Search button, you may find that the search does not work for certain fields especially if there is a huge amount of data. To resolve this, you should reindex the backend on the root suffix for the particular field you are searching. By default mail and user name fields are indexed by **dssetup** so this problem does not occur for those attributes.

For example, to create an index for **departmentNumber** so that you can search for the department and get a list of the users, do the following:

```
dsconf create-index -h localhost -p 389 dc=idc,dc=oracle,dc=com  departmentNumber

dsconf set-index-prop -h localhost -p 389 dc=idc,dc=oracle,dc=com departmentNumber
sub-enabled:on approx-enabled:on
```

```
dsconf reindex -h localhost -p 389 -t departmentNumber dc=idc,dc=oracle,dc=com

dsadm stop /var/opt/DS7Instance/

dsadm reindex -t departmentNumber /var/opt/DS7Instance/ dc=idc,dc=oracle,dc=com

dsadm start /var/opt/DS7Instance/
```

If you need to index a different attribute, replace the attribute **departmentNumber** with another attribute that you need to index like **city**. Note that you will have to confirm the existence of the attribute in LDAP.

7

Consolidating ACIs for Directory Server Performance

This chapter contains information on different types of Access Control Instructions, and how to interact with them.

Introduction

When you install Access Manager with Messaging Server and use an LDAP Schema 2 directory, a large number of Access Control Instructions (ACIs) initially are installed in the directory. Many default ACIs are not needed or used by Messaging Server.

The need to check these ACIs at run time can affect the performance of Directory Server, which can, in turn, affect the performance of Messaging Server look-ups and other directory operations.

You can improve the performance of the Directory Server by consolidating and reducing the number of default ACIs in the directory. Consolidating the ACIs also makes them easier to manage.

The approach to reducing ACIs is as follows:

- Combine, optimize, and simplify redundant ACIs
- Modify ACIs to use a simpler, more efficient syntax
- Consolidate ACIs with other ACIs (at the root suffix)
- Eliminate unused ACIs
- For directories with many organizations, allows organization ACIs to be removed on individual organization nodes

This chapter first describes how to use an LDIF file (**replacment.acis.ldif**) to consolidate ACIs at the root suffix and remove unused ACIs from the directory. See "Consolidating and Removing ACIs" for details.

Next, the chapter analyzes each ACI and recommends a method for handling it: removing it, revising it to make it more efficient, or rewriting it.

Note the following constraints in these recommendations:

- There is no end-user access for the Directory console.
- There is no end-user access to the Access Manager console.

Given these constraints, you must determine for yourself (according to the requirements of your installation) whether you can use the LDIF file to consolidate

and remove ACIs, or whether you need to retain certain ACIs as they now exist in the directory.

See "Analysis of the Existing ACIs" for more information.

Next, this chapter describes the ACIs that are consolidated by the **replacement.acis.ldif** file. It lists the existing ACIs before they are consolidated and the modified ACIs after they are consolidated. See "Analysis of How ACIs Are Consolidated" for more information.

Finally, the chapter lists the ACIs discarded by the **replacement.acis.ldif**. See "List of Unused ACIs to be Discarded" for more information.

Consolidating and Removing ACIs

The LDIF file listed in this section, **replacement.acis.ldif**, installs consolidated ACIs at the root suffix and deletes unused ACIs from the directory. This LDIF file is provided by Delegated Administrator, located in the following directory:

DelegatedAdmin_home/lib/config-templates

When you apply the **replacement.acis.ldif** file to the directory (with **ldapmodify**), the **ldapmodify** command removes all instances of the **aci** attribute at the root suffix and replaces these ACIs with the ACIs in the **replacement.acis.ldif** file.

Thus, this procedure will initially remove *all* ACIs from the root suffix and then replace them with the set of ACIs listed below. If the directory contains ACIs generated by another application such as Portal Server, you should save those ACIs to a file and reapply them to the directory after you apply the **replacement.acis.ldif** file.

See "Steps for Replacing ACIs" for instructions in using this LDIF file to clean up your ACIs.

replacement.acis.ldif File

```
dn: $rootSuffix
changetype: modify
replace: aci
aci: (targetattr = "*")(version 3.0; acl "Configuration Administrator";
   allow (all)
   userdn="ldap:///uid=admin,ou=Administrators,ou=TopologyManagement,
o=NetscapeRoot";)
aci: (target="ldap:///$rootSuffix")
  (targetfilter=(!(objectclass=sunServiceComponent)))
  (targetattr != "userPassword||passwordHistory
   ||passwordExpirationTime||passwordExpWarned||passwordRetryCount
  ||retryCountResetTime||accountUnlockTime||passwordAllowChangeTime")
  (version 3.0; acl "anonymous access rights";
   allow (read,search,compare)
   userdn = "ldap:///anyone"; )
aci: (targetattr != "nsroledn||aci||nsLookThroughLimit||nsSizeLimit
  ||nsTimeLimit||nsIdleTimeout||passwordPolicySubentry||passwordExpiration
    Time
  ||passwordExpWarned||passwordRetryCount||retryCountResetTime
  ||accountUnlockTime||passwordHistory||passwordAllowChangeTime||uid||mem
    berOf
  ||objectclass||inetuserstatus||ou||owner||mail||mailuserstatus
  ||memberOfManagedGroup||mailQuota||mailMsgQuota||mailhost
  ||mailAllowedServiceAccess||inetCOS||mailSMTPSubmitChannel")
  (version 3.0; acl "Allow self entry modification";
```

```
  allow (write)
  userdn ="ldap:///self";)
aci: (targetattr != " aci || nsLookThroughLimit || nsSizeLimit
  || nsTimeLimit|| nsIdleTimeout")
  (version 3.0; acl "Allow self entry read search";
  allow(write)
  userdn ="ldap:///self";)
aci: (target="ldap:///$rootSuffix")
  (targetattr="*")
  (version 3.0; acl "S1IS Proxy user rights";
  allow (proxy)
  userdn = "ldap:///cn=puser,ou=DSAME Users,
  $rootSuffix"; )
aci: (target="ldap:///$rootSuffix")
  (targetattr="*")
  (version 3.0; acl "S1IS special dsame user rights for all under the root
   suffix";
  allow (all)
  userdn = "ldap:///cn=dsameuser,ou=DSAME Users,
  $rootSuffix"; )
aci: (target="ldap:///$rootSuffix")
  (targetattr="*")
  (version 3.0; acl "S1IS special ldap auth user rights";
  allow (read,search)
  userdn = "ldap:///cn=amldapuser,ou=DSAME Users,
  $rootSuffix"; )
aci: (target="ldap:///$rootSuffix")
  (targetattr="*")
  (version 3.0; acl "S1IS Top-level admin rights";
  allow (all)
  roledn = "ldap:///cn=Top-level Admin Role,
  $rootSuffix"; )
aci: (targetattr="*")
  (version 3.0; acl "Messaging Server End User Administrator Read Only
   Access";
  allow (read,search)
  groupdn="ldap:///cn=Messaging End User Administrators Group,ou=Groups,
  $rootSuffix";)
aci: (targetattr="objectclass || mailalternateaddress || Mailautoreplymode
   || mailprogramdeliveryinfo || preferredlanguage || maildeliveryoption
   || mailforwardingaddress || mailAutoReplyTimeout
   || mailautoreplytextinternal
   || mailautoreplytext || vacationEndDate || vacationStartDate
   || mailautoreplysubject || maxPabEntries || mailMessageStore
   || mailSieveRuleSource || sunUCDateFormat || sunUCDateDeLimiter
   || sunUCTimeFormat || mailuserstatus || maildomainstatus
   || nswmextendeduserprefs || pabURI")
  (version 3.0; acl "Messaging Server End User Administrator All Access";
  allow (all)
  groupdn = "ldap:///cn=Messaging End User Administrators Group,ou=Groups,
  $rootSuffix";)
aci: (targetattr = "*")
  (version 3.0;acl "Allow Read-Only Access";
  allow (read,search,compare)
  groupdn = "ldap:///cn=Read-Only,ou=Groups,
  $rootSuffix";)
aci: (target="ldap:///cn=Organization Admin Role,($dn),$rootSuffix")
  (targetattr="*")
  (version 3.0; acl "S1IS Organization Admin Role access deny";
  deny (write,add,delete,compare,proxy)
```

```
  roledn = "ldap:///cn=Organization Admin Role,($dn),
  $rootSuffix";)
aci: (target="ldap:///($dn),$rootSuffix")
  (targetattr="*")
  (version 3.0; acl "Organization Admin Role access allow read";
  allow(read,search)
  roledn = "ldap:///cn=Organization Admin Role,[$dn],
  $rootSuffix" ;)
aci: (target="ldap:///($dn),$rootSuffix")
  (targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
  (entrydn=($dn),$rootSuffix))))
  (targetattr = "*")
  (version 3.0; acl "S1IS Organization Admin Role access allow";
  allow (all)
  roledn = "ldap:///cn=Organization Admin Role,[$dn],
  $rootSuffix";)
```

Steps for Replacing ACIs

This section contains information that you need before you begin and instructions for replacing the ACIs.

Before You Begin

Before you begin this procedure, examine the existing ACIs in your directory. You should determine whether you might need to keep any ACIs that would be deleted by the procedure.

This procedure will initially remove *all* ACIs from the root suffix and then replace them with the set of ACIs listed below. If the directory contains ACIs generated by applications other than Messaging Server, you should save those ACIs to a file and reapply them to the directory after you apply the **replacement.acis.ldif** file.

To help you analyze existing ACIs generated by Access Manager and Messaging Server, see the following sections later in this chapter:

- Analysis of the Existing ACIs
- Analysis of How ACIs Are Consolidated
- List of Unused ACIs to be Discarded

Replacing ACIs

The following procedure describes how to consolidate ACIS in the root suffix and remove unused ACIs.

To replace ACIs:

1. Save your existing ACIs currently on the root suffix.

   You can use the **ldapsearch** command, as in the following example:

   ```
   ldapsearch -D "cn=Directory Manager" -w password -s base -b $rootSuffix aci=*
   aci >filename
   ```

   where

   - *password* is the password of the Directory Server administrator.
   - *$rootSuffix* is your root suffix, such as **o=usergroup**.
   - *filename* is the name of the file into which the saved ACIs will be written.

2. Copy and rename the **replacement.acis.ldif** file.

   When you install Delegated Administrator, the **replacement.acis.ldif** file is installed in the following directory:

   ```
   DelegatedAdmin_home/lib/config-templates
   ```

3. Edit the **$rootSuffix** entries in your copy of the **replacement.acis.ldif** file.

   Change the root suffix parameter, $rootSuffix, to your root suffix (such as **o=usergroup**). The $rootSuffix parameter appears multiple times in the LDIF file; each instance must be replaced.

4. Use the LDAP directory tool **ldapmodify** to replace the ACIs.

   For example, you could run the following command:

   ```
   ldapmodify -D directory manager -w password-f replacement.acis.finished.ldif
   ```

   where

   - *directory manager* is the name of the Directory Server administrator.
   - *password* is the password of the Directory Service administrator.
   - *replacement.acis.finished.ldif* is the name of the edited LDIF file that consolidates and removes ACIs in the directory.

Eliminating Dynamic Organization ACIs

When you use the Delegated Administrator console to create an organization, a group of ACIs is created on the organization node.

The replacement ACIs installed in the preceding procedure eliminate the need for these per-organization ACIs. You can prevent the creation of the per-organization ACIs by using the Access Manager console.

To eliminate dynamic organization ACIs:

1. Log in to the Access Manager console as **amadmin**.

   The AM console is located at the following url:

   ```
   http://machine name:port/amconsole
   ```

   where

   - *machine name* is machine where Access Manager is running
   - *port* is the port

2. Select the **Service Configuration** tab.

   By default, the Administration configuration page is displayed.

3. In the right side of the console, scroll down to **Dynamic Administrative Role ACIs**.

4. Select and delete all ACIs in the text box for **Dynamic Administrative Role ACIs**.

5. Save the edited settings.

Analysis of the Existing ACIs

The listing in this section shows the ACIs installed in the directory when you install Access Manager and Messaging Server. It also describes the function of each ACI and recommends whether an ACI can be retained, consolidated, or discarded.

The ACIs are divided into the following categories:

Root Suffix

```
dn: $rootSuffix
#
# consolidate
#
aci:
(targetattr != "nsroledn || aci || nsLookThroughLimit || nsSizeLimit ||
nsTimeLimit || nsIdleTimeout || passwordPolicySubentry
|| passwordExpirationTime
|| passwordExpWarned || passwordRetryCount || retryCountResetTime
|| accountUnlockTime || passwordHistory || passwordAllowChangeTime")
(version 3.0; acl "Allow self entry modification except for nsroledn, aci,
resource limit attributes, passwordPolicySubentry and password policy state
attributes";
allow (write)
userdn ="ldap:///self";)
```

Action: Consolidate.

There is no requirement for self access to this suffix. This ACI is duplicated; it can be incorporated into the self ACIs on the root suffix.

```
#
# retain
#
aci:
(targetattr = "*")
(version 3.0; acl "Configuration Administrator";
allow (all)
userdn = "ldap:///uid=admin, ou=Administrators,
ou=TopologyManagement,o=NetscapeRoot";)
```

Action: Retain.

This is the admin user who would authenticate with Pass-Through Authentication to the **slapd-config** instance. If all configuration is to be performed as Directory Manager, using comm and line utilities, this ACI is not required. On the chance that someone needs to authenticate to the console as this user, this ACI can be kept here. Similar ACIs can be removed.

```
#
# discard
#
aci:
(targetattr =”*”)
(version 3.0;acl “Configuration Administrators Group”;
allow (all)
(groupdn = “ldap:///cn=Configuration Administrators, ou=Groups,
ou=TopologyManagement, o=NetscapeRoot”);)
```

Action: Discard on all DB back-ends.

This is the Configuration Administrators group that would have privileges if the console were being used to delegate server-administration privileges.

```
#
# discard
#
aci:
(targetattr =”*”)
(version 3.0;acl “Directory Administrators Group”;
allow (all)
(groupdn = “ldap:///cn=Directory Administrators, $rootSuffix”);)
```

Action: Discard on all DB back-ends.

This is the general Directory Administrators group privilege definition.

```
#
# discard
#
aci:
(targetattr = “*”)
(version 3.0; acl “SIE Group”;
allow (all)
groupdn = “ldap:///cn=slapd-whater, cn=Sun ONE Directory Server,
cn=Server Group, cn=whater.red.iplanet.com, ou=red.iplanet.com,
o=NetscapeRoot”;)
```

Action: Discard on all DB back-ends.

This is a Console/Administration server-related group privilege definition.

Access Manager

```
# retain
#
aci:
(target=”ldap:///$rootSuffix”)
(targetattr=”*”)
(version 3.0; acl “S1IS Proxy user rights”;
allow (proxy)
userdn = “ldap:///cn=puser,ou=DSAME Users,$rootSuffix”; )
```

Action: Retain.

This ACI grants access to a system user for Access Manager.

```
#
# retain
#
aci:
 (target="ldap:///$rootSuffix")
(targetattr="*")
(version 3.0; acl "S1IS special dsame user rights for all under the
root suffix";
allow (all)
userdn = "ldap:///cn=dsameuser,ou=DSAME Users,$rootSuffix"; )
```

Action: Retain.

This ACI grants access to a system user for Access Manager.

```
#
# retain
#
aci:
(target="ldap:///$rootSuffix")(targetattr="*")|
(version 3.0;acl "S1IS special ldap auth user rights";
allow (read,search)
userdn = "ldap:///cn=amldapuser,ou=DSAME Users,$rootSuffix"; )
```

Action: Retain.

This ACI grants access to a system user for Access Manager.

```
#
# discard
#
aci:
(target="ldap:///cn=amldapuser,ou=DSAME Users,$rootSuffix")
(targetattr = "*")
(version 3.0;
acl "S1IS special ldap auth user modify right";
deny (write)
roledn != "ldap:///cn=Top-level Admin Role,$rootSuffix";)
```

Action: Discard.

This ACI prevents the Top-Level Administrator (TLA) from modifying the **amldapuser** account.

```
#
# retain
#
aci:
(target="ldap:///$rootSuffix")
(targetattr="*")
(version 3.0; acl "S1IS Top-level admin rights";
allow (all)
roledn = "ldap:///cn=Top-level Admin Role,$rootSuffix"; )
```

Action: Retain.

This ACI grants access to the Top-Level Administrator role.

```
#
# discard
```

```
#
aci:
(targetattr="iplanet-am-saml-user || iplanet-am-saml-password")
(targetfilter="(objectclass=iplanet-am-saml-service)")
(version 3.0; acl "S1IS Right to modify saml user and password";
deny (all)
(roledn != "ldap:///cn=Top-level Admin Role,$rootSuffix")
AND (userdn != "ldap:///cn=dsameuser,ou=DSAME Users,$rootSuffix")
AND (userdn != "ldap:///cn=puser,ou=DSAME Users,$rootSuffix"); )
```

Action: Discard.

This ACI protects SAML-related attributes.

Top-level Help Desk Admin Role

```
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(!(nsroledn=cn=Top-level Admin Role,$rootSuffix)))
(targetattr = "*")
(version 3.0; acl "S1IS Top-level Help Desk Admin Role access allow";
allow (read,search)
roledn = "ldap:///cn=Top-level Help Desk Admin Role,$rootSuffix";)
```

Action: Discard.

```
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(!(nsroledn=cn=Top-level Admin Role,$rootSuffix)))
(targetattr = "userPassword")
(version 3.0; acl "S1IS Top-level Help Desk Admin Role access allow";
allow (write)
roledn = "ldap:///cn=Top-level Help Desk Admin Role,$rootSuffix";)
```

Action: Discard.

Top-level Policy Admin Role

```
#
# discard
#
aci:
target="ldap:///$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix))))
(targetattr = "*")
(version 3.0; acl "S1IS Top-level Policy Admin Role access allow";
allow (read,search)
roledn = "ldap:///cn=Top-level Policy Admin Role,$rootSuffix";)
```

Action: Discard.

This ACI pertains to the Top-level Policy Admin role.

```
#
# discard
```

```
#
aci:
(target="ldap:///ou=iPlanetAMAuthService,ou=services,*$rootSuffix")
(targetattr = "*")
(version 3.0; acl "S1IS Top-level Policy Admin Role access Auth Service
deny";
deny (add,write,delete)
roledn = "ldap:///cn=Top-level Policy Admin Role,$rootSuffix";)
```

Action: Discard.

This ACI pertains to the Top-level Policy Admin role.

```
#
# discard
#
aci:
(target="ldap:///ou=services,*$rootSuffix")
(targetattr = "*")
(version 3.0; acl "S1IS Top-level Policy Admin Role access allow";
allow (all)
roledn = "ldap:///cn=Top-level Policy Admin Role,$rootSuffix";)
```

Action: Discard.

This ACI pertains to the Top-level Policy Admin role.

```
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter="(objectclass=sunismanagedorganization)")
(targetattr = "sunRegisteredServiceName")
(version 3.0; acl "S1IS Top-level Policy Admin Role access allow";
allow (read,write,search)
roledn = "ldap:///cn=Top-level Policy Admin Role,$rootSuffix";)
```

Action: Discard.

This ACI pertains to the Top-level Policy Admin role.

AM Self

```
#
# consolidate
#
aci:
(targetattr = "*")
(version 3.0;
acl "S1IS Deny deleting self";
deny (delete)
userdn ="ldap:///self";)
```

Action: Consolidate into a single self-write ACI. The explicit deny is not required, since end users do not have permission to delete any entry, including themselves.

This is one of several ACIs that set self-privileges. The explicit deny prevents any entry from deleting itself.

```
#
# consolidate
```

```
#
aci:
(targetattr = "objectclass || inetuserstatus
|| iplanet-am-user-login-status
|| iplanet-am-web-agent-access-allow-list
|| iplanet-am-domain-url-access-allow
|| iplanet-am-web-agent-access-deny-list || iplanet-am-user-account-life
|| iplanet-am-session-max-session-time || iplanet-am-session-max-idle-time
|| iplanet-am-session-get-valid-sessions
|| iplanet-am-session-destroy-sessions
|| iplanet-am-session-add-session-listener-on-all-sessions
|| iplanet-am-user-admin-start-dn
|| iplanet-am-auth-post-login-process-class")
(targetfilter=(!(nsroledn=cn=Top-level Admin Role,$rootSuffix)))
(version 3.0; acl "S1IS User status self modification denied";
deny (write)
userdn ="ldap:///self";)
```

Action: Consolidate into a single self-write ACI.

This is one of several ACIs that set self-write privileges.

```
#
# consolidate
#
aci:
(targetattr != "iplanet-am-static-group-dn || uid || nsroledn || aci
|| nsLookThroughLimit || nsSizeLimit || nsTimeLimit || nsIdleTimeout
|| memberOf || iplanet-am-web-agent-access-allow-list
|| iplanet-am-domain-url-access-allow
|| iplanet-am-web-agent-access-deny-list")
(version 3.0; acl "S1IS Allow self entry modification except for nsroledn,
aci, and resource limit attributes";
allow (write)
userdn ="ldap:///self";)
```

Action: Consolidate into a single self-write ACI.

This is one of several ACIs that set privileges.

```
#
# consolidate
#
aci:
(targetattr != "aci || nsLookThroughLimit || nsSizeLimit || nsTimeLimit
|| nsIdleTimeout || iplanet-am-domain-url-access-allow")
(version 3.0; acl "S1IS Allow self entry read search except for nsroledn,
aci, resource limit and web agent policy attributes";
allow (read,search)
userdn ="ldap:///self";)
```

Action: Consolidate into a single self-write ACI.

This is one of several ACIs that set self-write privileges.

AM Anonymous

```
#
# consolidate
#
aci:
(target="ldap:///ou=services,$rootSuffix")
```

```
(targetfilter=(!(objectclass=sunServiceComponent)))
(targetattr = "*")
(version 3.0; acl "S1IS Services anonymous access";
allow (read, search, compare)
userdn = "ldap:///anyone";)
```

Action: Consolidate into a single anonymous ACI.

This is one of several ACIs that grant anonymous privileges.

```
#
# consolidate
#
aci:
(target="ldap:///ou=iPlanetAMAdminConsoleService,*,$rootSuffix")
(targetattr = "*")
(version 3.0; acl "S1IS iPlanetAMAdminConsoleService anonymous access";
allow (read, search, compare)
userdn = "ldap:///anyone";)
```

Action: Consolidate into a single anonymous ACI.

This is one of several ACIs that grant anonymous privileges.

```
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(entrydn=$rootSuffix))
(targetattr="*")
(version 3.0; acl "S1IS Default Organization delete right denied";
deny (delete)
userdn = "ldap:///anyone"; )
```

Action: Discard.

This ACI prevents any user (other than the **rootdn**) from deleting the default organization.

```
#
# discard
#
aci:
(target="ldap:///cn=Top-level Admin Role,$rootSuffix")
(targetattr="*")
(version 3.0; acl "S1IS Top-level admin delete right denied";
deny(delete)
userdn = "ldap:///anyone"; )
```

Action: Discard.

This ACI prevents any user (other than the **rootdn**) from deleting the Top-Level Administrator role.

AM Deny Write Access

```
#
# discard
#
aci: (targetattr = "*")
(version 3.0; acl "S1IS Deny write to anonymous user";
```

```
deny (add,write,delete)
roledn =”ldap:///cn=Deny Write Access,$rootSuffix”;)
```

Action: Discard.

This ACI pertains to the Deny Write Access Role.

AM Container Admin Role

```
#
# discard
#
aci:
(target=”ldap:///($dn),$rootSuffix”)
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix))))
(targetattr != “nsroledn”)
(version 3.0; acl “S1IS Container Admin Role access allow”;
allow (all)
roledn = “ldap:///cn=Container Admin Role,[$dn],$rootSuffix”;)
```

Action: Discard.

This ACI pertains to the Container Admin Role.

```
#
# discard
#
aci:
(target=”ldap:///cn=Container Admin Role,($dn),$rootSuffix”)
(targetattr=”*”)
(version 3.0; acl “S1IS Container Admin Role access deny”;
deny (write,add,delete,compare,proxy)
roledn = “ldap:///cn=Container Admin Role,($dn),$rootSuffix”;)
```

Action: Discard.

This ACI pertains to the Container Admin Role.

```
#
# discard
#
aci:
 (target=”ldap:///ou=People,$rootSuffix”)
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix)
(nsroledn=cn=Organization Admin Role,$rootSuffix)
(nsroledn=cn=Container Admin Role,$rootSuffix))))
(targetattr != “iplanet-am-web-agent-access-allow-list
|| iplanet-am-domain-url-access-allow
|| iplanet-am-web-agent-access-deny-list || nsroledn”)
(version 3.0; acl “S1IS Group and people container admin role”;
allow (all)
roledn = “ldap:///cn=ou=People_dc=red_dc=iplanet_dc=com,$rootSuffix”;)
```

Action: Discard.

This ACI pertains to the Group and People Container Admin Role.

Organization Help Desk

```
#
# discard
#
aci: (extra verses dreambig)
(target="ldap:///$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix)
(nsroledn=cn=Organization Admin Role,$rootSuffix))))
(targetattr = "*")
(version 3.0; acl "S1IS Organization Help Desk Admin Role access allow";
allow (read,search)
roledn = "ldap:///cn=Organization Help Desk Admin Role,$rootSuffix";)
```

Action: Discard.

This ACI pertains to the Organization Help Desk Admin Role.

```
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix)
(nsroledn=cn=Organization Admin Role,$rootSuffix))))
(targetattr = "userPassword")
(version 3.0; acl "S1IS Organization Help Desk Admin Role access allow";
allow (write)
roledn = "ldap:///cn=Organization Help Desk Admin Role,$rootSuffix";)
```

Action: Discard.

This ACI pertains to the Organization Help Desk Admin Role.

AM Organization Admin Role

```
#
# consolidate
#
aci: (different name - "allow all" instead of "allow")
(target="ldap:///($dn),$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix))))
(targetattr != "nsroledn")
(version 3.0; acl "S1IS Organization Admin Role access allow all";
allow (all)
roledn ="ldap:///cn=Organization Admin Role,[$dn],$rootSuffix";)
```

Action: Consolidate.

```
#
# consolidate
#
aci:
(target="ldap:///cn=Organization Admin Role,($dn),$rootSuffix")
(targetattr="*")
(version 3.0; acl "S1IS Organization Admin Role access deny";
```

```
deny (write,add,delete,compare,proxy)
roledn = "ldap:///cn=Organization Admin Role,($dn),$rootSuffix";)
```

Action: Consolidate.

This ACI pertains to the Organization Admin Role.

```
#
# consolidate
#
aci: (missing)
(target="ldap:///($dn),$rootSuffix")
(targetattr="*")
(version 3.0; acl "Organization Admin Role access allow read to org node";
allow (read,search)
roledn = "ldap:///cn=Organization Admin Role,($dn),$rootSuffix" ;)
```

Action: Consolidate.

This ACI pertains to the Organization Admin Role.

```
#
# consolidate
#
aci:
(target="ldap:///($dn),$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix))))
(targetattr != "nsroledn")
(version 3.0; acl "Organization Admin Role access allow";
allow (all)
roledn = "ldap:///cn=Organization Admin Role,[$dn],$rootSuffix";)
```

Action: Consolidate.

This ACI pertains to the Organization Admin Role.

```
#
# consolidate
#
aci:
(target="ldap:///($dn),$rootSuffix")
(targetattr!="businessCategory || description || facsimileTelephoneNumber
|| postalAddress || preferredLanguage || searchGuide || postOfficeBox ||
postalCode
|| registeredaddress || street || l || st || telephonenumber
||maildomainreportaddress
|| maildomainwelcomemessage || preferredlanguage || sunenablegab")
(version 3.0; acl "Organization Admin Role access deny to org node";
deny (write,add,delete)
roledn = "ldap:///cn=Organization Admin Role,($dn),$rootSuffix" ;)
```

Action: Consolidate.

This ACI pertains to the Organization Admin Role.

```
#
# consolidate
#
aci:
(target="ldap:///($dn),$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix))))
```

```
(targetattr != "nsroledn")
(version 3.0; acl "S1IS Organization Admin Role access allow all";
allow (all)
roledn = "ldap:///cn=Organization Admin Role,[$dn],$rootSuffix";)
```

Action: Consolidate.

AM Miscellaneous

```
#
# consolidate
#
aci:
(target="ldap:///$rootSuffix")
(targetattr="*")
(version 3.0; acl "Messaging Server End User Administrator Read
Access Rights -
product=SOMS,schema 2 support,class=installer,num=1,version=1";
allow (read,search)
groupdn="ldap:///cn=Messaging End User Administrators Group, ou=Groups,
$rootSuffix";)
```

Action: Discard.

Discarding this ACI disables the associated privileges to the attribute **iplanet-am-modifiable-by**.

Messaging Server

```
#
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetattr!="nsroledn")
(version 3.0; acl "S1IS Group admin's right to the users he creates";
allow (all)
userattr = "iplanet-am-modifiable-by#ROLEDN";)
```

Action: Consolidate.

This ACI grants permission to the Messaging End User Administrators Group.

```
#
# consolidate
#
aci:
(target="ldap:///$rootSuffix")
(targetattr="objectclass||mailalternateaddress||mailautoreplymode
||mailprogramdeliveryinfo||nswmextendeduserprefs||preferredlanguage
||maildeliveryoption||mailforwardingaddress
||mailAutoReplyTimeout||mailautoreplytextinternal||mailautoreplytext
||vacationEndDate||vacationStartDate||mailautoreplysubject||pabURI
||maxPabEntries||mailMessageStore||mailSieveRuleSource||sunUCDateFormat
||sunUCDateDeLimiter||sunUCTimeFormat")
(version 3.0; acl "Messaging Server End User Adminstrator Write
Access Rights -
product=SOMS,schema 2 support,class=installer,num=2,version=1";
allow (all)
groupdn="ldap:///cn=Messaging End User Administrators Group, ou=Groups,
```

```
$rootSuffix";)
```

Action: Consolidate.

This ACI grants permission to the Messaging End User Administrators Group.

```
#
# consolidate
#
aci:
(targetattr="uid||ou||owner||mail||mailAlternateAddress
||mailEquivalentAddress||memberOf
||inetuserstatus||mailuserstatus||memberOfManagedGroup||mailQuota
||mailMsgQuota||inetSubscriberAccountId||dataSource||mailhost
||mailAllowedServiceAcces||pabURI||inetCOS||mailSMTPSubmitChannel
||aci")
(targetfilter=(&(objectClass=inetMailUser)(!(nsroledn=cn=Organization
Admin Role,*))))
(version 3.0; acl "Deny write access to users over Messaging Server
protected attributes -
product=SOMS,schema 2 support,class=installer,num=3,version=1 ";
deny (write)
userdn = "ldap:///self";)
```

Action: Consolidate.

This is one of several ACIs that set self privileges.

Analysis of How ACIs Are Consolidated

The listing in this section shows the ACIs that have been consolidated in the replacement LDIF file, **replacement.acis.ldif**, which you can use to consolidate ACIs in the directory. See "Steps for Replacing ACIs" for instructions in how to replace ACIs.

The ACIs are divided into pairs. For each category, first the original ACIs and then the consolidated ACIs are listed:

Original Anonymous Access Rights

```
aci:
(targetattr != "userPassword || passwordHistory || passwordExpirationTime
|| passwordExpWarned || passwordRetryCount || retryCountResetTime ||
accountUnlockTime || passwordAllowChangeTime ")
(version 3.0; acl "Anonymous access";
allow (read, search, compare)
userdn = "ldap:///anyone";)
```

```
aci:
(target="ldap:///cn=Top-level Admin Role,$rootSuffix")
(targetattr="*")
version 3.0; acl "S1IS Top-level admin delete right denied";
deny (delete)
userdn = "ldap:///anyone"; )
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(entrydn=$rootSuffix))
(targetattr="*")
(version 3.0; acl "S1IS Default Organization delete right denied";
deny (delete)
userdn = "ldap:///anyone"; )
aci:
(target="ldap:///ou=services,$rootSuffix")
(targetfilter=(!(objectclass=sunServiceComponent)))
(targetattr = "*")
(version 3.0; acl "S1IS Services anonymous access";
allow (read, search, compare)
userdn = "ldap:///anyone";)
aci:
(target="ldap:///ou=iPlanetAMAdminConsoleService,*,$rootSuffix")
(targetattr = "*")
(version 3.0; acl "S1IS iPlanetAMAdminConsoleService anonymous access";
allow (read, search, compare)
userdn = "ldap:///anyone";)
```

Consolidated Anonymous Access Rights

```
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(!(objectclass=sunServiceComponent)))
(targetattr != "userPassword||passwordHistory
||passwordExpirationTime||passwordExpWarned||passwordRetryCount
||retryCountResetTime||accountUnlockTime||passwordAllowChangeTime")
(version 3.0; acl "anonymous access rights";
allow (read,search,compare)
userdn = "ldap:///anyone"; )
```

Analysis: This ACI, which is on the root, allows the same access as the original collection of anonymous ACIs. It does this by listing a set of excluded attributes. This replacement ACI improves performance by eliminating the (*) in the target.

Original Self ACIs

```
aci:
(targetattr != "nsroledn || aci || nsLookThroughLimit || nsSizeLimit ||
nsTimeLimit || nsIdleTimeout || passwordPolicySubentry ||
passwordExpirationTime
|| passwordExpWarned || passwordRetryCount || retryCountResetTime ||
accountUnlockTime || passwordHistory || passwordAllowChangeTime")
(version 3.0; acl "Allow self entry modification except for nsroledn, aci,
resource limit attributes, passwordPolicySubentry and password policy
state attributes";
allow (write)
userdn ="ldap:///self";)
aci:
```

```
(targetattr = "*")
(version 3.0; acl "S1IS Deny deleting self";
deny (delete)
userdn ="ldap:///self";)
aci:
(targetattr = "objectclass || inetuserstatus ||
planet-am-web-agent-access-allow-list
|| iplanet-am-domain-url-access-allow
|| iplanet-am-web-agent-access-deny-list
|| iplanet-am-user-account-life || iplanet-am-session-max-session-time
|| iplanet-am-session-max-idle-time
|| iplanet-am-session-get-valid-sessions
|| iplanet-am-session-destroy-sessions
|| iplanet-am-session-add-session-listener-on-all-sessions
|| iplanet-am-user-admin-start-dn
|| iplanet-am-auth-post-login-process-class")
(targetfilter=(!(nsroledn=cn=Top-levelAdmin Role,$rootSuffix)))
(version 3.0; acl "S1IS User status self modification denied";
deny (write)
userdn ="ldap:///self";)
aci:
(targetattr != "iplanet-am-static-group-dn || uid || nsroledn || aci
|| LookThroughLimit
|| nsSizeLimit || nsTimeLimit || nsIdleTimeout || memberOf ||
planet-am-web-agent-access-allow-list
|| iplanet-am-domain-url-access-allow ||
planet-am-web-agent-access-deny-list")
(version 3.0; acl "S1IS Allow self entry modification except
for nsroledn, aci, and resource limit attributes";
allow (write)
userdn ="ldap:///self";)
aci:
(targetattr != "aci || nsLookThroughLimit || nsSizeLimit || nsTimeLimit
|| nsIdleTimeout || iplanet-am-domain-url-access-allow")
(version 3.0; acl "S1IS Allow self entry read search except for
nsroledn, aci, resource limit and web agent policy attributes";
allow (read,search)
userdn ="ldap:///self";)
aci:
(targetattr="uid||ou||owner||mail||mailAlternateAddress
||mailEquivalentaddress||memberOf
||inetuserstatus||mailuserstatus||memberOfManagedGroup||mailQuota
||mailMsgQuota
||inetSubscriberAccountId||dataSource||mailhost||mailAllowedServiceAccess
||pabURI||inetCOS||mailSMTPSubmitChannel||aci")
(targetfilter=(&(objectClass=inetMailUser)(!(nsroledn=cn=Organization Admin
role,*))))
(version 3.0; acl "Deny write access to users over Messaging Server
protected attributes -
product=SOMS,schema 2 support,class=installer,num=3,version=1 ";
deny (write)
userdn = "ldap:///self";)
```

Consolidated Self ACIs

```
aci:
(targetattr != "nsroledn || aci || nsLookThroughLimit || nsSizeLimit
|| nsTimeLimit || nsIdleTimeout || passwordPolicySubentry ||
asswordExpirationTime
```

```
|| passwordExpWarned || passwordRetryCount || retryCountResetTime
|| accountUnlockTime || passwordHistory || passwordAllowChangeTime ||
id || memberOf
|| objectclass || inetuserstatus || ou || owner || mail || mailuserstatus
|| memberOfManagedGroup ||mailQuota || mailMsgQuota || mailhost
|| mailAllowedServiceAccess || inetCOS || mailSMTPSubmitChannel")
(version 3.0; acl "Allow self entry modification";
allow (write)
userdn ="ldap:///self";)
aci:
(targetattr != " aci || nsLookThroughLimit || nsSizeLimit
|| nsTimeLimit|| nsIdleTimeout")
(version 3.0; acl "Allow self entry read search";
allow(read,search)
userdn ="ldap:///self";)
```

Analysis: Missing all the **iplanet-am-*** attributes. Since **deny** is the default if an ACI is not present, all **deny** ACIs are removed. The ones that allow **write** are consolidated into a single ACI.

Original Messaging Server ACIs

```
aci:
(target="ldap:///$rootSuffix")
(targetattr="*")
(version 3.0; acl "Messaging Server End User Administrator Read
Access Rights -
product=SOMS,schema 2 support,class=installer,num=1,version=1";
allow (read,search)
groupdn="ldap:///cn=Messaging End User Administrators Group, ou=Groups,
rootSuffix";)
aci:
(target="ldap:///$rootSuffix")
(targetattr="objectclass||mailalternateaddress||mailautoreplymode||
mailprogramdeliveryinfo
||nswmextendeduserprefs||preferredlanguage||maildeliveryoption||
mailforwardingaddress
||mailAutoReplyTimeout||mailautoreplytextinternal||mailautoreplytext||
vacationEndDate
||vacationStartDate||mailautoreplysubject||pabURI||maxPabEntries||
mailMessageStore
||mailSieveRuleSource||sunUCDateFormat||sunUCDateDeLimiter||
sunUCTimeFormat")
(version 3.0; acl "Messaging Server End User Adminstrator Write
Access Rights -
product=SOMS,schema 2 support,class=installer,num=2,version=1";
allow (all)
groupdn="ldap:///cn=Messaging End User Administrators Group, ou=Groups,
rootSuffix";)
aci:
(targetattr="uid||ou||owner||mail||mailAlternateAddress||
mailEquivalentAddress||memberOf
||inetuserstatus||mailuserstatus||memberOfManagedGroup||mailQuota||
mailMsgQuota
||inetSubscriberAccountId||dataSource||mailhost||mailAllowedServiceAccess
||pabURI||inetCOS||mailSMTPSubmitChannel||aci")
(targetfilter=(&(objectClass=inetMailUser)(!(nsroledn=cn=Organization Admin
Role,*))))
(version 3.0; acl "Deny write access to users over Messaging Server
protected attributes -
```

```
product=SOMS,schema 2 support,class=installer,num=3,version=1 ";
deny (write)
userdn = "ldap:///self";)
```

Consolidated Messaging Server ACIs

The self ACI is handled in the self ACIs.

```
aci:
(targetattr="*")
(version 3.0; acl "Messaging Server End User Administrator
Read Only Access";
allow (read,search)
groupdn = "ldap:///cn=Messaging End User Administrators
group,ou=Groups,$rootSuffix"; )
aci:
(targetattr="objectclass || mailalternateaddress || Mailautoreplymode
|| mailprogramdeliveryinfo || preferredlanguage || maildeliveryoption
|| mailforwardingaddress || mailAutoReplyTimeout
|| mailautoreplytextinternal
|| mailautoreplytext || vacationEndDate || vacationStartDate
|| mailautoreplysubject || maxPabEntries || mailMessageStore
|| mailSieveRuleSource || sunUCDateFormat || sunUCDateDeLimiter
|| sunUCTimeFormat || mailuserstatus || maildomainstatus
|| nswmextendeduserprefs || pabURI"
)
(version 3.0; acl "Messaging Server End User Administrator All Access";
allow (all)
groupdn = "ldap:///cn=Messaging End User Administrators
group,ou=Groups,$rootSuffix";)
```

Analysis: Same as the original ACIs.

Original Organization Admin ACIs

```
aci: (different name - "allow all" instead of "allow")
(target="ldap:///($dn),$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix))))
(targetattr != "nsroledn")
(version 3.0; acl "S1IS Organization Admin Role access allow all";
allow (all)
roledn ="ldap:///cn=Organization Admin Role,[$dn],$rootSuffix";)
aci: (missing)
(target="ldap:///($dn),$rootSuffix")
(targetattr="*")
(version 3.0; acl "Organization Admin Role access allow read to org node";
allow (read,search)
roledn = "ldap:///cn=Organization Admin Role,($dn),$rootSuffix" ;)
aci:
(target="ldap:///($dn),$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix))))
(targetattr != "nsroledn")
(version 3.0; acl "Organization Admin Role access allow";
allow (all)
roledn = "ldap:///cn=Organization Admin Role,[$dn],$rootSuffix";)
aci:
```

```
(target="ldap:///($dn),$rootSuffix")
(targetattr!="businessCategory || description || facsimileTelephoneNumber
|| postalAddress || preferredLanguage || searchGuide || postOfficeBox
|| postalCode
|| registeredaddress || street || l || st || telephonenumber
|| maildomainreportaddress
|| maildomainwelcomemessage || preferredlanguage || sunenablegab")
(version 3.0; acl "Organization Admin Role access deny to org node";
deny (write,add,delete)
roledn = "ldap:///cn=Organization Admin Role,($dn),$rootSuffix" ;)
aci: (duplicate of per organization aci)
(target="ldap:///cn=Organization Admin Role,($dn),$rootSuffix")
(targetattr="*")
(version 3.0; acl "S1IS Organization Admin Role access deny";
deny (write,add,delete,compare,proxy)
roledn = "ldap:///cn=Organization Admin Role,($dn),$rootSuffix";)
aci:
(target="ldap:///cn=Organization Admin
Role,($dn),dc=red,dc=iplanet,dc=com")
(targetattr="*")
(version 3.0; acl "S1IS Organization Admin Role access deny";
deny (write,add,delete,compare,proxy)
roledn = "ldap:///cn=Organization Admin Role,($dn),$rootSuffix";)
aci:
(target="ldap:///o=fullOrg1,o=VIS,o=siroe.com,o=SharedDomainsRoot,
o=Business,rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,dc=red,dc=iplanet,dc=com))))
(targetattr = "nsroledn")
(targattrfilters="add=nsroledn:(nsroledn=*,o=fullOrg1,o=VIS,o=siroe.com,
o=SharedDomainsRoot,o=Business,$rootSuffix),
del=nsroledn:(nsroledn=*,o=fullOrg1,o=VIS,o=siroe.com,o=SharedDomainsRoot,
o=Business,$rootSuffix)")
(version 3.0;
acl "S1IS Organization Admin Role access allow";
allow (all)
roledn = "ldap:///cn=Organization Admin
Role,o=fullOrg1,o=VIS,o=siroe.com,o=SharedDomainsRoot,o=Business,
$rootSuffix";)
aci:
(target="ldap:///($dn),$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix))))
(targetattr != "nsroledn")
(version 3.0; acl "S1IS Organization Admin Role access allow all";
allow (all)
roledn = "ldap:///cn=Organization Admin
Role,[$dn],dc=red,dc=iplanet,dc=com";)
```

Consolidated Organization Admin ACIs

```
aci:
(target="ldap:///cn=Organization Admin Role,($dn),$rootSuffix")
(targetattr="*")
(version 3.0; acl "S1IS Organization Admin Role access deny";
deny (write,add,delete,compare,proxy)
roledn = "ldap:///cn=Organization Admin Role,($dn),$rootSuffix";)
aci:
(target="ldap:///($dn),$rootSuffix")
```

```
(targetattr="*")
(version 3.0; acl "Organization Admin Role access allow read";
allow(read,search)
roledn = "ldap:///cn=Organization Admin Role,[$dn],$rootSuffix" ;)
aci:
(target="ldap:///($dn),$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(entrydn=($dn),$rootSuffix))))
( targetattr = "*")
(version 3.0; acl "S1IS Organization Admin Role access allow";
allow (all)
roledn = "ldap:///cn=Organization Admin Role,[$dn],$rootSuffix";)
```

List of Unused ACIs to be Discarded

The listing in this section shows the unused default ACIs that are discarded from the directory when you apply the **replacement.acis.ldif** file to the directory.

The ACIs to be discarded are divided into the following categories:

- Suffix
- Top-level Help Desk Admin Role
- Top-level Policy Admin Role
- Access Manager Anonymous
- Access Manager Deny Write Access
- Access Manager Container Admin Role
- Organization Help Desk
- Access Manager Miscellaneous

Suffix

```
# discard
#
aci:
(targetattr ="*")
(version 3.0;acl "Configuration Administrators Group";
allow (all)
(groupdn = "ldap:///cn=Configuration Administrators, ou=Groups,
ou=TopologyManagement, o=NetscapeRoot");)
#
# discard
#
aci:
(targetattr ="*")
(version 3.0;acl "Directory Administrators Group";
allow (all)
(groupdn = "ldap:///cn=Directory Administrators, $rootSuffix");)
#
# discard
#
aci:
(targetattr = "*")
(version 3.0;
acl "SIE Group";
allow (all)
```

```
groupdn = "ldap:///cn=slapd-whater, cn=Sun ONE Directory Server, cn=Server
Group, cn=whater.red.iplanet.com, ou=red.iplanet.com, o=NetscapeRoot";)
#
# discard - prevents TLA from modifying the amldapuser account.
#
aci:
(target="ldap:///cn=amldapuser,ou=DSAME Users,$rootSuffix")
(targetattr = "*")
(version 3.0;
acl "S1IS special ldap auth user modify right";
deny (write)
roledn != "ldap:///cn=Top-level Admin Role,$rootSuffix";)
#
# discard - protects SAML related attributes
#
aci:
(targetattr="iplanet-am-saml-user || iplanet-am-saml-password")
(targetfilter="(objectclass=iplanet-am-saml-service)")
(version 3.0; acl "S1IS Right to modify saml user and password";
deny (all)
(roledn != "ldap:///cn=Top-level Admin Role,$rootSuffix")
AND (userdn != "ldap:///cn=dsameuser,ou=DSAME Users,$rootSuffix")
AND (userdn != "ldap:///cn=puser,ou=DSAME Users,$rootSuffix"); )
```

Top-level Help Desk Admin Role

```
#
# discard
 #
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(!(nsroledn=cn=Top-level Admin Role,$rootSuffix)))
(targetattr = "*")
(version 3.0; acl "S1IS Top-level Help Desk Admin Role access allow";
allow (read,search)
roledn = "ldap:///cn=Top-level Help Desk Admin Role,$rootSuffix";)
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(!(nsroledn=cn=Top-level Admin Role,$rootSuffix)))
(targetattr = "userPassword")
(version 3.0; acl "S1IS Top-level Help Desk Admin Role access allow";
allow (write)
roledn = "ldap:///cn=Top-level Help Desk Admin Role,$rootSuffix";)
```

Top-level Policy Admin Role

```
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix))))
(targetattr = "*")
(version 3.0; acl "S1IS Top-level Policy Admin Role access allow";
allow (read,search)
roledn = "ldap:///cn=Top-level Policy Admin Role,$rootSuffix";)
#
```

```
# discard
#
aci:
(target="ldap:///ou=iPlanetAMAuthService,ou=services,*$rootSuffix")
(targetattr = "*")
(version 3.0; acl "S1IS Top-level Policy Admin Role access
Auth Service deny";
deny (add,write,delete)
roledn = "ldap:///cn=Top-level Policy Admin Role,$rootSuffix";)
#
# discard
#
aci:
(target="ldap:///ou=services,*$rootSuffix")
(targetattr = "*")
(version 3.0; acl "S1IS Top-level Policy Admin Role access allow";
allow (all)
roledn = "ldap:///cn=Top-level Policy Admin Role,$rootSuffix";)
#
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter="(objectclass=sunismanagedorganization)")
(targetattr = "sunRegisteredServiceName")
(version 3.0; acl "S1IS Top-level Policy Admin Role access allow";
allow (read,write,search)
roledn = "ldap:///cn=Top-level Policy Admin Role,$rootSuffix";)
```

Access Manager Anonymous

```
#
# discard - prevents anyone other than rootdn from deleting
# default organization.
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(entrydn=$rootSuffix))
(targetattr="*")
(version 3.0; acl "S1IS Default Organization delete right denied";
deny (delete)
userdn = "ldap:///anyone"; )
#
# discard - prevents any user other than rootdn from deleting the
# TLA admin role.
#
aci:
(target="ldap:///cn=Top-level Admin Role,$rootSuffix")
(targetattr="*")
version 3.0; acl "S1IS Top-level admin delete right denied";
deny(delete)
userdn = "ldap:///anyone"; )
```

Access Manager Deny Write Access

```
#
# discard
#
aci:
(targetattr = "*")
```

```
(version 3.0; acl "S1IS Deny write to anonymous user";
deny (add,write,delete)
roledn ="ldap:///cn=Deny Write Access,$rootSuffix";)
```

Access Manager Container Admin Role

```
#
# discard
#
aci:
(target="ldap:///($dn),$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix))))
(targetattr != "nsroledn")
(version 3.0; acl "S1IS Container Admin Role access allow";
allow (all)
roledn = "ldap:///cn=Container Admin Role,[$dn],$rootSuffix";)
#
# discard
#
aci:
(target="ldap:///cn=Container Admin Role,($dn),$rootSuffix")
(targetattr="*")
(version 3.0; acl "S1IS Container Admin Role access deny";
deny (write,add,delete,compare,proxy)
roledn = "ldap:///cn=Container Admin Role,($dn),$rootSuffix";)
#
# discard
#
aci:
(target="ldap:///ou=People,$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix)
(nsroledn=cn=Organization Admin Role,$rootSuffix)
(nsroledn=cn=Container Admin Role,$rootSuffix))))
(targetattr != "iplanet-am-web-agent-access-allow-list
|| iplanet-am-domain-url-access-allow
|| iplanet-am-web-agent-access-deny-list || nsroledn")
(version 3.0; acl "S1IS Group and people container admin role";
allow (all)
roledn = "ldap:///cn=ou=People_dc=red_dc=iplanet_dc=com,$rootSuffix";)
```

Organization Help Desk

```
#
# discard
#
aci: (extra verses dreambig)
(target="ldap:///$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix)
(nsroledn=cn=Organization Admin Role,$rootSuffix))))
(targetattr = "*")
(version 3.0; acl "S1IS Organization Help Desk Admin Role access allow";
allow (read,search)
roledn = "ldap:///cn=Organization Help Desk Admin Role,$rootSuffix";)
#
```

```
# discard
#
aci:
(target="ldap:///$rootSuffix")
(targetfilter=(!(|(nsroledn=cn=Top-level Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Help Desk Admin Role,$rootSuffix)
(nsroledn=cn=Top-level Policy Admin Role,$rootSuffix)
(nsroledn=cn=Organization Admin Role,$rootSuffix))))
(targetattr = "userPassword")
(version 3.0; acl "S1IS Organization Help Desk Admin Role access allow";
allow (write)
roledn = "ldap:///cn=Organization Help Desk Admin Role,$rootSuffix";)
```

Access Manager Miscellaneous

```
#
# discard - Removal disables the associated privileges to the attribute
# iplanetam-modifiable-by
#
aci:
(target="ldap:///$rootSuffix")
(targetattr!="nsroledn")
(version 3.0; acl "S1IS Group admin's right to the users he creates";
allow (all)
userattr = "iplanet-am-modifiable-by#ROLEDN";)
```

8

Customizing Delegated Administrator

After you have installed (with **commpkg install**) and configured Oracle Communications Delegated Administrator (with **config-commda**), you can customize your configuration to meet your particular needs.

This chapter describes how to add customized attributes to the Delegated Administrator console. Next, it shows how to customize additional Delegated Administrator features. Finally, it lists the locations of the configuration files and explains how to redeploy a customized file to the proper location.

You should back up any existing Delegated Administrator configuration file before you begin customizing it.

Also, customized configuration data can be lost when you upgrade Delegated Administrator. Therefore, you should preserve your customized configuration before you upgrade Delegated Administrator or rerun the Delegated Administrator configuration program. For more information, see *Delegated Administrator Installation and Configuration Guide*.

Customizing the Delegated Administrator Console

This section describes how to add custom attributes that can be provisioned in the Delegated Administrator console in Access Manager mode only. In Direct LDAP mode, all LDAP attributes are retrieved by default.

How Customization Works

You can customize the Delegated Administrator console to manage user attributes that are not already provided by the service packages and the Service sections (such as Mail Service or Calendar Service) of the User Properties page.

After you add the custom attributes, the Delegated Administrator console operates as follows:

- The console displays a UI field representing each custom LDAP attribute.

  When you create a user, the custom attributes appear in a page, called **Custom Data Section**, in the Create New User wizard. This page appears just before the **Summary** page.

  In addition, you can edit the values of these attributes for existing users. The custom attributes appear in a **Custom Data Section** in the **User Properties** page.

- You can enter a value in the UI field or check the box.
- When you click **Save** in the console, Delegated Administrator stores the updated values in the corresponding LDAP attributes in the directory.

If you write business logic and validation in the Java class that implements the custom attribute, that logic is performed before the values are saved in the directory.

You can customize attributes for users only.

Customization Tasks

To add custom attributes to the console, you need to perform the following tasks:

- Update your LDAP schema and add the object classes and attributes to the LDAP directory. Directory Server must recognize these attributes before Delegated Administrator can successfully write attribute values to the directory.

  You also may add attributes that already belong to your schema and are recognized by Directory Server (attributes provided by the Directory Preparation Tool, **comm_dssetup**). For example, you may want to add a mail attribute that is not automatically provisioned by Delegated Administrator. In this case, you do not need to update your schema.

- Create an XML customization file that identifies the LDAP attributes and defines the UI fields to be displayed in the Delegated Administrator console.

  See "Creating a Customization File" for detailed instructions.

- Edit the properties file that activates the customization and identifies the Java class.

  See "Editing the daconfig.properties file" for detailed instructions.

- (Optional) Create a Java class that enables Delegated Administrator to respond to user input for the attributes. The customization subsystem calls the business logic and validation you write for the attributes before values are written to the LDAP directory.

  See "Creating a Java Class for the Custom Attributes" for detailed instructions.

You do not need to perform these tasks in any particular sequence.

However, after you complete these tasks, you must deploy your updated configuration files to the Web container used by the Delegated Administrator console. See "Creating a Java Class for the Custom Attributes" for a description of the deploy step.

- (In Access Manager mode) Edit the Delegated Administrator server **resource.properties** file to add the custom attribute in the section **userreturnattr-xx** and follow the steps mentioned in the section "Deploying a Customized Configuration File" to deploy the changes.

Creating a Customization File

See "Customization File Details" for information about the customization file, including guidelines for creating the file, descriptions of the XML elements, and an example.

1. Open the sample XML customization file, **sampleCustom.xml**, in a text editor.

   The **sampleCustom.xml** file is located by default in the following directory:

   DelegatedAdmin_home/**data/da/jsp/users**

2. Remove the comment markers from the XML entries.

   The **sampleCustom.xml** file shows the DTD definition and a skeleton sample:

```
<?xml version="1.0" encoding="UTF--8"?>
<!----DTD DEFINITION: PLEASE DO NOT EDIT THE FOLLOWING LINES---->
<!DOCTYPE ;property--list [
<!ELEMENT property--list (property)+>
<!ELEMENT>
<!ATTLIST property
label CDATA #REQUIRED
CDATA #REQUIRED
uifield (text | textarea | checkbox) #IMPLIED
syntax (string \| number) #IMPLIED>
]>
<!---- END DEFINITION
---->
<!---- Pls uncomment and edit below for customization
<property--list>
<property label="Customized label1" attr="custattr1" >
<property label="rsrc.key.label2" attr="custattr2" uifield="text"& >
</property--list>
---->
```

See "Sample Customization File" for an example of a customization file containing three attributes.

3. Edit the section of the file that starts with the *property-list* element.

   Add the XML elements to define the custom attributes. Add one entry, beginning with the **property** element, to define each attribute.

   See "Guidelines for Creating a Customization File" for instructions on how to define the attributes and their corresponding UI fields in the console.

   See "XML Elements Used in a Customization File" for definitions of the XML elements.

4. (Optional) Rename the customization file to a name suitable for your installation.

5. Save the customization file in the following directory:

   *DelegatedAdmin_home***/data/da/jsp/users**

   The customization file must be in this location.

Editing the daconfig.properties file

This task explains how to activate the customization and identify the Java class.

1. Open the **daconfig.properties** file in a text editor.

   The **daconfig.properties** file is located by default in the following directory:

   *DelegatedAdmin_home***/data/da/WEB-INF/classes/com/sun/comm/da/resources**

2. Find the following properties in the file:

   ```
   users.custom.plugin.exist=false
   users.custom.plugin.class=com.sun.comm.da.common.ReferenceCustomUIPluginImpl
   users.custom.plugin.separator=, \n\r
   ```

3. Change the value of the **users.custom.plugin.exist** property from **false** to **true**.

   By default, this value is **false**.

   For example:

   ```
   users.custom.plugin.exist=true
   ```

Setting this value to true activates the task of customization in Delegated Administrator (which includes enabling the customization plug-in).

Even if you do not write a Java class for customized business logic (as described in "Creating a Java Class for the Custom Attributes"), you must set the **users.custom.plugin.exist** property to **true** to enable the customization to work.

4. (Optional) Modify the Java class name for the customization plug-in.

   By default, the fully qualified Java class name is:

   ```
   com.sun.comm.da.common.ReferenceCustomUIPluginImpl
   ```

   To modify the Java class name, edit the name in the following property:

   ```
   users.custom.plugin.class=
   com.sun.comm.da.common.ReferenceCustomUIPluginImpl
   ```

   If you want to use the default name, you do not have to modify this property.

5. (Optional) Specify additional separators for multi-valued attributes.

   If you add a multi-valued LDAP attribute, line separators are needed to distinguish multiple entries for attribute. By default, four common separators are configured in the following property:

   ```
   users.custom.plugin.separator=, \n\r
   ```

   To add to or modify the separators, edit the values in this property.

Creating a Java Class for the Custom Attributes

This task explains how to write a Java class that implements the Java interface for adding business logic to handle the custom attributes.

This task is optional. If you do not change the default Java class name and do not need to add business logic and validation for the custom attributes, you do not need to perform this task.

1. Create the Java class file.

   The class implements the following two methods:

   - **public String getCustomFilename();**

     You should return the name of the XML customization file that you created in "Creating a Customization File" for the **getCustomFilename** method.

   - public String **performBusinessLogic(DAObjectContext ctx);**

     In the **performBusinessLogic** method, you can implement any business logic and validation you want to perform.

     The customization subsystem calls the **performBusinessLogic** method before it saves the custom attribute values in the LDAP directory. It passes the **DAObjectContext** object as a parameter. Developers can get information from this object through the following methods:

     ```
     public String getOrganizationDN();
     public DAUser getUserObject();
     ```

     The **DAUser** class is part of the Java Delegated Administrator API (JDAPI) library bundled with Delegated Administrator.

To see a simple example of how to implement this plug-in class, examine the default Java class file, named **ReferenceCustomUIPluginImpl** and located in the following path:

DelegatedAdmin_home/**WEB-INF/classes/**

This sample file is a fully working class for the default settings in the sample customization file, **sampleCustom.xml**. If you do not need to implement your own business logic and validation, you can use the sample Java class without modifying it. (Of course, if you change the name of the customization file, you also must change it in the **getCustomFilename** method in the Java class.)

2. Copy the Java class file to the following directory:

   DelegatedAdmin_home/**WEB-INF/classes/***Java class*

   where *Java class* is the fully qualified Java class name.

   The Java class file must be in this location.

3. Redeploy the edited files to the Web container used by the Delegated Administrator console.

   All the files you have created or updated in this task and in the preceding task, "Creating a Customization File", are located in the Delegated Administrator configuration path. These files include:

   - The XML customization file
   - The **daconfig.properties** file
   - Java class file

   Before the changes you have made can take effect, you must run the script that deploys the customized files to your Web container.

   See "Deploying a Customized Configuration File" for instructions on how to perform this task.

Customization File Details

The customization file provides the Delegated Administrator console with the information it needs to do the following:

- Display a field in the Delegated Administrator console that represents the LDAP attribute in the directory
- Accept user input in a text box, text area, or check box in the console
- Save the modified input as the value of the LDAP attribute in the directory

The customization file is defined in XML format.

Guidelines for Creating a Customization File

When you create a customization file, follow these rules and guidelines:

- You can define multiple attributes in a customization file.
- Each attribute is defined on a separate line.
- One element, **property**, identifies each attribute defined in the file.

  The **property** element appears at the beginning of each attribute definition (each line in the file). If you define multiple attributes, the **property** element appears repeatedly in the file.

- For each attribute, you must define a label to be displayed in the Delegated Administrator console, that describes the attribute.
- For each attribute, you must provide the name of the LDAP attribute whose value the **console** field represents.

XML Elements Used in a Customization File

Table 8–1 describes the XML elements in the customization file.

Table 8–1 Elements used to define a Delegated Administrator console field derived from an LDAP attribute

Element	Definition
property	Represents one UI field. Must be placed at the beginning of each line that defines an attribute. If multiple attributes are defined, this element appears multiple times in the customization file.
label	Specifies the display text that appears as the label in the Delegated Administrator console. This element is required. It must be used with each instance of the XML element **property**.
attr	Specifies the name of the LDAP attribute that will be represented by the display field. This element is required. By default, the LDAP attribute is assumed to be single-valued. To define a multi-valued LDAP attribute, add the prefix **multi** to the definition of the **attr** element. For example, to define a multi-valued attribute named **custldapttr**, enter the following: **element:attr="multi:custldapattr"**
uifield	Determines the type of field to be displayed. This field accepts user-supplied values for the LDAP attribute defined with **attr**. You can specify one of these types of fields: ■ **text** is a single-line text box ■ **textarea** is a multiline text box. ■ **checkbox** is a check box.The **uifield** element is optional. If it is not specified, it defaults to a **text** field.
syntax	Specifies the type of value permitted for user input. You can choose one of these syntax values: **string** or **number**. If the user enters an invalid value in the display field (for example, an alphanumeric character in a field defined as a **number**), Delegated Administrator does not save the value in the directory and displays an error message in the console.

Sample Customization File

The following sample customization file defines three attributes:

```
<property--list>
   <property label="Custom label 1" attr="custldapattr1" uifield="text"
syntax="string"/>
   <property label="Custom label 2" attr="custldapattr2" uifield="textarea"
syntax="number"/>
   <property label="Custom label 3" attr="custldapattr3" uifield="checkbox"/>
</property--list>
```

In this example, the first line defines an LDAP attribute, **custldapattr1**, with the following label in the console: Custom label 1. Next to the label, the console will display a text box that accepts an alphanumeric string as input.

For the second attribute defined in this example, the console displays a label and a text area that accepts a number as input.

For the third attribute, the console displays a label and a check box. Delegated Administrator returns a value of **true** (if the check box is checked) or **false** (if it is unchecked) to be stored with the LDAP attribute in the directory.

Configuring the Preferred Mail Host Using the Server-Wide Default

If you want the Preferred Mail Host and Preferred Mail Store to be set using the server-wide default, you can perform the tasks described in this section.

If you need to remove the Preferred Mail Host field from the console (specifically, from the New Organization Wizard and Organization Properties screens), you must perform the following steps:

- Edit the **Security.properties** file. This step is described in this section.
- Enable the MailHostStorePlugin. This step is described in the "Adding Plug-ins for Delegated Administrator" section.

The **Security.properties** file lets you customize the Delegated Administrator console for all or for individual roles.

To remove the Preferred Mail Host from the console:

1. Add the lines shown below to the **Security.properties** file.

```
# Remove Preferred Mail Host from UI
*.NewOrganizationPage6.PreferredMailHostProperty=INVISIBLE
*.NewOrganizationSummaryPage.PreferredMailHostSummaryProperty=INVISIBLE
*.OrgProperties.MailHostName=INVISIBLE
*.OrgProperties.MailHostNameText=INVISIBLE
*.OrgProperties.MailHostValue=INVISIBLE
```

Caution: Do not use the line:

```
*.NewOrganizationPage6.PreferredMailHostProperty=INVISIBLE
```

unless you turn on the server-side plug-in.

The **Security.properties** file is located in the following directory:

*DelegatedAdmin_home***/data/da/WEB-INF/classes/com/sun/comm/da/resources**

Caution: You may add lines to this file for your own customization, but do not edit the lines already present. Editing existing lines could result in exceptions being thrown on the console.

---

2. Redeploy the edited **Security.properties** file to the Web container used by the Delegated Administrator console.

   Before the change can take effect, you must run the script that deploys the customized **Security.properties** file to your Web container.

   See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

Syntax and Values for Security.properties File Properties

Properties in the **Security.properties** file are of the form:

Security_Element=status

where:

- *Security_Element* is of the form *Role.Container.Console_Element* and specifies the console element (for example: **MailHostNameText**) and role for which a status is being assigned.
- Valid roles for Delegated Administrator are:
  - **ProviderAdminRole** (SPA) See "Service Provider Administrator and Service Provider Organizations" for information about this role.
  - **OrganizationAdminRole** (OA)
  - **Top-levelAdminRole** (TLA)
- You can use an asterisk to represent multiple roles. For example, the security element ***.OrgProperties.MailHostNameText** applies to all three roles and is equivalent to:
  - **ProviderAdminRole.OrgProperties.MailHostNameText**
  - **OrganizationAdminRole.OrgProperties.MailHostNameText**
  - **Top-levelAdminRole.OrgProperties.MailHostNameText**
- You can assign a security element separate statuses of **VISIBLE/INVISIBLE** and **EDITABLE/NONEDITABLE**:
  - **VISIBLE**: indicates that the security element is visible and read-only.
  - **INVISIBLE**: indicates that the security element is invisible.
  - **EDITABLE**: indicates that the security element is editable.
  - **NONEDITABLE**: indicates that the security element is read-only.

In assigning statuses to security elements, a later assignment overrides an earlier assignment. For example, given the sequence:

```
OrganizationAdminRole.OrgProperties.MailHostNameText=EDITABLE
*.OrgProperties.MailHostNameText=NONEDITABLE
```

MailHostNameText is non-editable for all roles.

However, given the following sequence, **MailHostNameText** is non-editable for the SPA and TLA roles, but editable for the OA role:

```
*.OrgProperties.MailHostNameText=NONEDITABLE
OrganizationAdminRole.OrgProperties.MailHostNameText=EDITABLE
```

Adding Plug-ins for Delegated Administrator

You can customize Delegated Administrator to support the following plug-ins:

- **MailHostStorePlugin**

  By default, this plug-in is disabled. If no preferredmailhost is supplied when a business organization is created, an exception will be raised. If the plug-in is enabled, values from the flat file (described later in this section) will be used only if the corresponding attribute is absent.

- **MailDomainReportAddressPlugin**

Uses the domain value to return the desired DSN address. The default implementation is to return the string **MAILER-DAEMON@***domain*.

- **UidPlugin**

  Generates a unique id string. The default implementation generates a GUID to return to the caller.

Enabling the Plug-Ins

To enable these plug-ins, edit the commcli servlet **serverconfig.properties** file, located in the following directory:

```
DelegatedAdmin_home/data/WEB-INF/classes/sun/comm/cli/server/servlet/
serverconfig.properties
```

(By default, *DelegatedAdmin_home* for Oracle Solaris and Linux is **/opt/sun/comms**.)

The plug-ins are located in the **serverconfig.properties** file in a section headed as follows:

```
########################
# Plugin Configuration #
########################
```

Each has "plugin" as the suffix. The current list looks like:

```
jdapi-mailhoststoreplugin=disabled
jdapi-mailhoststorepluginclass=sun.comm.cli.server.util.MailHostStorePlugin
jdapi-mailhoststorepluginfile=/tmp/mailhostmailstore
jdapi-maildomainreportaddressplugin=enabled
jdapi-maildomainreportaddresspluginclass=sun.comm.cli.server.util.MailDomainReport
AddressPlugin
jdapi-uidautogenerationplugin=disabled
jdapi-uidautogenerationpluginclass=sun.comm.cli.server.util.UidPlugin
```

Each plug-in has at least two lines, which take the following form:

```
jdapi-namepluginclass= "enablted" | "disablted"
jdapi-namepluginclass=sun.comm.cli.server.util/ \ java class name
```

To enable a plug-in, change "disabled" to "enabled".

Plug-in classes are supplied for all the plug-ins listed in this section. The classes are located in the following directory:

```
DelegatedAdmin_home/data/WEB-INF/classes/sun/comm/cli/server/util
```

You do not need to do anything with these classes.

After you edit the **serverconfig.properties** file, redeploy it to the Web container used by the Delegated Administrator server.

Before the changes can take effect, you must run the script that deploys the customized **serverconfig.properties** file to your Web container.

See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

Additional Flat File Required for MailHostStorePlugin

The **MailHostStorePlugin** requires a flat file, which is included in a third line for the plug-in. The plug-in reads the value in the flat file and uses it to set attribute values. If the plug-in is enabled, the file must be present, or an error will occur.

```
jdapi-mailhoststoreplugin
o jdapi-mailhoststorepluginin=full file name
o file has one line
o value is that for :
o preferredmailhost attribute
o preferredmailmessagestore attribute
o form
o mailhost:mailpartition
```

Adding a Custom Object Class When You Create an LDAP Object

You can enable Delegated Administrator to add a custom object class to the LDAP entry of a user, group, resource, or organization. To accomplish this task, you customize the appropriate object-creation template installed in the directory by Access Manager.

For example, the BasicUser creation template determines which object classes and attributes are added to a user entry when you create a user. You can update the BasicUser creation template with your custom object class. Thereafter, the custom object class will be added to each user entry with the standard object classes.

The following procedure describes how to customize the BasicUser template. You can follow the same procedure to customize the BasicGroup, BasicResource, and BasicOrganization creation templates.

To add a custom object class to the user-creation process

1. Make sure your custom object class is defined in the directory schema. See Directory Schema for more information.
2. Locate the following directory entry:

   ```
   ou=basicuser,ou=creationtemplates,ou=templates,ou=default,
   ou=globalconfig,ou=1.0,ou=dai,ou=services,
   o=$Root_Suffix
   ```

   where *$Root_Suffix* is the root suffix of your directory.
3. Add the following *attribute:value* to the entry:

   ```
   sunkeyValue:required=objectClass=$Your_Custom_Objectclass.
   ```

   where *$Your_Custom_Objectclass* is your custom object class.

Customizing the User Log-In

When you run the Delegated Administrator configure program (**config-commda**), the value you use to log in to Delegated Administrator is set to be a **uid**.

For example, if you intend to log in as the TLA, and the TLA's **uid** is **john.doe**, you would use **john.doe** to log in to Delegated Administrator.

You can customize Delegated Administrator to enable you to use additional values for the user log-in. For example, you could add the mail address (**mail**).

How the User Log-In Value Is Set

The **config-commda** program sets this value to **uid** with the **loginAuth-idAttr** property in the **resource.properties** file, as shown in the following example:

```
loginAuth-searchBase=$rootSuffix
```

```
servicepackage-cosdefbasedn = $rootSuffix
loginAuth-idAttr-1=uid
```

where *$rootSuffix* is the root suffix in your directory.

Adding a User Log-In Value

You can set additional values for the user log-in by editing the **resource.properties** file.

The **resource.properties** file is located in

```
DelegatedAdmin_home/data/WEB-INF/classes/sun/comm/cli/server/servlet/
resource.properties
```

For example, to enable you to use a mail address (such as **john.doe@sesta.com**) to log in, you could add the following line to the **resource.properties** file:

```
loginAuth-searchBase=$rootSuffix
servicepackage-cosdefbasedn = $rootSuffix
loginAuth-idAttr-1=uid
loginAuth-idAttr-2=mail
```

where *$rootSuffix* is the root suffix in your directory.

You must add an increment to the **loginAuth-idAttr** property for each value. In this example, a second value is added, so you add **-2** to **loginAuth-idAttr**.

You can add multiple instances of the **loginAuth-idAttr** property:

```
loginAuth-idAttr-1=uid
loginAuth-idAttr-2=mail
|
loginAuth-idAttr-n=login-in value
```

After you edit the **resource.properties** file, redeploy it to the Web container used by the Delegated Administrator server.

Before the changes can take effect, you must run the script that deploys the customized **resource.properties** file to your Web container.

See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

Requiring Service Packages for Users

By default, Delegated Administrator lets you create a user without assigning a service package to the user. You can change the default setting so that all users must have at least one service package assigned to them.

To require users to have a service package assigned to them:

1. Open the **daconfig.properties** file in a text editor.

   The **daconfig.properties** file is located by default in the following directory:

   *DelegatedAdmin_home***/data/da/WEB-INF/classes/com/sun/comm/da/resources**

2. Change the value of the **user.atleastOneServicePackage** property from **false** to **true**.

   By default, this value is **false**.

   For example:

```
user.atleastOneServicePackage=true
```

3. Run the script that deploys the customized **daconfig.properties** file to your Web container.

   See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

Adding a Calendar Time Zone in Access Manager Mode

You can customize Delegated Administrator by adding a Calendar Server time zone. Delegated Administrator can then provision organizations, users, groups, and resources with the time zone.

To add a time zone, perform the following tasks. To administer the time zone with the Delegated Administrator utility, you must perform only the first task. To administer the time zone through Delegated Administrator console, you must perform both tasks.

- Adding a Time Zone in Delegated Administrator
- Displaying and Administering the Time Zone in the Delegated Administrator Console

Once the time zone has been added, you can set it as the default time zone for users created by performing the following task:

- Changing the Default Time Zone in Delegated Administrator

Adding a Time Zone in Delegated Administrator

You must perform this task before you provision users with the time zone. This task updates Access Manager with the time zone value.

To add a time zone in Delegated Administrator:

1. Add the time zone in Calendar Server.

   To accomplish this step, you must edit the **timezones.ics** file and other Calendar Server files. For Calendar 7, see *Calendar Server System Administrators Guide*.

2. Back up the **UserCalendarService.xml** and **DomainCalendarService.xml** files.

   The xml files are located by default in the following directory:

   *DelegatedAdmin_home***/lib/services**

3. Edit the **UserCalendarService.xml** and **DomainCalendarService.xml** files to add the time zone in Delegated Administrator.

   - In both the **UserCalendarService.xml** and **DomainCalendarService.xml** files, find the following entry heading:

     ```
     <AttributeSchema name="icstimezone"
     type="single choice"
     syntax="string"
     any="optional|adminDisplay">
     <ChoiceValues>
     ```

   - Add the time zone value to the list of <**ChoiceValues**>.

4. Run the Access Manager **amadmin** command to delete the current service and add the updated service. For both the **UserCalendarService.xml** and **DomainCalendarService.xml** files, run the following **amadmin** commands:

```
amadmin -u admin -w password -rCalendarService

./amadmin -u admin -w password
-sda_base/lib/services/CalendarService.xml
```

where *CalendarService* is either **UserCalendarService** or **DomainCalendarService**.

> **Note:** If you also intend to make the time zone your default, you can run these **amadmin** commands after you have performed both tasks. See "Changing the Default Time Zone in Delegated Administrator" for details.

5. Restart your Web container to enable the changes to take effect.
6. See "Displaying and Administering the Time Zone in the Delegated Administrator Console" to enable the Delegated Administrator console to show the time zone.

Displaying and Administering the Time Zone in the Delegated Administrator Console

This task adds the time zone to the list of time zones displayed in the console. Next, the task enables the time zone value to be saved in the directory.

1. Edit the **Resources.properties** file, located under your Delegated Administrator data directory.

   The **Resources.properties** file is located by default in the following directory:

   ```
   DelegatedAdmin_home/data/da/WEB-INF/classes/com/sun/
   comm/da/resources
   ```

   To edit **Resources.properties**, search for the **rsrc.Timezone** property and add the time zone to the appropriate list. You can localize this display value of the time zone.

2. Locate the list of time zone values in the **daconfig.properties** file, located under your Delegated Administrator data directory.

   The **daconfig.properties** file is located by default in the following directory:

   ```
   DelegatedAdmin_home/data/da/WEB-INF/classes/com/sun/
   comm/da/resources
   ```

   To find the list of time zone values, search for the following line:

   ```
   #Timezone values - only English
   ```

   These are the values stored in the LDAP directory. The time zone must be in English; this is the required format for the values stored in the directory.

3. Add the time zone to the list.

   For example, to add America/Miami to the list, assuming Timezone1 currently has 24 values, you would add

   ```
   rsrc.Timezone1-25=America/Miami
   ```

   This value would be the 25th time zone displayed in the Americas drop-down list in the console. The time zone may be displayed in another language, depending on what you specified in the **Resources.properties** file in the preceding task.

4. Locate the reverse-time zone mappings list in the **daconfig.properties** file.

This list keys the localized time zone value (displayed in the console) to the actual value, which you specified in Step 2, above.

To find the list of reverse mappings, search for the following line:

```
#reverse timezone mappings - used by DA in getting localized tz value
```

5. Add the value to the reverse-mapping list.

   For example, to add America/Miami to the list, you would add

   ```
   rsrcKey-America-Miami=rsrc.Timezone1-25
   ```

6. Redeploy the edited **daconfig.properties** and **Resources.properties** files to the Web container used by the Delegated Administrator console.

   Before the change can take effect, you must run the script that deploys the customized **daconfig.properties** file to your Web container.

   See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

   After you edit and redeploy the **daconfig.properties** and **Resources.properties files**, the time zone will appear in the appropriate list boxes in the Delegated Administrator console. It will be saved in the directory whenever you select the time zone in the Delegated Administrator console and click **Save**.

Changing the Default Time Zone in Delegated Administrator

To change the default time zone in Delegated Administrator:

1. In the **UserCalendarService.xml** and **DomainCalendarService.xml** files, edit the following value:

   ```
   <DefaultValues>
        <Value>America/Denver</Value>
   </DefaultValues>
   ```

   You can find <**DefaultValues**> under the following entry in the xml files:

   ```
   AttributeSchema name="icstimezone"
   ```

2. Run the Access Manager **amadmin** command to delete the current service and add the updated service.

   For both the **UserCalendarService.xml** and **DomainCalendarService.xml** files, run the following **amadmin** commands:

   ```
   ./amadmin -u admin -w password -r DomainCalendarService

   ./amadmin -u admin -w password
   -sda_base/lib/services/DomainCalendarService.xml
   ```

3. Restart your Web container to enable the changes to take effect.

Adding a Calendar Time Zone in Direct LDAP Mode

The directory **/opt/sun/comms/da/lib/services** contains reference information about the services for both modes. XML files are meant for Access Manager mode while LDIF files are meant for Direct LDAP mode.

However, these LDIF files are not quite exactly what has been employed in LDAP server.

This information is within the file **/var/opt/sun/comms/da/config/daAllServices.ldif**. Adding the time zone is performed by editing this file and then implementing its content into the LDAP service. This example describes the addition of the **Europe/Belgrade** time zone.

1. Save the backup copy of this file and open it in editor.
2. There are three places in which you will perform the same type of editing. They relate to the services **DomainCalendar**, **GroupCalendar** and **UserCalendar** and the specific attribute within each of them is called **icstimezone**. At each of these places, at the desired place within the list of available time zones, add the following:

   ```
   ChoiceValueEurope/Belgrade/ChoiceValue
   ```

3. If you want to declare this time zone as default, then perform the following. Again, within the same three places, inside the attribute **icstimezone**, substitute the default time zone **America/Denver** with this one. The end result looks like this:

   ```
   <DefaultValues><Value>Europe/Belgrade</Value></DefaultValues>
   ```

4. Finally perform the changes:

   ```
   ldapmodify -D 'cn=Directory Manager' -w -f daAllServices.ldif
   ```

However, this method may fail due of the improper formatting of the XML text (the one that is going to become the value of the LDAP attribute **sunserviceschema**). In that case, you will need to remove all End-of-line characters from that XML description. One way to do this is extracting necessary parts into three separate files (one for each of the Calendar services), reformat the text and apply the files into the LDAP service.

Another way is to run three LDAP searches and have the different files. Through LDAP searches you'll get the text without End-of-Line characters, but encoded in Base64. Beware that simple command to decode and re-encode in Base64 does not exist in Solaris 10. Follow the steps:

1. Get the current values of the Calendar services from the LDAP directory:

   ```
   ldapsearch -D 'cn=Directory Manager' -w -1 -T -b
   ou=1.0,ou=DomainCalendarService,o=daservices,o=comms-config "(|(objectClass=*)
   (objectClass=ldapsubentry))" sunserviceschema > /tmp/DomainCalendar.ldap
   ldapsearch -D 'cn=Directory Manager' -w -1 -T -b
   ou=1.0,ou=GroupCalendarService,o=daservices,o=comms-config "(|(objectClass=*)
   (objectClass=ldapsubentry))" sunserviceschema > /tmp/GroupCalendar.ldap
   ldapsearch -D 'cn=Directory Manager' -w -1 -T -b
   ou=1.0,ou=UserCalendarService,o=daservices,o=comms-config "(|(objectClass=*)
   (objectClass=ldapsubentry))" sunserviceschema > /tmp/UserCalendar.ldap
   ```

2. Extract the value of the **sunserviceschema** attribute:

   ```
   grep ^sunserviceschema /tmp/DomainCalendar.ldap | cut -d" " -f 2 >
   /tmp/DomainCalendar.base64
   grep ^sunserviceschema /tmp/GroupCalendar.ldap | cut -d" " -f 2 >
   /tmp/GroupCalendar.base64
   grep ^sunserviceschema /tmp/UserCalendar.ldap | cut -d" " -f 2 >
   /tmp/UserCalendar.base64
   ```

3. Decode the **Base64** files. This example uses the command base64 available in package GNU coreutils. It also adds End-Of-Line characters so that you can easily edit the textual contents:

```
base64 -d /tmp/DomainCalendar.base64 | sed -e 's/> />\n /g' -e 's/" /"\n /g' >
/tmp/DomainCalendar.xml
base64 -d /tmp/GroupCalendar.base64 | sed -e 's/> />\n /g' -e 's/" /"\n /g' >
/tmp/GroupCalendar.xml
base64 -d /tmp/UserCalendar.base64 | sed -e 's/> />\n /g' -e 's/" /"\n /g' >
/tmp/UserCalendar.xml
```

4. Add the time zone in the just created XML files, as described. Then remove the End-Of-Line characters and create the final LDIF files ready for deployment into LDAP service:

```
( echo -e -n "dn:
ou=1.0,ou=DomainCalendarService,o=daservices,o=comms-config\nchangetype:
modify\nreplace: sunserviceschema\nsunserviceschema: " ; cat
/tmp/DomainCalendar.xml | tr -d '\n' ) > /tmp/DomainCalendar.ldif
( echo -e -n "dn:
ou=1.0,ou=GroupCalendarService,o=daservices,o=comms-config\nchangetype:
modify\nreplace: sunserviceschema\nsunserviceschema: " ; cat
/tmp/GroupCalendar.xml | tr -d '\n' ) > /tmp/GroupCalendar.ldif
( echo -e -n "dn:
ou=1.0,ou=UserCalendarService,o=daservices,o=comms-config\nchangetype:
modify\nreplace: sunserviceschema\nsunserviceschema: " ; cat
/tmp/UserCalendar.xml | tr -d '\n' ) > /tmp/UserCalendar.ldif
```

5. Finally, perform the changes:

```
ldapmodify -D 'cn=Directory Manager' -w -f /tmp/DomainCalendar.ldif
ldapmodify -D 'cn=Directory Manager' -w -f /tmp/GroupCalendar.ldif
ldapmodify -D 'cn=Directory Manager' -w -f /tmp/UserCalendar.ldif
```

Once you have finished modifying LDAP, you need to add the time zone into Delegated Administrator's configuration files. This is done in the same way that it was done for the Access Manager mode.

At this point one feature is highlighted that, if you already went through the effort of adding the time zone, may be preferable to all others. This explains how to position in on the top of the list of time zones displayed in the Delegated Administrator GUI, which saves you from scrolling every time through the list of time zones. The list is short for this example (Europe and Africa) but is relatively extensive compared to the other two world areas available in the GUI.

1. Change to the directory containing configuration files:

```
cd /var/opt/sun/comms/da/da/WEB-INF/classes/com/sun/comm/da/resources
```

2. For each of the **Resource*.properties** files, both English and localized ones, perform the following

   For each line that begins with **rsrc.Timezone2**- increment the latter number by one. For example, in file **Resources.properties**, change the line:

```
rsrc.Timezone2-16=(GMT+04:00) Europe/Samara
```

   to become like this:

```
rsrc.Timezone2-17=(GMT+04:00) Europe/Samara
```

   This way, the leading place (numbered "0") is being freed to receive your time zone. Use the following table to enter time zone description within each of the files:

```
Resources.properties: rsrc.Timezone2-0=(GMT+01:00) Europe/Belgrade
```

```
Resources_de.properties: rsrc.Timezone2-0=(GMT+01:00) Europa/BelgradR
esources_es.properties: rsrc.Timezone2-0=(GMT+01:00) Europa/Belgrado
Resources_fr.properties: rsrc.Timezone2-0=(GMT+01:00) Europe/Belgrade
Resources_ja.properties: rsrc.Timezone2-0=(GMT+01:00)
\u30e8\u30fc\u30ed\u30c3\u30d1\u30d9\u30aa\u30b0\u30e9\u30fc\u30c9
Resources_ko.properties: rsrc.Timezone2-0=(GMT+01:00)
\uc720\ub7fd\ubca8\uadf8\ub77c\ub4dc
Resources_zh.properties: rsrc.Timezone2-0=(GMT+01:00)
\u6b50\u6d32\u8d1d\u5c14\u683c\u83b1\u5fb7
Resources_zh_CN.properties: rsrc.Timezone2-0=(GMT+01:00)
\u6b50\u6d32\u8d1d\u5c14\u683c\u83b1\u5fb7
Resources_zh_TW.properties: rsrc.Timezone2-0=(GMT+01:00)
\u6b50\u6d32\u8c9d\u723e\u683c\u840a\u5fb7
Resources_zh_cn.properties: rsrc.Timezone2-0=(GMT+01:00)
\u6b50\u6d32\u8d1d\u5c14\u683c\u83b1\u5fb7
Resources_zh_tw.properties: rsrc.Timezone2-0=(GMT+01:00)
\u6b50\u6d32\u8c9d\u723e\u683c\u840a\u5fb7
```

3. Edit file **daconfig.properties**

   - Increment secondary number in each of the lines starting with **rsrc.Timezone2-**, just the same as you did with previous files
   - In the freed space, enter the line:

     ```
     rsrc.Timezone2-0=Europe/Belgrade
     ```

   - Increment secondary number in each of the lines starting with **rsrckey** and ending with **rsrc.Timezone2-**, just the same as you did with previous files. For example, change the line:

     ```
     rsrcKey-Europe-Samara=rsrc.Timezone2-16
     ```

     to become like this:

     ```
     rsrcKey-Europe-Samara=rsrc.Timezone2-17
     ```

   - In the freed space, enter the line:

     ```
     rsrckey-Europe-Belgrade=rsrc.Timezone2-0
     ```

4. Restart your Web container to enable the changes to take effect.

Adding Support for the Local Language in Delegated Administrator

This example shows how to add support for local language in Delegated Administrator. It is not the translation of the Web interface, but rather it's about making it able to have custom values for the **preferredDomain** LDAP attribute. In this example, Serbian language is going to be added. Similarly to Chinese, it has two different transcription: Serbian Latin and Serbian Cyrillic, which will be herein referenced as sr and cp, respectively.

1. Add the language into the list of available resources.

   Change into the directory containing description of the resources:

   ```
   cd /var/opt/sun/comms/da/da/WEB-INF/classes/com/sun/comm/da/resources
   ```

   Change the following line near the top of the file **daconfig.properties**:

   ```
   locale.supported=sr,cp,en,de,es,fr,ja,ko,zh_CN,zh_TW
   ```

Add these lines to the file **Resources.properties**:

```
locale.sr=Serbian Latin
locale.cp=Serbian Cyrillic
```

Add these lines to the file **Resources_de.properties**:

```
locale.sr=Serbisch latinisch
locale.cp=Serbisch kyrillisch
```

Add these lines to the file **Resources_es.properties**:

```
locale.sr=Lat\u00ednico serbio
locale.cp=Cir\u00edlico serbio
```

Add these lines to the file **Resources_fr.properties**:

```
locale.sr=Serbe cyrillique
locale.cp=Serbe latin
```

Add these lines to the file **Resources_ja.properties**:

```
locale.sr=\u30bb\u30eb\u30d3\u30a2\u8a9e\u30e9\u30c6\u30f3\u8a9e
locale.cp=\u30bb\u30eb\u30d3\u30a2\u8a9e\u30ad\u30ea\u30eb\u6587\u5b57
```

Add these lines to the file **Resources_ko.properties**:

```
locale.sr=\ub77c\ud2f4 \uc138\ub974\ube44\uc544\uc5b4
locale.cp=\ud0a4\ub9b4 \uc138\ub974\ube44\uc544\uc5b4
```

Add these lines to the file **Resources_zh_CN.properties**:

```
locale.sr=\u585e\u5c14\u7ef4\u4e9a\u62c9\u4e01\u8a9e
locale.cp=\u585e\u5c14\u7ef4\u4e9a\u897f\u91cc\u5c14\u6587
```

Add these lines to the file **Resources_zh_TW.properties**:

```
locale.sr=\u585e\u723e\u7dad\u4e9e\u62c9\u4e01\u8a9e
locale.cp=\u585e\u723e\u7dad\u4e9e\u897f\u91cc\u723e\u6587
```

Make sure not to miss other Chinese localization files:

```
cp Resources_zh_CN.properties Resources_zh_cn.properties
cp Resources_zh_TW.properties Resources_zh_tw.properties
cp Resources_zh_CN.properties Resources_zh.properties
```

2. Add the language to the tasks related to the Organizations (to add new and view current) Change into the directory containing actions over Organizations:

```
cd /var/opt/sun/comms/da/da/jsp/organizations
```

Add these lines to the file **newOrganization.xml**:

```
<option label="locale.sr" value="sr" />
<option label="locale.cp" value="cp" />
```

Add these lines to the file **newOrganization_nowrap.xml**:

```
<option label="locale.sr" value="sr" />
<option label="locale.cp" value="cp" />
```

Within the file **OrgPropsPropertySheet.xml** remove the closing of the cc tag and add the following lines, so that it looks:

```
<cc name="PreferredLanguageValue"
```

```
tagclass="com.sun.web.ui.taglib.html.CCDropDownMenuTag" >
  <option label="locale.sr" value="sr" />
  <option label="locale.cp" value="cp" />
  <option label="locale.en" value="en" />
  <option label="locale.fr" value="fr" />
  <option label="locale.de" value="de" />
  <option label="locale.es" value="es" />
  <option label="locale.ja" value="ja" />
  <option label="locale.ko" value="ko" />
  <option label="locale.zh_CN" value="zh_CN" />
  <option label="locale.zh_TW" value="zh_TW" />
</cc>
```

3. Add the language to the tasks related to the Users (to add new and view current) Change into the directory containing actions over Users:

   ```
   cd /var/opt/sun/comms/da/da/jsp/users
   ```

   Within the file **newUser.xml** remove the following lines:

   ```
   <!-- <option label="newuser.wizard.preferredlanguage.en"
   value="newuser.wizard.preferredlanguage.en" />
        <option label="newuser.wizard.preferredlanguage.de"
   value="newuser.wizard.preferredlanguage.de" />
        <option label="newuser.wizard.preferredlanguage.pl"
   value="newuser.wizard.preferredlanguage.pl" />
   --> </cc>
   ```

   and add the following lines in place of the ones you just removed:

   ```
   <option label="Serbian Latin" value="sr" />
   <option label="Serbian Cyrillic" value="cp" />
   <option label="English" value="en" />
   <option label="German" value="de" />
   <option label="Spanish" value="es" />
   <option label="French" value="fr" />
   <option label="Japanese" value="ja" />
   <option label="Korean" value="ko" />
   <option label="Simplified Chinese" value="zh_CN" />
   <option label="Traditional Chinese" value="zh_TW" />
   </cc>
   ```

   Within the file **userProperties.xml** remove the closing of the cc tag and add the following lines, so that it appears as follows:

   ```
   <cc name="PreferredLanguageValue"
   tagclass="com.sun.web.ui.taglib.html.CCDropDownMenuTag">
     <option label="Serbian Latin" value="sr" />
     <option label="Serbian Cyrillic" value="cp" />
     <option label="English" value="en" />
     <option label="German" value="de" />
     <option label="Spanish" value="es" />
     <option label="French" value="fr" />
     <option label="Japanese" value="ja" />
     <option label="Korean" value="ko" />
     <option label="Simplified Chinese" value="zh_CN" />
     <option label="Traditional Chinese" value="zh_TW" />
   </cc>
   ```

4. Restart your Web container to enable the changes to take effect.

Deploying a Customized Configuration File

When you configure Delegated Administrator with the **config-commda** program, **config-commda** places the configuration files in the installation directory. Next, the program deploys the configuration files to the application repository of the Web container, where you deployed Delegated Administrator. Thus the deployed location of the configuration files varies, depending on which Web container is used.

At run time, Delegated Administrator uses the property values of the configuration files in their deployed locations, that is, in the Web container repository to which Delegated Administrator has been deployed.

To customize a configuration file:

1. Edit the original configuration file located in the Delegated Administrator installation directory.
2. Use a script provided by Delegated Administrator to redeploy the configuration file to the Web container.

When you customize a configuration file, the values do not take effect until the file has been redeployed to the Web container.

To deploy a customized configuration file:

1. Log in as (or become) root and go to the following directory:

   DelegatedAdmin_home/sbin

2. Run the appropriate deploy script to redeploy your customized configuration file to the Web container used by Delegated Administrator.

   You must redeploy the configuration file to the Web container where you deployed Delegated Administrator the last time you ran the Delegated Administrator configuration program (**config-commda**).

   Use the **deploy** script that applies both to your customized configuration file and to the correct Web container.

   For example, to redeploy the **resource.properties** file to Web Server 6, run this command:

   ```
   # ./config-wbsvr-commcli
   ```

See "Configuration File Deploy Scripts" for a list of the deploy scripts.

The remainder of this section describes the following topics:

Original (Standard) Locations of the Configuration Files

When Delegated Administrator is configured (after you run the **config-commda** program), the configuration files are located in the following directories:

- Delegated Administrator console:
  - **logger.properties**
  - **Resources.properties**

  - **Security.properties**
  - **logger.properties**

  Location: *DelegatedAdmin_home*/data/da/WEB-INF/classes/com/sun/comm/da/resources/

- Delegated Administrator server:
  - **resource.properties**
  - **serverconfig.properties** (only Delegated Administrator 7)

  Location: *DelegatedAdmin_home*/data/WEB-INF/classes/sun/comm/cli/server/servlet/

Deployed Locations of the Configuration Files

After you run the **config-commda** program, the configuration files are deployed to the following locations, depending on which Web container you have chosen to deploy Delegated Administrator.

Deployed Location of Delegated Administrator Server File (resource.properties)

The **resource.properties** file is deployed to one of the following default locations:

- Web Server 6.*x*

  `/opt/SUNWwbsvr/https-`*`hostname`*`/webapps/https-`*`hostname`*`/commcli/WEB-INF/classes/sun/comm/cli/server/servlet/`

- Web Server 7.*x*

  `/var/opt/SUNWwbsvr7/https-`*`hostname`*`/webapps/`*`hostname`*`/commcli/WEB-INF/classes/sun/comm/cli/server/servlet/`

- Application Server 7.*x*

  `/var/opt/SUNWappserver7/domains/domain1/server1/applications/j2ee--modules/commcli/WEB-INF/classes/sun/comm/cli/server/servlet/`

- Application Server 8.*x*

  `/var/opt/SUNWappserver/domains/domain1/applications/j2ee-modules/commcli/WEB-INF/classes/sun/comm/cli/server/servlet/`

Deployed Location of Delegated Administrator Console Configuration Files

The following files are deployed to the same default location:

- **daconfig.properties**
- **logger.properties**
- **Resources.properties**
- **Security.properties**

These properties files are deployed to one of the following default locations, depending on the Web container you have chosen to deploy Delegated Administrator:

- Web Server 6.*x*

  `/opt/SUNWwbsvr/https-`*`hostname`*`/webapps/https-`*`hostname`*`/da/WEB-INF/classes/com/sun/comm/da/resources`

- Web Server 7.*x*

  `/var/opt/SUNWwbsvr7/https-`*`hostname`*`/webapps/`*`hostname`*`/da/WEB--INF/classes/com/sun/comm/da/resources`

- Application Server 7.*x*

  `/var/opt/SUNWappserver7/domains/domain1/server1/applications/j2ee-modules/Delegated_Administrator/WEB-INF/classes/com/sun/comm/da/resources`

- Application Server 8.*x*

  `/var/opt/SUNWappserver/domains/domain1/applications/j2ee-modules/Delegated_Administrator/WEB-INF/classes/com/sun/comm/da/resources`

Configuration File Deploy Scripts

There are two deploy scripts for each Web container. One script deploys the Delegated Administrator server files. The other deploys Delegated Administrator console files:

- Delegated Administrator server configuration files: **resource.properties** and **serverconfig.properties**.
- Delegated Administrator console configuration files: **daconfig.properties**, **Security.properties**, **Resources.properties**, and **logger.properties**.

Table 8–2 describes the deploy scripts.

Table 8–2 Deploy Scripts

Web Container	Deploy Scripts
Web Server 6	■ Deploy script for Delegated Administrator server files (**resource.properties** and **serverconfig.properties**): config-wbsvr-commcli ■ Deploy script for Delegated Administrator console files: **config-wbsvr-da** ■ To run the scripts, enter these commands: **# ./config-wbsvr-commcli** **# ./config-wbsvr-da**
Web Server 7.x	■ Deploy script for Delegated Administrator server files (**resource.properties** and **serverconfig.properties**): **config-wbsvr7x-commcli** ■ Deploy script for Delegated Administrator console files: **config-wbsvr7x-da** ■ To run the scripts, enter these commands: **# ./config-wbsvr7x-commcli** **# ./config-wbsvr7x-da**

Table 8–2 (Cont.) Deploy Scripts

Web Container	Deploy Scripts
Application Server 7.x	■ Deploy script for Delegated Administrator server files (**resource.properties** and **serverconfig.properties**): **config-appsvr-commcli** ■ Deploy script for Delegated Administrator console files: **config-appsvr-da** ■ To run the scripts, enter these commands: **# ./config-appsvr-commcli deploy** **# ./config-appsvr-da deploy** You must use the argument **deploy** with these commands.
Application Server 8.x	■ Deploy script for Delegated Administrator server files (**resource.properties** and **serverconfig.properties**): **config-appsvr8x-commcli** ■ Deploy script for Delegated Administrator console files: **config-appsvr8x-da** ■ To run the scripts, enter these commands: **# ./config-appsvr8x-commcli deploy** **# ./config-appsvr8x-da deploy** You must use the argument **deploy** with these commands.

9

Troubleshooting Delegated Administrator

This chapter describes how to troubleshoot problems encountered while using Oracle Communications Delegated Administrator.

Troubleshooting Problems

You can obtain log information for Delegated Administrator by examining log files generated by the Delegated Administrator components, by the Web container to which Delegated Administrator has been deployed, and by Directory Server and Access Manager.

The Delegated Administrator is a multi-client and server environment. A fully deployed environment has dependencies on LDAP, two web containers, a server, two clients, and possibly Access Manager.

Thus, a problem description contains three critical elements:

1. A basic description of the problem
2. Version information
3. The answer to the question, "Does the problem happen when you use **commadmin**, Delegated Administration Console (web client), or both?"

Collecting this information helps to minimize problem resolution time.

Troubleshooting the Command-Line Utilities

To debug the Delegated Administrator utility (**commadmin**), you can print debug messages in the client by using the **-v** option with the **commadmin** command.

This feature is interactive, you can enable it at any time, with any full (object + task) usage.

If you are unable to solve the problem by using the verbose output, consider expanding your analysis to include the Delegated Administration Server. Refer to the debugging features described later in this information.

If you are submitting a problem to Oracle Technical Support, save both normal and verbose output.

Delegated Administrator Console Log

By default, Delegated Administrator Console logging is off.

The Delegated Administrator console creates a run-time log file:

- Default log file name: **da.log**
- Default location: *DelegatedAdmin_home***/log/**

You can specify your own log file by editing a log properties file named **logger.properties**.

To specify your own Delegated Administrator Console log file:

1. Open the **logger.properties** file in a text editor.

   The **logger.properties** file is located by default in the following directory:*DelegatedAdmin_home*/data/da/WEB-INF/classes/com/sun/comm/da/resources

2. You can change the following properties in the **logger.properties** file:

   - **da.logging.enable=yes** or **no**

     where **yes** enables logging and **no** disables logging. By default, logging is disabled.

     To turn on logging, you must set this value to **yes**.

   - **da.log.file=***full pathname* specifies the directory and file to which logging statements are written. This property changes **da.log** to a file name and location you specify.

3. Redeploy the edited **logger.properties** file to the Web container used by the Delegated Administrator console.

   Before the change can take effect, you must run the script that deploys the customized **logger.properties** file to your Web container.

   See "Deploying a Customized Configuration File" for instructions on how to deploy a customized properties file to a particular Web container.

Delegated Administrator Server Log

You can create a Delegated Administrator Server log that contains error messages generated by the Delegated Administrator Server's servlets installed on the web container.

To do so, you enable a Debug servlet to log debug messages from the Delegated Administrator Server's servlets execution.

You use the **commadmin debug log** command to enable Delegated Administrator Server to write messages to a debug log. See "commadmin debug log" for additional information.

For example, enter the following command:

```
commadmin debug log -D paul -n sesta.com -t on
```

In the preceding command, Paul is the Top Level Administrator. The command would log Debug servlet messages to the default path and file:

```
/tmp/commcli.log
```

> **Note:** The log can only be created in the **/tmp/** or **/var/tmp/** directory.

Whenever you restart the Web container, you must run the **commadmin debug log** command again.

Web Container Server Logs

You can troubleshoot Delegated Administrator by examining the server logs generated by your Web container.

Table 9–1 Web Container Server Logs

Web Container	Container Description	File Path	File Path Variable Description
Web Server 6.*x*	Maintains access and error logs	*web_server6_base*/**https-***host.domain*/**logs**	*web_server6_base* is the base where Web Server 6.*x* software is installed. For example: **/opt/SUNWwebsvr**. *host.domain* is the host and domain name of the machine where Web Server 6.*x* is running.
Web Server 7.*x*	Maintains access and error logs	*web_server7_config_base*/**https-***host.domain*/**logs**	*web_server7_config_base* is the path where Web Server 7.*x* configuration and log files are installed. For example: **/var/opt/SUNWwbsvr7**. *host.domain* is the host and domain name of the machine where Web Server 7.*x* is running.
Application Server 7.*x*	Maintains access and error logs	*/application_server7_base*/**domains/domain1/server1/logs**	*application_server7_base* is the path where Application Server 7.*x* software is installed.
Application Server 8.*x*	Maintains access and error logs	Server log: */application_server8_base*/**domains/domain1/logs** Access log: */application_server8_base*/**domains/domain1/logs/access/server_access_log**	*application_server8_base* is the path where Application Server 8.*x* software is installed.
Application Server 9.0	Maintains access and error logs For more information, see About Logging.	Server log: */application_server9_base*/**domains/domain1/logs** Access log: */application_server9_base*/**domains/domain1/logs/acces/server_access_log**	*application_server9_base* is the path where Application Server 9.*x* software is installed.

Directory Server and Access Manager Logs

You can debug Delegated Administrator by examining the logs generated by Directory Server and Access Manager.

Directory Server maintains access and error logs, located in the following path:

/var/opt/mps/serverroot/slapd-*hostname***/logs** where *hostname* is the name of the machine where Directory Server is running.

For more information on Directory Server logs, see the documentation about Directory Server Enterprise Edition 6.3

Access Manager maintains log files in the following paths:

/var/opt/SUNWam/debug

The preceding path contains the **amProfile** and **amAuth** logs.

/var/opt/SUNWam/logs

The preceding path contains the **amAdmin.access** and **amAdmin.error** logs.

10

Support For Additional Values of mailuserstatus

Oracle Communications Delegated Administrator enables you to choose from several **mailuserstatus** values including **active**, **inactive**, **deleted**, and **hold**.

To support additional **mailuserstatus** values, you must perform the following steps:

1. Add the following entries to the **da webapp Resources_*.properties** file:

```
userlist.xxxxLabel = xxxx
userproperties.mailstatus.xxxx = xxxx
```

2. Modify the **da webapp userProperties.xml** file:

```
<property name="MailStatusProperty">
    <label name="MailStatus" defaultValue="userproperties.mailstatus"
         Id="MailStatusFor" labelFor="MailStatusValue"/>
     <cc name="MailStatusValue"
tagclass="com.sun.web.ui.taglib.html.CCDropDownMenuTag">
         <option label="userproperties.mailstatus.active" value="active"/>
         <option label="userproperties.mailstatus.inactive" value="inactive" />
         <option label="userproperties.mailstatus.deleted" value="deleted" />
         <option label="userproperties.mailstatus.hold" value="hold" />
         <option label="userproperties.mailstatus.disabled" value="disabled" />
     </cc>
</property>
```

   to add the value as mentioned in step 1.

You can choose from several other additional values such as **disabled**, **removed**, **overquota**, and so on.

A

Service Package Details

This appendix describes various service package details in Oracle Communications Delegated Administrator.

Service Attributes Provided by the Sample Templates

This section describes the LDAP attributes provided by the sample service packages with mail service, instant messaging service, and contacts service.

> **Note:** Sample service packages with calendar service do not provide specific calendar attributes. For sample service package templates that combine services such as mail service and calendar service, mail attributes are provided, but no calendar attributes.

Mail Service Attributes

Mail service includes LDAP attributes defined for mail users. Table A–1 defines these attributes.

Table A–1 Mail service attributes that can be used in a service package

LDAP Attribute	Display Name	Definition
mailMsgMaxBlocks	Max Message Size (blocks)	Size in units of MTA blocks of the largest message that can be sent to the user or group.
mailAllowedServiceAccess	Allowed Services	Filter specifying the available client access to specified services. For example: **+imap:ALL$+pop:ALL$+smtp:ALL$+http:ALL**
mailMsgQuota	Max no. of Messages	Maximum number of messages permitted for a user (including all user folders).
mailQuota	Mail Quota	Disk space (in bytes) allowed for the user s mailbox.

For more information about these attributes, see *Communications Suite Schema Reference*.

Instant Messaging Service Attributes

IM service includes LDAP attributes defined for IM users. Table A–2 defines these attributes.

Table A–2 Instant Messaging Server service attributes that can be used in a service package

LDAP Attribute	Display Name	Definition
inetUserStatus	User Status	User status is set by default when the package is assigned using either **commadmin** command or the console.

For more information about Instant Messaging Server attributes, see the discussion about Instant Messaging object classes and attributes in *Communications Suite Schema Reference*.

Contacts Service Attributes

Contacts service includes LDAP attributes defined for Contacts users. Table A–3 defines these attributes.

Table A–3 Contacts service attributes that can be used in a service package

LDAP Attribute	Display Name	Definition
nabStatus	Contacts Service Status	Set by default when package is assigned using both the **commadmin** command and the console. Enables or disables the service. Absence of this attribute or a value of active indicates active status. Values of removed, deleted, or inactive disable the service. Any other value also may enable the service but is not recommended. Allowed values: active, inactive, removed, or deleted.
nabStore	Back-end Store	Indicates the back-end host in which a user's data resides if the deployment is setup with multiple back ends.
corpDirectoryUrl	CorpDir Url	Causes a domain to point to a different corporate directory. Also allows support for multiple corporate directories within that domain. You can add one or more **corpDirectoryUrl** attributes to the domain entry. The value of this attribute must be a valid corporate directory LDAP URL.

For more information about these attributes, see the discussion about Delegated Administrator object classes and attributes in *Communications Suite Schema Reference*.

Sample Class-of-Service Templates

This section lists the sample Class-of-Service templates and attribute values provided by the templates. These templates are contained in the **cos.sample.ldif** file.

User Mail Sample Templates

Platinum

```
mailMsgMaxBlocks: 800
mailquota: 10000000
mailmsgquota: 6000
mailAllowedServiceAccess: +imaps:ALL$+pops:ALL$+smtps:ALL$+http:ALL
daServiceType: mail user
```

Gold

```
mailMsgMaxBlocks: 700
mailquota: 8000000
mailmsgquota: 3000
mailAllowedServiceAccess: +imaps:ALL$+pops:ALL$+smtps:ALL$+http:ALL
daServiceType: mail user
```

Silver

```
mailMsgMaxBlocks: 300
mailquota: 6291456
mailmsgquota: 2000
mailAllowedServiceAccess: +pop:ALL$+imap:ALL$+smtp:ALL$+http:ALL
daServiceType: mail user
```

Bronze

```
mailMsgMaxBlocks: 700
mailquota: 5242288
mailmsgquota: 3000
mailAllowedServiceAccess: +pop:ALL$+imap:ALL$+smtp:ALL$+http:ALL
daServiceType: mail user
```

Ruby

```
mailMsgMaxBlocks: 600
mailquota: 1048576
mailmsgquota: 2000
mailAllowedServiceAccess: +pops:ALL$+smtps:ALL$+http:ALL
daServiceType: mail user
```

Emerald

```
mailMsgMaxBlocks: 600
mailquota: 2097152
mailmsgquota: 2000
mailAllowedServiceAccess: +pop:ALL$+smtp:ALL$+http:ALL
daServiceType: mail user
```

Diamond

```
mailMsgMaxBlocks: 5000
mailquota: 3145728
mailmsgquota: 3000
mailAllowedServiceAccess: +imaps:ALL$+smtps:ALL$+http:ALL
daServiceType: mail user
```

Topaz

```
mailMsgMaxBlocks: 3000
mailquota: 4194304
mailmsgquota: 2000
mailAllowedServiceAccess: +imap:ALL$+smtp:ALL$+http:ALL
daServiceType: mail user
```

User Calendar Sample Template

None (standardUserCalendar)

There is no predefined Class-of-Service template that provides calendar service and contains attribute values. Calendar service is provided without associated attributes. Because no sample template exists, Delegated Administrator generates a default

service package, without a template, directly from the User Calendar Class-of-Service definition. Its name is the same as that of the Class-of-Service definition: **standardUserCalendar**. This service package provides calendar service only.

User IM Sample Template

imsample

```
#
# User im templates
#
dn: cn=imsample,o=imuser,o=cosTemplates,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
cn: imsample
inetUserStatus: active
daServiceType: im user
```

User Contacts Sample Template

contactssample

```
dn: cn=contactssample,o=contactsuser,o=cosTemplates,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
nab: value
cn: contactssample
daServiceType: contacts user
```

User Mail and Calendar Sample Templates

The following sample templates apply both mail and calendar service.

Mercury

```
mailMsgMaxBlocks: 800
mailquota: 10000000
mailmsgquota: 6000
mailAllowedServiceAccess: +imaps:ALL$+pops:ALL$+smtps:ALL$+http:ALL
daServiceType: mail user
daServiceType: calendar user
```

Venus

```
mailMsgMaxBlocks: 700
mailquota: 8000000
mailmsgquota: 3000
mailAllowedServiceAccess: +imaps:ALL$+pops:ALL$+smtps:ALL$+http:ALL
daServiceType: mail user
daServiceType: calendar user
```

Earth

```
mailMsgMaxBlocks: 300
mailquota: 6291456
mailmsgquota: 2000
```

```
mailAllowedServiceAccess: +pop:ALL$+imap:ALL$+smtp:ALL$+http:ALL
daServiceType: mail user
daServiceType: calendar user
```

Mars

```
mailMsgMaxBlocks: 700
mailquota: 5242288
mailmsgquota: 3000
mailAllowedServiceAccess: +pop:ALL$+imap:ALL$+smtp:ALL$+http:ALL
daServiceType: mail user
daServiceType: calendar user
```

User Mail and IM Sample Template

The following sample template applies both mail and IM services.

mailimsample

```
Mail Services
mailMsgMaxBlocks: 800
mailquota: 9M
mailmsgquota: 6000
mailAllowedServiceAccess: +imaps:ALL$+pops:ALL$+smtps:ALL$+http:ALL
IM Services
inetUserStatus: active
daServiceType: mail user
daServiceType: im user
```

User Calendar and IM Sample Template

The following sample template applies both calendar and IM services. The template provides IM service attributes but no calendar service attributes.

calendarimsample

```
Calendar Services
N/A
IM Services
inetUserStatus: active
daServiceType: calendar user
daServiceType: im user
```

User Mail and Contacts Sample Template

The following sample template applies mail and contacts services.

mailcontactssample

```
Contacts Services
dn: cn=mailcontactssample,o=mailcontactsuser,o=cosTemplates,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
cn: mailcontactssample
nab: value
Mail Services
```

```
mailMsgMaxBlocks: 300
mailquota: 6291456
mailmsgquota: 2000
mailAllowedServiceAccess: +pop:ALL$+imap:ALL$+smtp:ALL$+http:ALL
daServiceType: mail user
daServiceType: contacts user
```

User Calendar and Contacts Sample Template

The following sample template applies calendar and contacts services. The template provides contacts service attributes but no calendar service attributes.

calendarcontactssample

```
Contacts Services
dn: cn=calendarcontactssample,o=calendarcontactsuser,o=cosTemplates,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
cn: calendarcontactssample
inetUserStatus: active
nab: value
Calendar Services
N/A
daServiceType: calendar user
daServiceType: contacts user
```

User Mail, Calendar, and IM Sample Template

The following sample template applies mail, calendar, and IM services. The template provides mail and IM service attributes but no calendar service attributes.

mailcalendarimsample

```
Mail Services
mailMsgMaxBlocks: 800
mailquota: 9M
mailmsgquota: 6000
mailAllowedServiceAccess: +imaps:ALL$+pops:ALL$+smtps:ALL$+http:ALL
Calendar Services
N/A
IM Services
inetUserStatus: active
daServiceType: mail user
daServiceType: calendar user
daServiceType: im user
```

User Mail, Calendar, IM, and Contacts Sample Templates

Neptune

```
Contacts Services
dn: cn=neptune,o=mailcalendarimcontactsuser,o=cosTemplates,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleobject
objectclass: cosTemplate
```

```
cn: neptune
Mail Services
mailMsgMaxBlocks: 800
mailquota: 10000000
mailmsgquota: 6000
mailAllowedServiceAccess:
+imap:ALL$+imaps:ALL$+pop:ALL$+pops:ALL$+smtp:ALL$+http:ALL
nab: value
inetUserStatus: active
Calendar Services
N/A
IM Services
inetUserStatus: active
daServiceType: calendar user
daServiceType: mail user
daServiceType: im user
daServiceType: contacts user
```

Group Mail Sample Templates

Atlantic

```
mailMsgMaxBlocks: 800
daServiceType: mail group
```

Pacific

```
mailMsgMaxBlocks: 900
daServiceType: mail group
```

Indian

```
mailMsgMaxBlocks: 1000
daServiceType: mail group
```

Arctic

```
mailMsgMaxBlocks: 1200
daServiceType: mail group
```

Group Calendar Sample Template

None (standardGroupCalendar)

There is no predefined Class-of-Service template that provides calendar service to groups and contains attribute values. Calendar service is provided without associated attributes.

Because no sample template exists, Delegated Administrator generates a default service package, without a template, directly from the Group Calendar Class-of-Service definition. Its name is the same as that of the Class-of-Service definition: standardGroupCalendar.

This service package provides calendar service (to groups) only.

Group Mail and Calendar Sample Templates

The following sample templates apply both mail and calendar service to groups.

Nile

```
mailMsgMaxBlocks: 1600
daServiceType: mail group
daServiceType: calendar group
```

Amazon

```
mailMsgMaxBlocks: 1800
daServiceType: mail group
daServiceType: calendar group
```

Thames

```
mailMsgMaxBlocks: 2000
daServiceType: mail group
daServiceType: calendar group
```

Danube

```
mailMsgMaxBlocks: 2200
daServiceType: mail group
daServiceType: calendar group
```

Class-of-Service Definitions

This release of Delegated Administrator provides a Class-of-Service definition for each type of service package:

- User mail service
- User calendar service
- User instant messaging service
- User contacts service
- User mail and calendar service
- User mail and im service
- User mail and contacts service
- User calendar and im service
- User calendar and contacts service
- User mail, calendar, and im service
- User mail, calendar, im, and contacts
- Group mail service
- Group calendar service
- Group mail and calendar service

When you configure Delegated Administrator, the Class-of-Service definitions are installed in the directory.

In each definition, the **daServiceType** attribute determines the type of service package with the following syntax:

```
daServiceType: service type target
```

where *service type* is **mail, calendar,** or **im,** and *target* is either **user** or **group**. If a Class-of-Service applies to multiple service types, **daServiceType** is listed multiple times.

Mail Service for Users

The user mail service is defined in a Class-of-Service definition called **standardUserMail**:

```
#
#  Definition for user mail service bundle
#
dn: cn=standardUserMail,ugldapbasedn
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=mailuser,o=cosTemplates,ugldapbasedn
cosSpecifier: inetCos
cosAttribute: mailAllowedServiceAccess
cosAttribute: mailMsgMaxBlocks
cosAttribute: mailquota
cosAttribute: mailmsgquota
daServiceType: mail user
NOTE: When the Delegated Administrator configuration program installs the
standardUserMail definition in the directory, the variable
ugldapbasedn, shown above, is replaced by your root suffix
(such as o=usergroup).
```

The **daServiceType** attribute defines this as a mail service for users.

Calendar Service for Users

The user calendar service is defined in a Class-of-Service definition called **standardUserCalendar**:

```
#
#  Definition for user calendar service bundle
#
dn: cn=standardUserCalendar,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=calendaruser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
cosAttribute: icsPreferredHost
cosAttribute: icsDWPHost
cosAttribute: icsFirstDay
daServiceType: calendar user
NOTE: When the Delegated Administrator configuration program installs the
standardUserCalendar definition in the directory, the variable
<ugldapbasedn>, shown above, is replaced by your root suffix
(such as o=usergroup).
```

The **daServiceType** attribute defines this as a calendar service for users.

Note: The calendar service definition also includes calendar attributes such as **icsPreferredHost**.

However, Delegated Administrator does not provide service-package templates that specify values for these attributes. The Delegated Administrator console provides one service package with calendar service only: the **standardUserCalendar** service package. This package does not include calendar attributes.

Instant Messaging Service for Users

The user IM service is defined in a Class-of-Service definition called **standardUserIM**:

```
#
# Definition for user im service bundle
#
dn: cn=standardUserIM,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=imuser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
cosAttribute: inetUserStatus
daServiceType: im user
```

Customers may add additional class-of-service entries like the following.

```
cosAttribute: jabberServiceStatus
cosAttribute: sunPresencePolicy
cosAttribute: sunIMPolicy
cosAttribute: sunFileTransferPolicy
cosAttribute: sunMediaPolicy
cosAttribute: sunFederatedXMPPDomainFilter
cosAttribute: sunAllowedJabberAccessPoint
cosAttribute: sunIMAllowedComponent
NOTE: When the Delegated Administrator configuration program installs the
standardUserIM definition in the directory, the variable
ugldapbasedn, shown above, is replaced by your root suffix
(such as o=usergroup).
```

The **daServiceType** attribute defines this as an instant messaging service for users.

Contacts Service for Users

The contacts service is defined in a Class-of-Service definition called **standardUserContacts**:

```
#
#  Definition for user contacts service bundle
#
dn: cn=standardUserContacts,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
```

```
objectclass: cosClassicDefinition
cosTemplateDn: o=contactsuser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
cosAttribute: nab
daServiceType: contacts user
```

The **daServiceType** attribute defines this as a contacts service for users.

Mail and Calendar Service for Users

The user mail and calendar service is defined in a Class-of-Service definition called **standardUserMailCalendar**:

```
#
#  Definition for user mail and user calendar service bundle
#
dn: cn=standardUserMailCalendar,ugldapbasedn
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=mailcalendaruser,o=cosTemplates,ugldapbasedn
cosSpecifier: inetCos
cosAttribute: icsPreferredHost
cosAttribute: icsDWPHost
cosAttribute: icsFirstDay
cosAttribute: icsQuota
cosAttribute: mailAllowedServiceAccess
cosAttribute: mailMsgMaxBlocks
cosAttribute: mailquota
cosAttribute: mailmsgquota
daServiceType: calendar user
daServiceType: mail user
NOTE: When the Delegated Administrator configuration program installs the
standardUserMailCalendar definition in the directory, the variable
ugldapbasedn, shown above, is replaced by your root suffix
(such as o=usergroup).
```

The two **daServiceType** attribute entries define this as a calendar service and mail service for users.

Mail and IM Service for Users

The user mail and IM service is defined in a Class-of-Service definition called **standardUserMailIm**:

```
#
# Definition for user mail and user im service bundle
#
dn: cn=standardUserMailIM,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=mailimuser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
```

```
cosAttribute: mailAllowedServiceAccess
cosAttribute: mailMsgMaxBlocks
cosAttribute: mailquota
cosAttribute: mailmsgquota
cosAttribute: inetUserStatus
daServiceType: mail user
daServiceType: im user
```

The two **daServiceType** attribute entries define this as a mail service and instant messaging service for users.

Mail and Contacts Service for Users

The user mail and contacts service is defined in a Class-of-Service definition called **standardUserMailContacts**:

```
#
# Definition for user contacts and mail service bundle
#
dn: cn=standardUserMailContacts,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=mailcontactsuser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
cosAttribute: nab
cosAttribute: mailAllowedServiceAccess
cosAttribute: mailMsgMaxBlocks
cosAttribute: mailquota
cosAttribute: mailmsgquota
daServiceType: mail user
daServiceType: contacts user
```

The two **daServiceType** attribute entries define this as a mail service and contacts service for users.

Calendar and IM Service for Users

The user calendar and IM service is defined in a Class-of-Service definition called **standardUserCalendarIm**:

```
#
# Definition for user calendar and user im service bundle
#
dn: cn=standardUserCalendarIM,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=calendarimuser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
cosAttribute: icsPreferredHost
cosAttribute: icsDWPHost
cosAttribute: icsFirstDay
cosAttribute: icsQuota
```

```
cosAttribute: inetUserStatus
daServiceType: calendar user
daServiceType: im user
```

The two **daServiceType** attribute entries define this as a calendar service and instant messaging service for users.

Calendar and Contacts Service for Users

The user calendar and contacts service is defined in a Class-of-Service definition called **standardUserCalendarContacts**:

```
#
# Definition for user calendar and user contacts service bundle
#
dn: cn=standardUserCalendarContacts,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=calendarcontactsuser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
cosAttribute: icsPreferredHost
cosAttribute: icsDWPHost
cosAttribute: icsFirstDay
cosAttribute: icsQuota
cosAttribute: inetUserStatus
cosAttribute: nab
daServiceType: calendar user
daServiceType: contacts user
```

The two **daServiceType** attribute entries define this as a calendar service and contacts service for users.

Mail, Calendar, and IM Service for Users

The user mail, calendar, and IM service is defined in a Class-of-Service definition called **standardUserMailCalendarIM**:

```
#
# Definition for user mail and user calendar and user im service bundle
#
dn: cn=standardUserMailCalendarIM,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=mailcalendarimuser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
cosAttribute: icsPreferredHost
cosAttribute: icsDWPHost
cosAttribute: icsFirstDay
cosAttribute: icsQuota
cosAttribute: mailAllowedServiceAccess
cosAttribute: mailMsgMaxBlocks
cosAttribute: mailquota
```

```
cosAttribute: mailmsgquota
cosAttribute: inetUserStatus
daServiceType: calendar user
daServiceType: mail user
daServiceType: im user
```

The three **daServiceType** attribute entries define this as a mail service, calendar service, and instant messaging service for users.

Mail, Calendar, IM, and Contacts Service for Users

The user mail, calendar, IM, and contacts service is defined in a Class-of-Service definition called **standardUserMailCalendarIMContacts**:

```
#
# Definition for user mail and user calendar and user im and user contacts service
bundle
#
dn: cn=standardUserMailCalendarIMContacts,<ugldapbasedn>
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=mailcalendarimcontactsuser,o=cosTemplates,<ugldapbasedn>
cosSpecifier: inetCos
cosAttribute: icsPreferredHost
cosAttribute: icsDWPHost
cosAttribute: icsFirstDay
cosAttribute: icsQuota
cosAttribute: mailAllowedServiceAccess
cosAttribute: mailMsgMaxBlocks
cosAttribute: mailquota
cosAttribute: mailmsgquota
cosAttribute: inetUserStatus
cosAttribute: nab
daServiceType: calendar user
daServiceType: mail user
daServiceType: im user
daServiceType: contacts user
```

The four **daServiceType** attribute entries define this as a mail service, calendar service, instant messaging service, and contacts service for users.

Mail Service for Groups

The group mail service is defined in a Class-of-Service definition called **standardGroupMail**:

```
#
#  Definition for group mail service bundle
#
dn: cn=standardGroupMail,ugldapbasedn
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
```

```
cosTemplateDn: o=mailgroup,o=cosTemplates,ugldapbasedn
cosSpecifier: inetCos
cosAttribute: mailMsgMaxBlocks
daServiceType: mail group
NOTE: When the Delegated Administrator configuration program installs the
standardGroupMail definition in the directory, the variable ugldapbasedn,
shown above, is replaced by your root suffix (such as o=usergroup).
```

The **daServiceType** attribute defines this as a mail service for groups.

Calendar Service for Groups

The group calendar service is defined in a Class-of-Service definition called **standardGroupCalendar**:

```
#
#  Definition for group calendar service bundle
#
dn: cn=standardGroupCalendar,ugldapbasedn
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
objectclass: cosClassicDefinition
cosTemplateDn: o=calendargroup,o=cosTemplates,ugldapbasedn
cosSpecifier: inetCos
cosAttribute: icsdoublebooking
cosAttribute: icsautoaccept
daServiceType: calendar group
NOTE: When the Delegated Administrator configuration program installs the
standardGroupCalendar definition in the directory, the variable ugldapbasedn,
shown above, is replaced by your root suffix (such as o=usergroup).
```

The **daServiceType** attribute defines this as a calendar service for groups.

Note: The calendar service definition also includes calendar attributes such as **icsdoublebooking**.

However, Delegated Administrator does not provide service-package templates that specify values for these attributes. The Delegated Administrator console provides one service package for groups with calendar service only: the **standardGroupCalendar** service package. This package does not include calendar attributes.

Mail and Calendar Service for Groups

The user mail and calendar service is defined in a Class-of-Service definition called **standardGroupMailCalendar**:

```
#
#  Definition for group mail and group calendar service bundle
#
dn: cn=standardGroupMailCalendar,ugldapbasedn
changetype: add
objectclass: top
objectclass: LDAPsubentry
objectclass: extensibleObject
objectclass: cosSuperDefinition
```

```
objectclass: cosClassicDefinition
cosTemplateDn: o=mailcalendargroup,o=cosTemplates,ugldapbasedn
cosSpecifier: inetCos
cosAttribute: mgrpMsgMaxSize
cosAttribute: mailMsgMaxBlocks
daServiceType: calendar group
daServiceType: mail group
NOTE: When the Delegated Administrator configuration program installs the
standardGroupMailCalendar definition in the directory, the variable
ugldapbasedn, shown above, is replaced by your root suffix
(such as o=usergroup).
```

The two **daServiceType** attribute entries define this as a calendar service and mail service for groups.

Location of Class-of-Service Definitions and Packages

In the LDAP Directory Information Tree (DIT), the Class-of-Service definitions are located in a node directly under the root suffix. Because they are stored at the top of the DIT, the service packages can be assigned to all user entries in the directory. Figure A–1 shows the location of the mail and calendar service definitions and packages in the DIT.

Figure A–1 Location of Mail and Calendar Class-of-Service Definitions and Packages in the Directory Tree

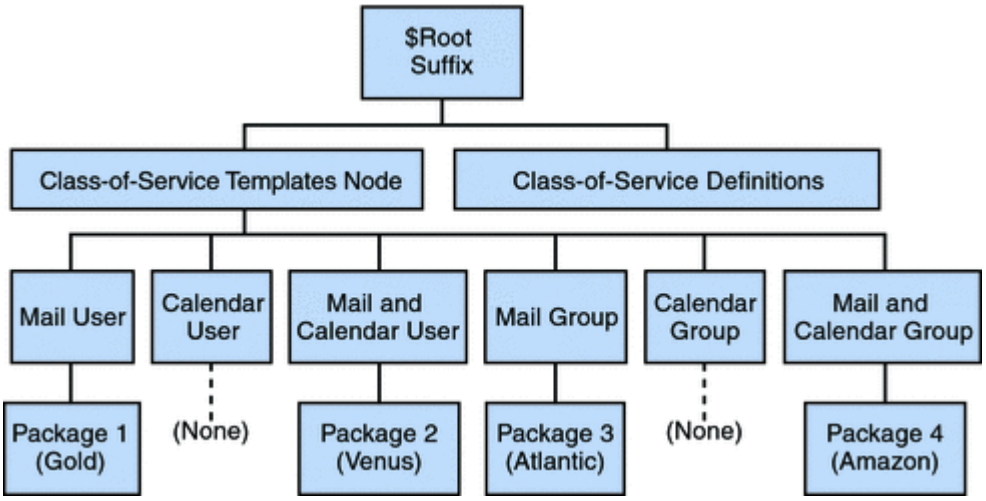


Figure A–2 shows the location of the IM service definitions and packages in the DIT.

Figure A–2 Location of IM Class-of-Service Definitions and Packages in the Directory Tree

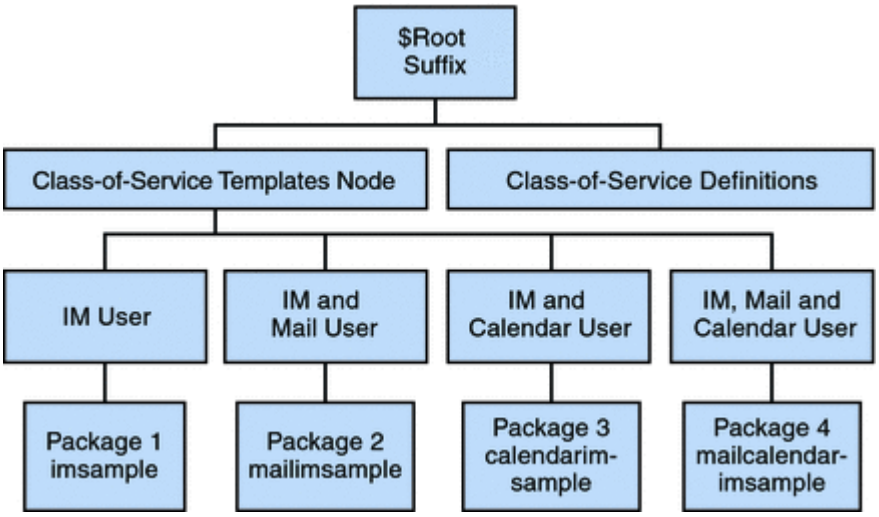
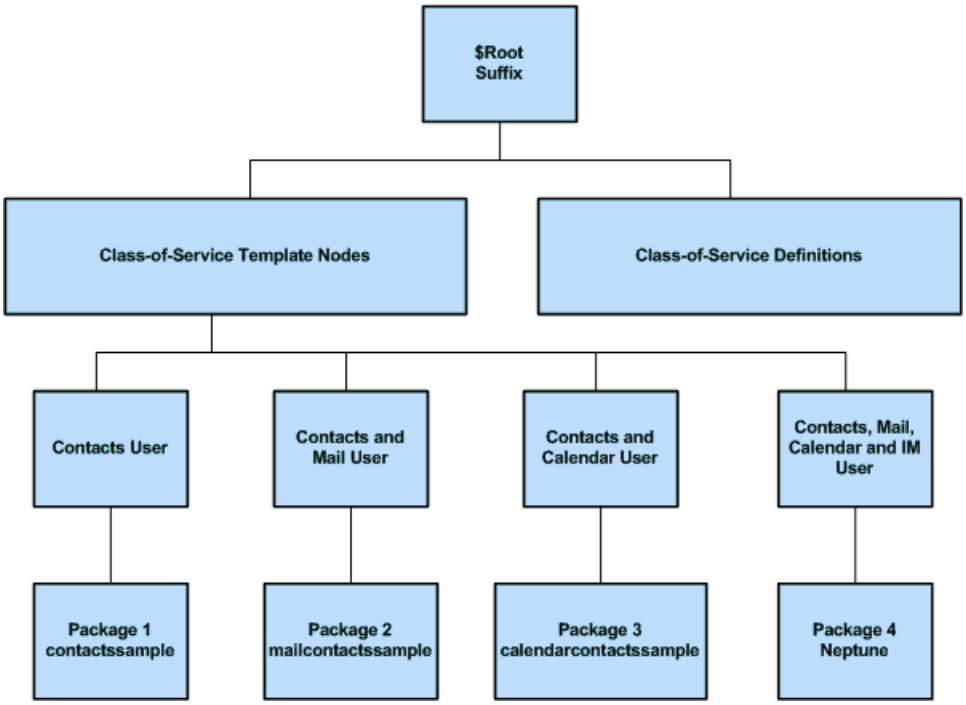


Figure A–3 shows the location of the contacts service definitions and packages in the DIT.

Figure A–3 Location of Contacts Class-of-Service Definitions and Packages in the Directory Tree



Each type of Class-of-Service template is located under its own node. Thus, a template providing mail service to users is located under the Mail User node. This structure enables Delegated Administrator to use the correct Class-of-Service definition (such as **standardUserMail**) when it assigns a service package to a user or group.

Delegated Administrator uses the classic Class-of-Service definition.

For more information about the Class-of-Service mechanism, see Defining Class-of-Service (CoS) in *Sun Java System Directory Server Administration Guide*.

The *Sun Java System Directory Server Administration Guide* also describes related topics such as determining which service attribute value takes precedence if an attribute defined in a service package assigned to a user already exists in that individual user entry.

Viewing the Service Packages in LDAP

The COS templates are stored as LDAP Subentries, so you need to filter explicitly for the **objectclass=LDAPsubentry** when you use **ldapsearch** to find the service packages.

```
# ldapsearch -D "cn=Directory Manager" -w password -b
"o=costemplates,<ugldapbasedn>" "(|(objectclass=*)(objectclass=LDAPsubentry))"
```

B

Attribute Values

This appendix lists the attribute values that can be used with the **-P**, **-R**, and **-T** options for multiple commands in Oracle Communications Delegated Administrator.

Attribute Values

Table B–1 lists the attributes that can be used with the **-P** option for the following commands: **commadmin domain create** and **commadmin domain modify**. The attributes are either bit-oriented attributes or multivalued attributes. For more information on these commands, see "Delegated Administrator Reference".

Table B–1 Attributes for -P Option

Attribute	Value	Description
createLowerCase	yes/no	Specifies whether a lowercase calendar is to be created for a user. Also, when looking up calendar, whether to lookup lowercase calendars or not.
filterPrivateEvents	yes/no	Specifies whether to filter the private or confidential events when querying the server
fbIncludeDefCal	yes/no	Specifies whether user~s default calendar is included in user~s **freebusy-calendar-list**.
subIncludeDefCal	yes/no	Specifies whether the user~s default calendar is to be included in user~s **subscribed-calendar-list** or not.
resourceDefaultAcl	yes/no	Specifies whether to use the default ACL for resource calendars.
calmasterCred	string	Credentials of user specified as the Calendar Server administrator.
calmasterUid	string	User ID of the Calendar Server Administrator defined by **service.admin.calmaster.userid**.
calmasterAccessOverride	yes/no	Specifies whether the Calendar Server administrator can override access control.
setPublicRead	yes/no	Sets the default user calendars to public read or private write. If **no** is selected, sets user calendars to private read or private write.
uiBaseUrl	string	BaseServerAddress, for example, **"https://proxyserver/"**
uiConfigFile	string	Configuration file for the user interface.
uiProxyUrl	string	Proxy Server Address to append in the HTML user interfaces JavaScript file, for example, **"https://web_portal.iplanet.com/"**

Table B–1 (Cont.) Attributes for -P Option

Attribute	Value	Description
domainAccess	string	Access control string for domain. Used in cross domain searching. An access control information (ACI) string comprises one or more ACI segments separated by semicolons. The ACI is used in cross-domain searches to permit external domains to search this domain. The ACI string may include specified external domain name(s). For more information about Calendar Server ACIs, see the discussion about access control in *Calendar Server WCAP Developer's Guide*.
uiAllowAnyone	yes/no	Specifies whether to allow the HTML user interface to show and use the **Everybody** ACL.
allowProxyLogin	yes/no	Specify whether to allow proxy login

Table B–2 lists attributes that can be used with the **-R** option for the following commands: **commadmin domain create** and **commadmin domain modify**. The attributes have a bit-oriented value.For more information on these commands, see "Delegated Administrator Reference".

For information about WCAP and the WCAP **set-userprefs** command, see the *Calendar Server WCAP Developer's Guide*.

Table B–2 Attributes for -R Option

Attribute	Value	Description
allowUserDoubleBook	bit 8	Allows this calendar to be scheduled more than once for the same time slot.
allowResourceDoubleBook	bit 9	Allows this resource calendar to be scheduled more than once for the same time slot.
allowModifyUserPreferences	bit 4	Allows the Calendar Server administrator to modify user preferences for users.
allowModifyPassword	bit 5	Allows users to change their password using this server.
allowCalendarCreation	bit 0	Allows calendars to be created.
allowCalendarDeletion	bit 1	Allows calendars to be deleted.
allowPublicWritableCalendars	bit 2	Allows users to own publicly writable calendars.
allowSetCn	bit 10	Allows **set-userprefs.wcap** to modify the **cn** user preference.
allowSetGivenName	bit 11	Allows **set_userprefs.wcap** to modify the **givenname** user preference.
allowSetGivenMail	bit 12	Allows **set_userprefs.wcap** to modify the **mail** user preference.
allowSetPrefLang	bit 13	Allows **set_userprefs.wcap** to modify the **preferredlanguage** user preference.
allowSetSn	bit 14	Allows **set-userprefs.wcap** to modify the **sn** user preference.
allowGroupDoubleBook	bit 15	Allows this group calendar to be double-booked (scheduled more than once for the same time slot).

C

Calendar Time Zones

This appendix lists the time zone strings that can be used with the **-T** time zone option for multiple commands in Oracle Communications Delegated Administrator.

About Calendar Time Zones

The following time zone strings can be used with the **-T** time zone option for the **commadmin domain create**, **commadmin domain modify**, **commadmin resource create**, **commadmin resource modify**, **commadmin user create**, and **commadmin user modify** commands. For more information on these commands, see "Delegated Administrator Reference".

You also can add a time zone and set it as the default time zone. See "Customizing Delegated Administrator" for details.

- Africa/Cairo
- Africa/Casablanca
- Africa/Johannesburg
- Africa/Lagos
- Africa/Tripoli
- Africa/Windhoek
- America/Adak
- America/Anchorage
- America/Buenos_Aires
- America/Caracas
- America/Chicago
- America/Costa_Rica
- America/Cuiaba
- America/Denver
- America/Godthab
- America/Grand_Turk
- America/Halifax
- America/Havana
- America/Indianapolis

- America/Los_Angeles
- America/Miquelon
- America/New_York
- America/Phoenix
- America/Port-au-Prince
- America/Santiago
- America/Sao_Paulo
- America/St_Johns
- Asia/Alma-Ata
- Asia/Amman
- Asia/Anadyr
- Asia/Aqtau
- Asia/Aqtobe
- Asia/Baku
- Asia/Bangkok
- Asia/Beirut
- Asia/Bishkek
- Asia/Calcutta
- Asia/Dacca
- Asia/Irkutsk
- Asia/Jerusalem
- Asia/Kabul
- Asia/Kamchatka
- Asia/Karachi
- Asia/Katmandu
- Asia/Krasnoyarsk
- Asia/Magadan
- Asia/Novosibirsk
- Asia/Rangoon
- Asia/Riyadh
- Asia/Shanghai
- Asia/Tokyo
- Asia/Ulan_Bator
- Asia/Vladivostok
- Asia/Yakutsk
- Asia/Yekaterinburg
- Asia/Yerevan

- Atlantic/Azores
- Atlantic/Cape_Verde
- Atlantic/South_Georgia
- Atlantic/Stanley
- Australia/Adelaide
- Australia/Brisbane
- Australia/Darwin
- Australia/Hobart
- Australia/Lord_Howe
- Australia/Sydney
- Europe/Bucharest
- Europe/Istanbul
- Europe/London
- Europe/Minsk
- Europe/Moscow
- Europe/Paris
- Europe/Riga
- Europe/Samara
- Europe/Simferopol
- Europe/Warsaw
- Pacific/Apia
- Pacific/Auckland
- Pacific/Chatham
- Pacific/Easter
- Pacific/Fiji
- Pacific/Gambier
- Pacific/Guadalcanal
- Pacific/Honolulu
- Pacific/Kiritimati
- Pacific/Marquesas
- Pacific/Norfolk
- Pacific/Noumea
- Pacific/Pitcairn
- Pacific/Rarotonga
- Pacific/Tongatapu

D

Delegated Administrator Files and Directories

This appendix describes Oracle Communications Delegated Administrator property files.

Configuration and Data Files

- The default directory for configuration and data files is **/var/opt/SUNWcomm**.

> **Note:** Select the directory where you want to store the Delegated Administrator configuration and data files. The default configuration directory is **/var/opt/SUNWcomm**. This directory should be separate from the *DelegatedAdmin_home* directory, which is **/opt/sun/comms/da** by default.

For more information, see *Delegated Administrator Installation and Configuration Guide*.

Legacy Directory Conventions

Table D–1 lists legacy directory conventions.

Table D–1 Legacy directory conventions

Platform	Communications Suite 5	Communications Suite 6
DelegatedAdmin-base	/opt/SUNWcomm	/opt/sun/comms/commcli
DelegatedAdminConfig-base	/var/opt/SUNWcomm	/var/opt/sun/comms/commcli

These directory conventions can be found in older documents on docs.sun.com.

For more information, see *Delegated Administrator Installation and Configuration Guide*.

After configuration, the *DelegatedAdminConfig-base* is linked to *DelegatedAdmin-base***/data/**. So the term *DelegatedAdminConfig-base* is used only for the configuration (**config-commda**) documentation, to provide the actual path for configuration and data files.

```
Please specify the directory where the configuration and data files should be
installed. It is recommended that you configure Delegated Administrator in the
directory suggested by the configurator program.

Specify the directory in which the configuration and data files should be
installed [/var/opt/SUNWcomm] {"<" goes back, "!" exits}:
```

(Source: running **config-commda** in text mode)

Delegated Administrator Utility

- **cli-usrprefs.properties**

Location

*DelegatedAdmin_home***/config/**

The Communications Suite 6 default location for both Solaris OS or Red Hat Linux:

```
/opt/sun/comms/da/data/config/cli-usrprefs.properties
```

The Communications Suite 5 default location for Solaris OS:

```
/opt/SUNWcomm/data/config/cli-usrprefs.properties
```

The Delegated Administrator utility (**commadmin**) is not a web container, so there are no original and deployed locations. Instead, the preference is found by following *DelegatedAdmin***/config/**, a softlink to the directory with the properties file:

```
# cd /opt/SUNWcomm/
# ls -ld config
lrwxrwxrwx   1 root     root    27 May 26 02:27 config -> /var/opt/SUNWcomm/config
```

Contents

```
IdentityServerHost=ISHost
IdentityServerPort=ISPort
IdentityServerSSLPort=ISDefSSLPort
DefaultDomain=defEmailDomain
appName=commcli
```

These are the default values used by **commadmin**. They are the equivalent of:

```
./commadmin -d defEmailDomain -X ISHost -p ISPort
```

Delegated Administrator Console

This section reviews the configuration and log files of the Delegated Administrator console.

Configuration Files

- **daconfig.properties**
- **logger.properties**
- **Resources.properties**
- **Security.properties**

Location: *DelegatedAdmin_ home***/data/da/WEB-INF/classes/com/sun/comm/da/resources/**

Log File

- Default log file name: **da.log**
- Default location: *DelegatedAdmin_home***/log/**

Configured by **logger.properties** file:

```
da.logging.enable=YES
da.log.file=/opt/SUNWcomm/log/da.log
```

(Example from: Communications Suite 5)

Delegated Administrator Server

This section reviews the files related to Delegated Administrator server.

"resource" Configuration File

- **resource.properties**

See "Original (Standard) Locations of the Configuration Files" for the default location of the **resource.properties** file.

Location: *DelegatedAdmin_home***/data/WEB-INF/classes/sun/comm/cli/server/servlet**

"Server" configuration file (DA 7 only)

- **serverconfig.properties**

> **Note:** Most of the configuration related properties are moved from **resource.properties** to the **serverconfig.properties** file.

Location: *DelegatedAdmin_home***/data/WEB-INF/classes/sun/comm/cli/server/servlet**

Source: *Delegated Administrator Installation and Configuration Guide*.

Log File

Not enabled by default.

The default location is: **/tmp/commcli.log**

Delegated Administrator Configuration (config-commda)

This section reviews the log files, state files and patch log directory related to the Delegated Administrator configuration process.

Log File

A log file for the **config-commda** program is created in the **/opt/SUNWcomm/install** directory. The name of the log file is **commda-config_***YYYYMMDDHHMMSS***.log**, where *YYYYMMDDHHMMSS* identifies the 4-digit year, month, date, hour, minute, and second of the configuration.

State File

The silent installation **saveState** file is stored in the **/opt/SUNWcomm/data/setup/commda-config_***YYYYMMDDHHMMSS*/ directory, where *YYYYMMDDHHMMSS* identifies the 4-digit year, month, date, hour, minute, and second of the **saveState** file.

Patch Log Directory

The *DelegatedAdmin_home***/install/patch/** directory contains the pre-patch and post-patch output from running **patchadd**.

E

Delegated Administrator Reference

This appendix provides command-line descriptions of individual **commadmin** commands in Oracle Communications Delegated Administrator.

commadmin admin add

The **commadmin admin add** command grants the Organization Administrators privileges to a user for a particular domain. Only a Top-Level Administrator or an ISP administrator can execute this command.

Syntax

```
commadmin admin add -D login -l login -n domain password -d domain [-h] [-i
inputfile ] [-p DA server port] [-X DA server host name] [-?] [-s] [-v] [-V]
```

Options

Table E–1 displays the options that are mandatory:

Table E–1 commadmin admin add Mandatory Options

Option	Description
-D *login*	The user ID of the Top-Level Administrator.
-l *login*	The user ID of the user to whom you want to grant organization administrative privileges. The user should be present in the directory and be a part of the domain specified by the **-d** option.
-n *domain*	The domain of the Top-Level Administrator. If not specified, the default domain stored in the **cli-usrprefs.properties** file is used.
-d *domain*	The domain to which you want to grant administrative privileges. If not specified, the domain specified by the **-n** option is used.

Table E–2 displays the options that are non-mandatory:

Table E–2 commadmin admin add Non-Mandatory Options

Option	Description
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Use this option to specify an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.

Table E–2 (Cont.) commadmin admin add Non-Mandatory Options

Option	Description
-X *DA server host name*	Specify the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used
-h, **-?**	Prints command usage syntax.
-V	Prints information about the command and its version.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.

Examples

The following grants Organization Administrator privileges to the user with the user ID **admin1**.

```
commadmin admin add -D chris -n example.com -l admin1 -d example.com
```

The following grants Organization Administrator privileges to the user with the user ID **admin2** for the **domain florizel.com**.

```
commadmin admin add -D chris -l admin2 -n example.com -d example.com
```

commadmin admin remove

The **commadmin admin remove** command removes the Organization Administrator privileges from an existing Organization Administrator. Only a Top-Level Administrator can execute this command.

To remove Organization Administrator privileges from multiple users, use the **-i** option.

Syntax

```
commadmin admin remove -D login -l login -n domain -d domain name [-h] [-?] [-i]
inputfile [-p] DA server port [-X] DA server host name [-s] [-v] [-V]
```

Options

Table E–3 displays the options that are mandatory:

Table E–3 commadmin admin remove Mandatory Options

Option	Description
-D *login*	The user ID of the Top-Level Administrator.
-l *login*	The user ID of the user whose administrator privileges need to be revoked.
-n *domain*	The domain of the Top-Level Administrator.
-d *domain name*	The domain to which administrator privileges are revoked. If **-d** is not specified, the domain specified by **-n** is used.

Table E–4 displays the options that are non-mandatory:

Table E–4 commadmin admin remove Non-Mandatory Options

Option	Description
-h, **-?**	Displays command usage syntax.

Table E–4 (Cont.) commadmin admin remove Non-Mandatory Options

Option	Description
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Use this option to specify an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-X *DA server host name*	Specify the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.
-V	Displays information about the command and its version.

Example

The following command removes Organization Administrator privileges from the administrator with user ID **admin5**:

```
commadmin admin remove -D chris -n example.com -l admin5 -d test.com
```

commadmin admin search

The **commadmin admin search** command searches and displays a specific or all Organization Administrators in a domain.

Syntax

```
commadmin admin search -D login -n -domain [-l login] [-d domain]
```

Options

Table E–5 displays the options that are mandatory:

Table E–5 commadmin admin search Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-n *domain*	The domain of the user specified with the **-D** option.

Table E–6 displays the options that are non-mandatory:

Table E–6 commadmin admin search Non-Mandatory Options

Option	Description
-l *login*	The user ID of the Organization Administrator searched for. If **-l** is not specified or **-l** is specified with the wildcard operator (**-l*** or **-l '*'**) all Organization Administrators of the domain are displayed.
-d *domain*	Searches for users who have Organization Administrator privileges for the specified domain. If **-d** is not specified, the domain specified by **-n** is used.

Example

To search for all Organization Administrators of the **test.com** domain:

```
commadmin admin search -D chris -n example.com -d test.com
```

commadmin Command Definition

This page provides general information about how to use the **commadmin** command. It includes the following topics:

Execution Modes

The command line execution has three possible modes:

- Execute with options specified in a file

  ```
  commadmin object task -i inputfile
  ```

  Analyzes *inputfile* and executes it. The input file should be in the format "Command File Format".

- Immediate or shell execution

  ```
  commadmin object task [options]
  ```

  If all needed information is provided, the command will execute immediately.

- Interactive

  ```
  commadmin object task
  ```

  The command contains some, but not all of the required options for command execution. The administrator is queried for the remainder of the options and attributes.

Command results

When an operation succeeds, the following message appears:

```
OK
```

If a failure occurs, the following message appears:

```
FAIL

message
```

Where *message* displays the error text.

Command File Format

The options can be specified within a file, using the **-i** option.

Within the file discussed in "Execution Modes", option names are separated from option values by white space. The option value begins with the first non-white space character and extends to the end-of-line character. Option sets are separated by blank lines.

The general syntax is:

```
option name [option value, if any]
```

```
option name [option value, if any]
...
option name [option value, if any]
blank line
option name [option value, if any]
option name [option value, if any]
...
option name [option value, if any]
```

The option value given in the command line becomes the default for each option set. Alternatively, these options can be specified for each option set. The value then overrides any default specified on the command line.

Following is an example of the format and syntax for the file specified by the **-i** option for the **commadmin user create** command.

```
l newuser1
F new
L user1

l newuser2
F new
L user2

l newuser3
F new
L user3

and so forth...
```

Mandatory commadmin Options

Table E–7 displays the options that are the mandatory options used for authenticating the administrator or the user.

> **Note:** Beginning with Communications Suite 7 Update 2, the **-w** *password* option is no longer used. Instead, the **commadmin** command has been made more secure by the removal of ability to specify the administrative password on the command line or to use a password file. All **commadmin** passwords must now be typed using a no-echo prompt.

Table E–7 Mandatory Options for Administrator or User Authentication

Option	Description
-D *userid*	User ID used to bind to the directory.
-n *domain*	The domain the administrator belongs to. (For more information, see the Note shown below this table.)

Note: Configuring Delegated Administrator for Directory Access Through Access Manager (Legacy Mode)

This note applies only if you choose to access the directory using Access Manager in Legacy mode when you first configure Delegated Administrator after installation.

In this case, the Access Manager Host (**-X**), Access Manager Port (**-p**), and the default domain (**-n**) values are specified when you run the configuration program, **config-commad**, and store them in the **cli-usrprefs.properties** file.

If the **-X**, **-p**, and **-n** options are not specified at the time when a **commadmin** command is executed, their values are taken from *Delegated Administrator property* files.

commadmin debug log

The **commadmin debug log** command creates a Delegated Administrator server log file that contains error messages generated by the Delegated Administrator servlets installed on the web container. If the log file already exists, the new error messages are appended to the end of the existing file. If the log file does not exist, a new file is created.

This command sets the debug mode on Delegated Administrator. Debug mode is useful for analyzing problems that affect the Delegated Administrator Console and the Delegated Administrator Utility. For analyzing problems with the Delegated Administrator Utility (**commadmin**) client itself, use the **-v** option, and capture the command output.

Syntax

```
commadmin debug log -D login -n domain -t [ on|off ] [ -f debug-log-file ] [-s]
```

Options

Table E–8 displays the options that are mandatory:

Table E–8 commadmin debug log Mandatory Options

Option	Description
-D *login*	The user ID of the Top-Level Administrator.
-n *domain*	The domain of the Top-Level Administrator.
-t [*on* | *off*]	Toggles between turning on the debug log and turning it off. The value *on* causes the server to start writing error messages to the log. The value *off* causes the server to stop writing error messages to the log.

Table E–9 displays the options that is non-mandatory:

Table E–9 commadmin debug log Non-Mandatory Options

Option	Description

-f *debug-log-file*	The full path where the log will be created, including the file name of the log. The path must be one of the following two directories:
	`/tmp/`
	`/var/tmp/`
	
	The *debug-log-file* can be any file name. If the **-f** option is not specified, the default value is **/tmp/commcli.log**.

-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.

Example

To create a new debug log, enter:

```
commadmin debug log -D paul -n example.com -t on -f /tmp/debug.log
```

To turn off logging, enter:

```
commadmin debug log -D paul -n example.com -t off
```

Note: You do not have to specify the file name when you turn off the log.

commadmin domain create

The **commadmin domain create** command creates a single domain in the LDAP directory. To create multiple domains, use the **-i** option.

Syntax

```
commadmin domain create -D login -d domain name -n domain [-A [+]
attributename:value] [-h] [-?] [-i inputfile] [-o organization RDN] [-p DA server
port]

[-s] [-v] [-V] [-X DA server host name] [-S mail -H preferred mail host] [-S cal
[-B backend calendar data server] [-C searchable domains]

[-g access control string] [-P propertyname[:value]] [-R right[:value]] [-T
calendar time zone string] ] [-S im]
```

Options

Table E–10 displays the options that are mandatory:

Table E–10 commadmin domain create Mandatory Options

Option	Description
-D *login*	The user ID of the Top-Level Administrator.
-d *domain name*	DNS domain name of the domain that is being created.
-n *domain*	The domain of the Top-Level Administrator.

Table E–11 displays the options that options are non-mandatory:

Table E–11 commadmin domain create Non-Mandatory Options

Option	Description
-A [+] *attributename*:*value*	An attribute to modify. The *attributename* is defined in the LDAP schema and the *value* specified replaces any and all current values for this attribute in the directory. Repeat this option to modify multiple attributes at the same time, or to specify multiple values for the same attribute. A + before the *attributename* indicates adding the value to the current list of attributes. If the action value (+) is not specified, the default action is to add the existing value.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-o *organization RDN*	Specifies the organization RDN for the domain. For example, **o=example.example.com.** If this option is not specified, the organization is created under the **osisuffix,** with **o=** the name of the domain, **o=osiSuffix.**
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

-S *service*	Specifies the service or services to be added to the domain.
	service can have the value of a single service or multiple services. The valid *service* values are **mail**, **cal**, and **im**. These values are case-insensitive.
	If the **-S mail** option is specified, then the **-H** option must be specified.
	Can be listed as a comma-separated list.
	For Example:
	`-S mail,cal,im`
	Delegated Administrator 7:
	`-S im`
	AM provisioning mode: A domain is created with the services mentioned, depending on the value of the particular service definition present in the configuration file of Access Manager.

Table E–12 displays the options that, if the **-S mail** option is specified, are non-mandatory options that are allowed:

Table E–12 commadmin domain create Options Allowed if -S mail Specified

Option	Description
-H *preferred mail host*	The preferred mail host for the domain. The host must be a fully qualified host name, for example, **mailhost.example.com**. This option is mandatory if the **-S mail** option is specified.

Table E–13 displays the options that, if the **-S cal** option is specified, are the non-mandatory options that are allowed:

Table E–13 commadmin domain create Options Allowed if -S cal Specified

Option	Description
-B *back-end calendar data server*	Specifies the default back-end host assigned to a user or resource in a domain.
-C *searchable domains*	Specifies the domains to be searched when looking for calendars or users.
-g *access control string*	Specifies the Access Control List (ACL) for a newly created user calendar.
-P *propertyname*[:*value*]	Sets values for multi-valued and bit oriented attributes. See "Attribute Values" for attributes, their descriptions and values.
-R *right*[:*value*]	Sets calendar domain attribute **icsAllowRights**. The attribute holds a bitmap value. See "Attribute Values" for a list of attributes, their value(s), and description(s).
-T *calendar time zone string*	Specifies the time zone ID used when importing files. See Calendar Time Zone Strings for a list of the valid time zone strings.

Example

To create a new domain with mail, calendar, and instant messaging services, enter:

```
commadmin domain create -D chris -d example.com -n example.com -S mail,cal,im -H
mailhost.example.com
```

Note: Product Version Information

Current Features: Features documented on this page were introduced in the following product release versions of Delegated Administrator 7:

- Direct LDAP domain creation (no dependence on AM)
- Support for IM service

commadmin domain delete

The **commadmin domain delete** command marks a single hosted domain as deleted from the server. To mark multiple hosted domains as deleted, use the **-i** option.

When you mark a domain as deleted, all user and group entries in the domain are marked as deleted.

The "commadmin domain purge" command will permanently remove the domain.

To disable Organization Administrators usage of a service like calendar service or mail service, use the **-S** option. Here **S** is in uppercase.

Syntax

```
commadmin domain delete -D login -d domain name -n domain [-h] [-?] [-i inputfile]
[-p DA server port] [-s] [-S service] [-v] [-V] [-X DA server host name]
```

Options

Table E–14 displays the options that are mandatory:

Table E–14 commadmin domain delete Mandatory Options

Option	Description
-D *login*	The user ID of the Top-Level Administrator.
-d *domain name*	The DNS domain name that is being deleted. If **-d** is not specified, the domain specified by **-n** is used.
-n *domain*	The domain of the Top-Level Administrator.

Table E–15 displays the options that are non-mandatory:

Table E–15 commadmin domain delete Non-Mandatory Options

Option	Description
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured during installation.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-S *service*	Modifies the value of the specified service status attribute value to delete. Multiple services are separated by a comma. The valid *service* values are **mail**, and **cal**, and **im**. These values are case-insensitive.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

To delete an existing domain:

```
commadmin domain delete -D chris -d example.com -n example.com
```

To delete just the mail service from the **florizel.com** domain:

```
commadmin domain delete -D chris -d example.com -n example.com -S mail
```

commadmin domain modify

The **commadmin domain modify** command modifies attributes of a single domain directory entry. To modify multiple domains, use the **-i** option.

Syntax

```
commadmin domain modify -D login -d domain -n domain
   [-A [+|-]attributename:value] [-h] [?] [-i inputfile] [-p DA server port] [-s]
[-v] [-V]
   [-X DA server host name]
   [-S mail -H preferred mailhost]
   [-S cal [-g access string] [-C cross domain search domains] [-B backend
calendar data server]
   [-P [action] propertyname[:value]] [-R propertyname[:value]] [-T calendar time
zone string]]
   [-S im]
```

Options

Table E–16 displays the options that are mandatory:

Table E–16 commadmin domain modify Mandatory Options

Option	Description
-D *login*	The user ID of the Top-Level Administrator.
-d *domain*	The DNS domain name to be modified. If **-d** is not specified, the domain specified by **-n** is used.
-n *domain*	The domain of the Top-Level Administrator.

Table E–17 displays the options that are non-mandatory:

Table E–17 commadmin domain modify Non-Mandatory Options

Option	Description
-A [+ *or* -] *attributename:value*	An attribute to modify. The *attributename* is defined in the LDAP schema and *value* replaces any and all current values for this attribute in the directory. Repeat this option to modify multiple attributes at the same time, or to specify multiple values for the same attribute. A + before the *attributename* indicates adding the value to the current list of attributes. A - indicates removing the value. If the - is used, it must be preceded by two backslashes if the command is specified on the command line. If the option is provided within an input file, one backslash must precede the - sign. If the action value (+ or -) is not specified, the default action is to replace the existing value.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.
-V	Prints information about the command and its version.

Table E–17 (Cont.) commadmin domain modify Non-Mandatory Options

Option	Description
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.
-S *service*	Adds the specified service or services to the domain during modification. The valid *service* values are **mail**, and **cal**, and **im**. These values are case-insensitive. The services listed with the **-S** option are separated by a comma. If **-S mail** is specified, then the **-H** option must be specified.

Table E–18 displays the options that, if the **-S mail** option is specified, are the non-mandatory options that are allowed:

Table E–18 commadmin domain modify Options Allowed if -S mail Specified

Option	Description
-H *preferred mail host*	The preferred mail host for the domain. This option is mandatory if the **-S mail** option is specified.

Table E–19 displays the options that, if the **-S cal** option is specified, are the non-mandatory options that are allowed:

Table E–19 commadmin domain modify Options Allowed if -S cal Specified

Option	Description
-B *back-end calendar data server*	The default back-end host assigned to a user or resource in a domain.
-C *cross domain search domains*	Specifies the domains to be searched when looking for calendars or users.
-g *access string*	Specifies the Access Control List (ACL) for newly created user calendar.
-P [*action*]*propertyname*[:*value*]	Sets the values for multi-valued and bit oriented attributes. See the Attribute Values tables in *Delegated Administrator System Administrator's Guide* for the descriptions and values of *propertyname*.
-T *calendar time zone string*	Time zone ID used when importing files. See Calendar Time Zone Strings for a list of the valid time zone strings.
-R *propertyname*[:*value*]	Sets calendar domain attribute **icsAllowRights**. The attribute holds a bitmap value. See the Attribute Values tables in *Delegated Administrator System Administrator's Guide* for a list of property names, their value, and description.

Example

To modify an existing domain:

```
commadmin domain modify -D chris -n example.com -d example.com -A
preferredmailhost:test.example.com
```

commadmin domain purge

The **commadmin domain purge** command permanently removes all entries or service of entries that have been marked for removal. This can include domains, users, groups, and resources.

As part of periodic maintenance operations, use the **commadmin domain purge** command to remove all entries that have been deleted for a time period that is longer than the specified delay period (grace period).

You can perform a purge at any time by invoking the command in **commadmin**. There is no equivalent in Delegated Administrator Console.

If the **-d*** option is specified, all domains are searched for users and domains that are marked as deleted. Users that are marked as deleted will be purged from their domain, but the domain will not be purged unless it is also marked as deleted. If a domain is marked as deleted, it will be purged along with all users within that domain.

When you invoke the command, the directory is searched and a list of domains are created whose entries include domains that have been marked for deletion longer than the specified grace period. The default value for the grace period is set to **5** days.

After a service has been marked as deleted, a command that removes resources such as mailboxes or calendars must be run before the service can be purged from the directory:

- For mail services, run the **msuserpurge** command.
- For information about the command see *Messaging Server Reference*.
- For calendar services, run the **davadmin** command for Calendar Server 7. Run the **csclean** command for Calendar Server 6.
- For information about the **davadmin** command see *Calendar Server System Administrator's Guide.* For information about the **csclean** command see *Calendar Server 6.3 Administration Reference*.

> **Note:** The **commadmin domain purge** command must be run by the Top-Level Administrator.

For more information about how to remove users and services from a domain, see *Removing Users, Groups, and Services from a Domain*.

Syntax

```
commadmin domain purge -D login -n domain -d domain [-g grace] [-h] [-?] [-i
inputfile] [-p DA server port] [-s] [-S service] [-v] [-V] [-X DA server host
name]
```

Options

Table E–20 displays the options that are mandatory:

Table E–20 commadmin domain purge Mandatory Options

Option	Description
-D *login*	The user ID of the Top-Level Administrator.
-n *domain*	Domain of the Top-Level Administrator.

Table E–20 (Cont.) commadmin domain purge Mandatory Options

Option	Description
-d *domain*	Purge specified domain. The * operator (**-d***) may be used to search for a pattern.

Table E–21 displays the options that are non-mandatory:

Table E–21 commadmin domain purge Non-Mandatory Options

Option	Description
-g *grace*	Delay period (grace period) in days before the domain is purged. Domains marked for deletion for fewer than *grace* days will not be purged. For example, if you use **-g 7**, all entries that have been marked for deletion for 7 days and more are purged, but entries marked for deletion for 6 days and fewer are not purged. A 0 indicates purge immediately. The default value is **5** days. The default value cannot be changed permanently. You can change the grace period only by using the **-g** *grace* option in the **commadmin domain purge** command.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-S *service*	Removes service related object classes and attributes from the domain. If the domain contains users and resources it removes the service specific data from the directory for these users and resources. The list of services is separated by the comma (,) delimiter. The valid *service* values are **mail**, **cal**, and **im**. These values are case-insensitive.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

In the following example, the **example.org** domain is purged and all entries within the domain are also removed:

```
commadmin domain purge -D chris -d example.org -n example.com
```

commadmin domain search

The **commadmin domain search** command obtains all the directory properties associated with domains.

- If no domain is specified with **-d**, all domains will be displayed.

- To obtain all the directory properties for multiple domains, use the **-i** option.
- When **-S** is specified in this command, only the domains having active specified services are displayed.

Syntax

```
commadmin domain search -D login -n domain [-d domain] [-h] [-?] [-i inputfile]
[-p DA server port] [-s] [-S service] [-t Search Template] [-v] [-V] [-X DA server
host name]
```

Options

Table E–22 displays the options that are mandatory:

Table E–22 commadmin domain search Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-n *domain*	The domain of the user specified with the **-D** option.

Table E–23 displays the options that are non-mandatory:

Table E–23 commadmin domain search Non-Mandatory Options

Option	Description
-d *domain*	Search for this domain. If **-d** is not specified or **-d*** is specified, all domains are displayed.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-S *service*	Specifies the services to be searched in the active domains. *service* can have the value of a single service or multiple services. The valid *service* values are **mail**, **cal**, and **im**. These values are case-insensitive. The list of services is separated by the comma (,) delimiter. For Example: `-S mail,cal,im`
-t *Search template*	Specifies the name of the search templates to be used instead of the default search templates. Only active domains are displayed after the search.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

commadmin group create

The **commadmin group create** command adds a single group in the LDAP directory. To create multiple groups, use the **-i** option.

If a group is created without any members, by default, it is a static group.

> **Note:** Groups cannot contain both static and dynamic members.

An email distribution list is one type of group. When a message is sent to the group address, Messaging Server sends the message to all members in the group.

Syntax

```
commadmin group create -D login -G groupname -n domain [-A [+]attributename:value]
[-d domain] [-f ldap-filter] [-h] [-?] [-i inputfile]

[-m internal-member] [-p DA server port] [-s] [-v] [-V] [-X DA server host name]
[-S service [-H mailhost] [-E email] [-M external-member] [-o owner] [-r
moderator]]

[-a true|false ] [-b true|false ] [-c group id] [-j DWPHost] [-q secondary owner]
[-t time zone]
```

Options

Table E–24 displays the options that are mandatory:

Table E–24 commadmin group create Mandatory Options

Option	Description
-D *login*	The user ID of the user who has permission to execute this command.
-n *domain*	The domain of the user specified by the **-D** option.
-G *groupname*	The name of the group (for example, **mktg-list**).

Table E–25 displays the options that are non-mandatory:

Table E–25 commadmin group create Non-Mandatory Options

Option	Description
-A[+] *attributename*:*value*	An attribute to modify. The *attributename* is defined in the LDAP schema and *value* replaces any and all current values for this attribute in the directory. Repeat this option to modify multiple attributes at the same time, or to specify multiple values for the same attribute. A + before the *attributename* indicates adding the value to the current list of attributes.
-d *domain*	The fully qualified domain name of the group (for example, **example.com**). The default is the local domain. If **-d** is not specified, the domain specified by **-n** is used.

Table E–25 (Cont.) commadmin group create Non-Mandatory Options

Option	Description
-f *ldap-filter*	Creates dynamic groups. Setup the LDAP filter by specifying an attribute or a combination of attributes. Multiple **-f** commands can be specified to define many LDAP filters for members of a group. The LDAP filter should define members within the group's organization. Even if the LDAP filter specifies another organization, this value defaults to the group's organization. This constraint prevents members who belong to an outside organization from being added to the group.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-m *internal-member*	User ID or mail: address of the internal members added to this group. To add more than one member, use multiple **-m** options. This option should be used to create static groups.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.
-V	Prints information about the commands and its version.
-S *service*	Specifies the services to be added to the Group. *service* can have the value of a single service or multiple services. The valid service values are mail and cal. These values are case-insensitive. The list of services is separated by the comma (,) delimiter. For Example: `-S mail,cal`

Table E–26 displays the options that are allowed:

Table E–26 commadmin group create Options Allowed if -S mail Specified

Option	Description
-o *owner*	The group owner's email address. An owner is the individual responsible for the distribution list. (This option is also allowed, and is mandatory, when the **-S cal** option is specified.)
-E *email*	The email address of the group. (This option is also allowed when the **-S cal** option is specified.)
-H *mail host*	The mail host to which this group responds (for example, **mailhost.example.com**). The default is the local mail host.

Table E–26 (Cont.) commadmin group create Options Allowed if -S mail Specified

Option	Description
-M *external-member*	Adds an external member to this group. The value of *external-member* is the user email address. To add more than one member, use multiple **-M** options.
-r *moderator*	The moderator's email address.

Table E–27 displays the options that, if the **-S cal** option is specified, are mandatory:

Table E–27 commadmind group create Option Mandatory if -S cal specified

Option	Description
-o *owner*	The group owner's email address. An owner is the individual responsible for the Calendar group's distribution list. The group owner must have Calendar service. (This option is also allowed when the **-S mail** option is specified.)

Table E–28 displays the options that, if the **-S cal** option is specified, are the non-mandatory options that are allowed:

Table E–28 commadmin group create Options Allowed if -S cal specified

Option	Description
-a *true* | *false*	Allows or disallows calendar appointments to be accepted automatically. *true* enables automatic acceptance of appointments. *false* disables automatic acceptance of appointments.
-b *true* | *false*	Allows or disallows calendar appointments to be double-booked, permitting more than one appointment at the same time. *true* enables double-booking of appointments. *false* disables double-booking of appointments.
-c *group id*	Specifies a group ID for the Calendar group. If this option is not specified, Delegated Administrator automatically supplies a group ID.
-E *email*	The email address of the group. This address is used to notify group members of Calendar events. (This option is also allowed when the **-S cal** option is specified.)
-j *DWPHost*	The DNS name of the back-end calendar server which hosts this Calendar group's calendar. This host is the Database Wire Protocol (DWP) server that stores the calendar and its data. If the DNS name of the back-end calendar server is not specified, the value stored in the {{ics.conf}} file of the server is used as the default value.
-q *secondary owner*	The secondary owner's email address. A secondary owner can manage the Calendar group's distribution list. To add more than one secondary owner, use multiple **-q** *secondary owner* options. All secondary owners must have Calendar service.
-t *time zone*	The time zone used to display the Calendar group's calendar in the calendar's user interface. See Calendar Time Zone Strings for a list of the valid time zone strings.

Example

To create a group **testgroup** in the domain **example.com**:

```
commadmin group create -D chris -n example.com -G testgrou -d example.com -m
lorca@example.com -S mail,cal -M achiko@example.com -o achiko@example.com -c
calgroup1
```

commadmin group delete

The **commadmin group delete** command marks a single group as deleted. To mark multiple groups as deleted, use the **-i** option.

To disable a group's usage of services such as Calendar Server or Messaging Server, use the **-S** option. Here **S** is in uppercase.

> **Note:** In order to permanently remove a group, you must run the following command: **commadmin domain purge**. See "commadmin domain purge" for details.

Syntax

```
commadmin group delete -D login -G groupname -n domain [-d domain] [-h] [-?] [-i
inputfile] [-p DA server port] [-s] [-S service] [-v] [-V] [-X DA server host
name]
```

Options

Table E–29 displays the options that are mandatory:

Table E–29 commadmin group delete Mandatory Options

Option	Description
-D *login*	The user ID of the user who has permission to execute this command.
-G *groupname*	The name of the group to be marked as deleted. For example, **mktg-list**.
-n *domain*	The domain of the user specified by the **-D** option.

Table E–30 displays the options that are non-mandatory:

Table E–30 commadmin group delete Non-Mandatory Option

Option	Description
-d *domain*	The domain of the group. If **-d** is not specified, the domain specified by the **-n** option is used.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.

Table E–30 (Cont.) commadmin group delete Non-Mandatory Option

Option	Description
-S *service*	Modifies the value of the specified service status attribute value to deleted. The services listed with the **-S** option are separated by a comma. The valid *service* values are **mail** and **cal**. These values are case-insensitive.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Examples

The following example marks the group **testgroup@example.com** as deleted:

```
commadmin group delete -D chris -n example.com -G testgroup -d example.com
```

The following example marks the mail service for **testgroup@example.com** as deleted:

```
commadmin group delete -D chris -n example.com -G testgroup -d example.com -S mail
```

commadmin group modify

The **commadmin group modify** command changes the attributes of a single group that already exists in the LDAP directory. To change the attributes of multiple groups, use the **-i** option.

A mailing list is one type of group. When a message is sent to the group address, Messaging Server sends the message to all members in the group.

Syntax

```
commadmin group modify -D login -G groupname -n domain [-A
[+|-]attributename:value] [-d domain] [-f [action]ldap-filter] [-h] [-?] [-i
inputfile]

[-m [+|-]internal-member] [-p DA server port] [-s] [-v] [-V] [-X DA server host
name] [-S mail [-o owner] [-E email] [-H mailhost] [-M external-member] [-r
moderator]

[-a true|false ] [-b true|false ] [-c group id] [-j DWPHost] [-q secondary owner]
[-t time zone]
```

Options

Table E–31 displays the options that are mandatory:

Table E–31 commadmin group modify Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-G *groupname*	The name of the group to be modified. For example, **mktg-list**.
-n *domain*	The domain of the user specified by the **-D** option.

Table E–32 displays the following non-mandatory options:

Table E–32 commadmin group modify Non-Mandatory Options

Option	Description
-A [+ | -] *attributename:value*	An attribute to modify. The *attributename* is defined in the LDAP schema and *value* replaces any and all current values for this attribute in the directory. Repeat this option to modify multiple attributes at the same time, or to specify multiple values for the same attribute. A + before the *attributename* indicates adding the value to the current list of attributes. A - indicates removing the value. If the - is used, it must be preceded by two backslashes or enclosed in quotes if the command is specified on the command line. If the option is provided within an input file, one backslash must precede the - sign.
-d *domain*	The domain of the group. If **-d** is not specified, the domain specified by the **-n** option is used.
-f [*action*] *ldap-filter*	Indicates whether a ldap filter is added to or removed from the group A + before the *ldap-filter* indicates that it is to be added to the existing filters. A - indicates removing the existing filter. Type **-f**- to remove all the filters. If the - is used, it must be preceded by two backslashes or enclosed in quotes if the command is specified on the command line. If *action* is not specified, by default the filter is added provided it is not already present. Otherwise an error message is displayed. The LDAP filter should define members within the group's organization. Even if the LDAP filter specifies another organization, this value defaults to the group's organization. This constraint prevents members who belong to an outside organization from being added to the group.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-m [*action*] *internal-member*	Indicates whether to add or remove an internal member. An *action* value of: + adds the member to an existing list of internal members. - removes the member from an existing list of internal members. If the - is used, it must be preceded by two backslashes or enclosed in quotes if the command is specified on the command line. The value of *internal--member* is either a mail address or user ID. **-m*** removes all the internal members.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the local host if no default was configured at install time.

Table E–32 (Cont.) commadmin group modify Non-Mandatory Options

Option	Description
-S *service*	Specifies the services to be added to the group during modification. Before a service is added, Delegated Administrator validates whether the service already exists. If the service exists, an error message is displayed. *service* can have the value of a single service or multiple services. The valid service values are **mail** and **cal**. These values are case-insensitive. The list of services is separated by the comma (,) delimiter. For Example: `-S mail,cal`

Table E–33 displays the options that, if the **-S mail** option is specified, are allowed:

Table E–33 commadmin group modify Options Allowed if -S mail Specified

Option	Description
-o *owner*	The group owner's email address. An owner is the individual responsible for the distribution list. (This option is also allowed, and is mandatory, when the **-S cal** option is specified.)
-E *email*	The email address of the group. (This option is also allowed when the **-S cal** option is specified.)
-H *mail host*	The mail host to which this group responds (for example, **mailhost.example.com**). The default is the local mail host.
-M *external-member*	Adds an external member to this group. The value of *external-member* is the user's email address. To add more than one member, use multiple **-M** options.
-r *moderator*	The moderator's email address.

Table E–34 displays the options that, if the **-S cal** option is specified, are the options that are mandatory:

Table E–34 commadmin group modify Options Mandatory if -S cal Specified

Option	Description
-o *owner*	The group owner's email address. An owner is the individual responsible for the Calendar group's distribution list. The group owner must have Calendar service. (This option is also allowed when the **-S mail** option is specified.)

Table E–35 displays the options that, if the **-S cal** option is specified, are the non-mandatory options that are allowed:

Table E–35 commadmin group modify Options Allowed if -S cal Specified

Option	Description
-a *true*|*false*	Allows or disallows calendar appointments to be accepted automatically. *true* enables automatic acceptance of appointments. *false* disables automatic acceptance of appointments.

Table E–35 (Cont.) commadmin group modify Options Allowed if -S cal Specified

Option	Description
-b *true*|*false*	Allows or disallows calendar appointments to be double-booked, permitting more than one appointment at the same time. *true* enables double-booking of appointments. *false* disables double-booking of appointments.
-c *group id*	Specifies a group ID for the Calendar group. If this option is not specified, Delegated Administrator automatically supplies a group ID.
-E *email*	The email address of the group. This address is used to notify group members of Calendar events. (This option is also allowed when the **-S cal** option is specified.)
-j *DWPHost*	The DNS name of the back-end calendar server which hosts this Calendar group's calendar. This host is the Database Wire Protocol (DWP) server that stores the calendar and its data. If the DNS name of the back-end calendar server is not specified, the value stored in the {{ics.conf}} file of the server is used as the default value.
-q *secondary owner*	The secondary owner's email address. A secondary owner can manage the Calendar group's distribution list. To add more than one secondary owner, use multiple **-q** *secondary owner* options. All secondary owners must have Calendar service.
-t *time zone*	The time zone used to display the Calendar group's calendar in the calendar's user interface. See Calendar Time Zone Strings for a list of the valid time zone strings.

Examples

To remove an internal member (**jsmith**) from the group **testgroup** within the domain **example.com:**

```
commadmin group modify -D chris -d example.com -G testgroup -n example.com -m
jsmith
```

To add Calendar service to the group **testgroup** within the domain **example.com:**

```
commadmin group modify -D chris -d example.com -G testgroup -n example.com -S cal
-o achiko@example.com -c calgroup1
```

commadmin group search

The **commadmin group search** command obtains all the directory properties associated with a single group. To obtain all the directory properties for multiple groups, use the **-i** option.

Syntax

```
commadmin group search -D login -n domain [-d domain] [-E string] [-G string] [-h]
[-?] [-i inputfile] [-p DA server port] [-s] [-S service]

[-t search template] [-v] [-V] [-X DA server host name]
```

Options

Table E–36 displays the options that are mandatory:

Table E–36 commadmin group search Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-n *domain*	The domain of the user specified by the **-D** option.

Table E–37 displays the options that are non-mandatory:

Table E–37 commadmin group search Non-Mandatory Options

Option	Description
-d, *domain*	The domain of the group to be searched. If **-d** is not specified, all domains are searched.
-E *string*	Email address of the group. The wildcard operator (*) may be used within any part of *string*.
-G *string*	The name of the group to be searched. For example, **mktg-list**. If **-G** is not specified, all groups in the domain specified by **-d** are displayed. The wildcard operator (*) may be used within any part of *string*.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where the IS server is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-S *service*	Specifies the service to be searched. The only valid value for *service* is **mail**. This value is case-insensitive. For Example: {{-S mail}}. Only groups with active services are displayed.
-t *Search Template*	Specifies the name of the search templates to be used instead of the default search templates. This is an entry in the directory that defines the filter for the search. Only active groups are searched for.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

To search for a group named **developers** under the **example.com** domain:

```
commadmin group search -D chris -n example.com -G developers -d example.com
```

commadmin resource create

The **commadmin resource create** command creates a directory entry for a resource.

See "Creating a Resource" for instructions on creating a resource.

Syntax

```
commadmin resource create -D login -E email -n domain -u identifier -N name [-c
```

calendar identifier] [-A [+]*attributename*:*value*]

[-C *DWPHost*] [-d *domainname*] [-h] [-?] [-i *inputfile*] [-o *owner*] [-p *DA server port*] [-s] [-T *time zone*] [-u *uid*] [-v] [-V] [-X *DA server host name*]

Options

Table E–38 displays the options that are mandatory:

Table E–38 commadmin resource create Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-E *email*	Specifies the resource's email address.
-n *domain*	Domain of the user specified with the **-D** option.
-u *identifier*	Resource's unique identifier. This *identifier* value should be unique within the domain name space or within all the users and resources the calendar manages in the calendar mode.
-N *name*	Friendly name used to display the resource in the calendar GUI.
-c *calendar identifier*	Identifier for this resource's calendar. The *identifier* value should be unique throughout all the calendars managed by the Calendar Server.

Table E–39 displays the options that are non-mandatory:

Table E–39 commadmin resource create Non-Mandatory Options

Option	Description
-A [+] *attributename*:*value*	An attribute to modify. The *attributename* is defined in the LDAP schema and *value* replaces any and all current values for this attribute in the directory. Repeat this option to modify multiple attributes at the same time, or to specify multiple values for the same attribute. A + before the *attributename* indicates adding the value to the current list of attributes.
-C *DWPHost*(Applies to Calendar Server 6)	The DNS name of the back end calendar server which hosts this user's calendars. If the DNS name of the back-end calendar server is not specified, the value stored in the ics.conf file of the server is used as the default value.
-d *domain name*	Domain of the resource. If **-d** is not specified, the domain specified by **-n** is used.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-o *owner*	Specifies the owner of the resource's calendar (user ID). The resource owner must be a user ID that resides in the domain of the resource.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-T *time zone*	The time zone used to display the resource's calendar in the calendar's user interface. See Calendar Time Zone Strings for a list of the valid time zone strings.
-u *uid*	Specifies the resource's unique ID.

Table E–39 (Cont.) commadmin resource create Non-Mandatory Options

Option	Description
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

To create a resource with name **peter** in the calendar **cal.siroe.com** under the domain **example.com**:

```
commadmin resource create -D chris -n example.com -w bolton -d example.com -u id
-c calid -N peter -C cal.example.com
```

Creating a Resource

A resource consists of two data descriptions: a directory entry and a calendar in the Calendar Server database. The directory entry has an attribute, **icsCalendar**, whose value is the name of the calendar associated with the resource.

You can create a resource with the two data descriptions, using either of the following methods:

- Use **commadmin resource create** to create a directory entry.

  The calendar for the resource is created automatically when the resource is first invited to an event. The **ics.conf** parameter, **resource.invite.autoprovision**, determines whether a resource's calendar is created automatically when the resource is invited to an event. By default, the value of this parameter is set to **Yes**. To create the resource's calendar before any invitations are sent to the resource, use the **cscal** command.

  Example:

  Use **commadmin resource create** to create a directory entry:

  ```
  commadmin resource create -D amadmin -n blink.example.com -X blink -p 5555 -d
  example.com -u resourceOne -N firstResource -c resourceOneCalendar
  ```

  The directory entry is as follows:

  ```
  dn: uid=resourceONE,ou=People,o=example,o=domainroot
  uid: resourceONE
  objectClass: icsCalendarResource
  objectClass: top
  cn: firstResource
  icsStatus: active
  icsCalendar: resourceOne
  ```

- Use the **csresource** command by itself. The **csresource** command creates a directory entry and a calendar.

  However, using **csresource** to create both the directory entry and the calendar is only recommended if the directory is in a Schema 1 environment and you are not using Access Manager.

You can now log in as any user and invite the resource to an event.

commadmin resource delete

The **commadmin resource delete** command marks the resource as deleted.

> **Note:** To permanently remove the resource, run the "commadmin domain purge" command.

Syntax

```
commadmin resource delete -D login -u identifier -n domain [-d domainname] [-h]
[-?] [-i inputfile] [-p DA server port] [-s] [-v] [-V] [-X DA server host name]
```

Options

Table E–40 displays the options that are mandatory:

Table E–40 commadmin resource delete Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-n *domain*	Domain of the user specified with the **-D** option.
-u *identifier*	Resource's unique identifier.

Table E–41 displays the options that are non-mandatory:

Table E–41 commadmin resource delete Non-Mandatory Options

Option	Description
-d *domainname*	Domain of the resource. If **-d** is not specified, the domain specified by **-n** is used.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specify the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

To mark a resource as deleted:

```
commadmin resource delete -D chris -n example.com -u bill023
```

commadmin resource modify

The **commadmin resource modify** command modifies the resource.

Syntax

```
commadmin resource modify -D login -n domain -u identifier [-A
[+|-]attributename:value] [-d domainname ] [-h] [-?] [-i inputfile] [-N name] [-p
DA server port] [-s] [-T time zone] [-v] [-V] [-X DA server host name]
```

Options

Table E–42 displays the options that are mandatory:

Table E–42 commadmin resource modify Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-n *domain*	Domain of the user specified with the **-D** option.
-u *identifier*	Resource's unique identifier.

Table E–43 displays the options that are non-mandatory:

Table E–43 commadmin resource modify Non-Mandatory Options

Option	Description
-A [+ *or* -] *attributename:value*	An attribute to modify. The *attributename* is defined in the LDAP schema and *value* replaces any and all current values for this attribute in the directory. Repeat this option to modify multiple attributes at the same time, or to specify multiple values for the same attribute. A + before the *attributename* indicates adding the value to the current list of attributes. A - indicates removing the value. If a - is used, it must be preceded by two backslashes if the command is specified on the command line. If the option is provided within an input file, one backslash must precede the - sign.
-d *domainname*	Domain of the resource. If **-d** is not specified, the domain specified by **-n** is used.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-N *name*	Common name used to display the resource in the calendar user interface.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-T *time zone*	The time zone used to display resource's calendar in the calendar GUI. See Calendar Time Zone Strings for a list of the valid time zone strings.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

To modify a resource with the unique identifier **bill023** with a new common name **bjones**:

```
commadmin resource modify -D chris -n example.com -d test.com -u bill023 -N bjones
```

commadmin resource search

The **commadmin resource search** command searches for a resource.

Syntax

```
commadmin resource search -D login -n domain [-d domain] [-h] [-?] [-i inputfile]
[-N string] [-p DA server port] [-s] [-t Search Template]

[-u string] [-V] [-v] [-X DA server host name]
```

Options

Table E–44 displays the options that are mandatory:

Table E–44 commadmin resource search Mandatory Options

Option	Description
-D *login*	The user ID of the user with the permission to execute this command.
-n *domain*	Domain of the user specified with the **-D** option.

Table E–45 displays the options that are non-mandatory:

Table E–45 commadmin resource search Non-Mandatory Options

Option	Description
-d *domain*	Domain of the resource. Search is performed only in the domain. If **-d** is not specified or **-d*** is specified, then all domains are searched.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-N *string*	Enter the resource's common name. The wildcard operator (*) may be used within any part of *string*.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-t *Search Template*	Specifies the name of the search templates to be used instead of the default search templates. This is an entry in the directory that defines the filter for the search. Only active resources are searched for.
-u *string*	The resource identifier specified must be unique for the domain name space or for all the users and resources the calendar manages. The wildcard operator (*) may be used within any part of *string*. If the identifier is not specified or **-l*** is specified all resources are displayed during the search.
-v	Enables debugging output.
-V	Prints information about the command and its version.

Table E–45 (Cont.) commadmin resource search Non-Mandatory Options

Option	Description
-X *DA server host name*	Specify the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

To search for a resource **arabella** in the domain **example.com**:

```
commadmin resource search -D serviceadmin -n example.com -d example.com -u
arabella
```

commadmin user create

The **commadmin user create** command creates a single user in the LDAP directory. To create multiple users, use the **-i** option.

> **Note:** Starting with Delegated Administrator for Oracle Communications Unified Communications Suite, the **-S** and **-A** (inetcos) options should not be used together for service modifications as the two provisioning models compete.

Syntax

```
commadmin user create -D login -F firstname -n domain -L lastname -l userid [-A
[+]attributename:value] [-d domain] [-I initial]

[-h] [-?] [-i inputfile] [-p DA server port] [-s] [-v] [-V] [-X DA server host
name] [-S mail [-E email] [-H mailhost]] [-S cal [-B DWPHost]

[-E email] [-k calid_type] [-J First Day of Week] [-T time zone]] [-S im]
```

Options

Table E–46 displays the options that are mandatory:

Table E–46 commadmin user create Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-F *first_name*	The user's first name; must be a single word without any spaces.
-n *domain*	The domain of the user specified with the **-D** option.
-l *user_id*	The user's login name. Values entered with this option are limited to printable ASCII characters. For mail users, the following additional restrictions apply: ■ These characters are not allowed: % * ? & / : \ ■ You cannot enter a **-** (dash) as the leading character. The - is reserved to indicate negative rights. That is, the IMAP ACL extension reserves a leading **-** to deny permissions to the access rights that follow it in the ACL. ■ You cannot enter **group=** as the leading term. It is reserved for group IDs ■ These words are reserved and are not allowed: **anonymous**, **anybody**, **anyone**, **anyone@***domain*

Table E–46 (Cont.) commadmin user create Mandatory Options

Option	Description
-L *last_name*	The user's last name.

Table E–47 displays the options that are non-mandatory:

Table E–47 commadmin user create Non-Mandatory Options

Option	Description
-A[+] *attributename:value*	An attribute to modify. The *attributename* is defined in the LDAP schema and *value* replaces any and all current values for this attribute in the directory. Repeat this option to modify multiple attributes at the same time, or to specify multiple values for the same attribute. A + before the *attributename* indicates adding the value to the current list of attributes.
-d *domain*	Domain of the user. If **-d** is not specified, the domain specified by **-n** is used.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-I *initial*	User's middle initial.
-h, **-?**	Prints command usage syntax.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the directory.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.
-S *service*	Adds the specified service to the user during creation. *service* can have the value of a single service or multiple services. The valid *service* values are **mail**, **cal**, and **im**. These values are case-insensitive. The list of services is separated by the comma (,) delimiter. For Example: `-S mail,cal,im`

Table E–48 displays the options that, if the **-S mail** option is specified, are the non-mandatory options that are allowed:

Table E–48 commadmin user create Options Allowed if -S mail Specified

Option	Description
-E *email*	The email address of the user.
-H *mail host*	The mail host of the user.

Table E–49 displays the options that, if the **-S cal** option is specified, are the non-mandatory options that are allowed:

Table E–49 commadmin user create Options Allowed if -S cal Specified

Option	Description
-B *DWPHost*	DNS name of the back-end calendar that hosts the user's calendar.
-E *email*	The email address of the calendar user.
-J *First Day of Week*	First day of the week shown when the calendar is displayed in the calendar server user interface. The valid values are 0-6 (0 is Sunday, 1 is Monday, and so on).
-k *calid_type*	Specifies the type of calendar id that is created. The accepted values are **legacy** and **hosted**. If **-k legacy** is specified, only the calendar id is used (for example, **jsmith**). If **-k hosted** is specified, the calendar id plus domain is used (for example, **jsmith@example.com**). If the **-k** option is not specified, the default is to use the calendar id plus domain (**hosted**). You can set the value of the calendar id type that is created if the **-k** option is not specified. To do so, add the following parameter to the **resource.properties** file: `switch-caltype=`*`value`* where *value* is **hosted** or **legacy**. The **resource.properties** file is located in the following directory: *`DelegatedAdmin_home`*`/data/WEB-INF/classes/sun/comm/cli/server/servlet/resource.properties`
-T *time zone*	The time zone in which the user's calendar is displayed. See Calendar Time Zone Strings for a list of the valid time zone strings.

Example

To create a new user, **smith**, enter:

```
commadmin user create -D chris -n example.com -F smith -l john -L major -S mail -H
mailhost.example.com
```

commadmin user delete

The **commadmin user delete** command marks a single user as deleted. To mark multiple users as deleted, use the **-i** option.

This command only marks a user as deleted; it does not remove the user entry from the directory.

No undelete command exists. However, you can use the **ldapmodify** command to change the status attribute of a user entry to **active** at any time before the purge grace period has expired and a purge is set to run against the entry.

To remove a user

The following steps summarize how to remove a user from the directory. For more information, see Removing Users, Groups, and Services from a Domain.

1. Mark the user as deleted by running the **commadmin user delete** command.
2. Remove resources from the user.

A resource can be a mailbox or a calendar.

For mail services, the command is called **msuserpurge**.

For calendar services, the program is **csclean**. Refer to *Sun Java System Calendar Server System Administrator's Guide* for information about the **csclean** command.

3. Permanently remove the user by invoking the **commadmin domain purge** command.

   See "commadmin domain purge" for reference details.

Syntax

```
commadmin user delete -D login -n domain -l login name [-d domain] [-h] [-?] [-i
inputfile] [-p DA server port] [-s] [-S service] [-v] [-V]

[-X DA server host name]
```

Options

Table E–50 displays the options that are mandatory:

Table E–50 commadmin user delete Mandatory Options

Option	Description
-D *login*	The user ID of the user with the permission to execute this command.
-n *domain*	The domain of the user specified with the **-D** option.
-l *userid*	The user ID of the user to be deleted.

Table E–51 displays the options that are non-mandatory:

Table E–51 commadmin user delete Non-Mandatory Options

Option	Description
-d *domain*	Domain of the user. If **-d** is not specified, the domain specified by **-n** is used.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the directory.
-S *service*	Specifies the services to be removed from the user. The user remains active, but only the specified services are deactivated. If **-S** is not specified, then the user is deleted. *service* can have the value of a single service or multiple services. The valid *service* values are **mail**, **cal**, and **im**. These values are case-insensitive. The list of services is separated by the comma (,) delimiter. For example: `-S mail,cal,im`
-v	Enables debugging output.
-V	Prints information about the command and its version.

Table E–51 (Cont.) commadmin user delete Non-Mandatory Options

Option	Description
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

To mark an existing user as deleted:

```
commadmin user delete -D chris -n example.com -l smith
```

To delete the mail services only from user **smith**:

```
commadmin user delete -D chris -n example.com -l smith -S mail
```

commadmin user modify

The **commadmin user modify** command modifies attributes of a single user's directory entry. To modify multiple users, use the **-i** option.

Syntax

```
commadmin user modify -D login -n domain -l userid [-A [+|-]attributename:value]
[-d domain] [-h] [-?] [-i inputfile] [-p DA server port]

[-s] [-v] [-V] [-X DA server host name] [-S mail -H mailhost [-E email]] [-S cal
[-B DWPHost] [-E email] [-k calid_type] [-J First Day of Week]

[-T time zone]] [-S im]
```

Options

Table E–52 displays the options that are mandatory:

Table E–52 commadmin user modify Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-n *domain*	Domain of the user specified with the **-D** option.
-l *userid*	User's login ID.

Table E–53 displays the options that are non-mandatory:

Table E–53 commadmin user modify Non-Mandatory Options

Option	Description
-A [+ *or* -] *attributename*:*value*	An attribute to modify. The *attributename* is defined in the LDAP schema and *value* replaces any and all current values for this attribute in the directory. You can repeat this option to modify multiple attributes at the same time, or to specify multiple values for the same attribute. A + before the *attributename* indicates adding the value to the current list of attributes. A - indicates removing the value. If the - is used, it must be preceded by two backslashes if the command is specified on the command line. If the option is provided within an input file, one backslash must precede the - sign.

Table E–53 (Cont.) commadmin user modify Non-Mandatory Options

Option	Description
-d *domain*	Domain of the user or group. If **-d** is not specified, the domain specified by **-n** is used.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-p *DA server port*	Specifies an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the Delegated Administrator Server or to Access Manager.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.
-S *service*	Adds the specified services to the user after validating whether the user has the service specified with **-S** option. If the user already has the service an error message is displayed. *services* can have the value of a single service or multiple services. The valid *service* values are **mail**, **cal**, and **im**. These values are case-insensitive. The list of services is separated by the comma (,) delimiter. For example: `-S mail,cal,im`

Table E–54 displays the options that, if the **-S mail** option is specified, are the non-mandatory options that are allowed:

Table E–54 commadmin user modify Options Allowed if -S mail Specified

Option	Description
-E *email*	Specifies the email address of the user.
-H *mail host*	The mail host of the user. This option is mandatory if the **-S mail** option is specified.

Table E–55 displays the options that, if the **-S cal** option is specified, are the non-mandatory options that are allowed:

Table E–55 commadmin user modify Options Allowed if -S cal Specified

Option	Description
-B *DWPHost*	Specifies the DNS name of the back-end calendar server that hosts this user's calendars. Note: This attribute can only be added and cannot be modified if it already exists.
-E *email*	Specifies the email address for the calendar user.

Table E–55 (Cont.) commadmin user modify Options Allowed if -S cal Specified

Option	Description
-J *First Day of Week*	The first day of the week shown when the calendar is displayed in the calendar server user interface. The valid values are 0-6 (0 is Sunday, 1 is Monday, and so on).

-k *calid_type*	Specifies the type of calendar id that is created (when adding the calendar service). The accepted values are **legacy** and **hosted**. If **-k legacy** is specified, only the calendar id is used (for example, **jsmith**).
	If **-k hosted** is specified, the calendar id plus domain is used (for example, **jsmith@example.com**).
	If the **-k** option is not specified, the default is to use the calendar id plus domain (**hosted**).
	You can set the value of the calendar id type that is created if the **-k** option is not specified. To do so, add the following parameter to the **resource.properties** file:
	`switch-caltype=value`
	where *value* is **hosted** or **legacy**.
	The **resource.properties** file is located in the following directory:
	`DelegatedAdmin_home/data/WEB-INF/classes/sun/comm/cli/server/servlet/resource.properties`

-T *time zone*	The time zone in which the user's calendar is displayed. See Calendar Time Zone Strings for a list of the valid time zone strings.

Examples

The following example adds a mail service for the user **smith**:

```
commadmin user modify -D chris -n example.com -l smith -A description:"new
description" -S mail -H mail host.siroe.com
```

In this example, a mail forwarding address is added for user **smith**:

```
commadmin user modify -D chris -n example.com -l smith -A
+mailforwardingaddress:tsmith@siroe.com
```

commadmin user search

The **commadmin user search** command displays all provisioned directory properties associated with a single user. To obtain all the directory properties for multiple users, use the **-i** option. Only active users are displayed after a search.

Syntax

```
commadmin user search -D login -n domain [-d domain] [-E string] [-F string] [-h]
[-?] [-i inputfile] [-L string] [-l string]

[-p DA server port] [-s] [-S service] [-t Search Template] [-v] [-V] [-X DA server
host name]
```

Options

Table E–56 displays the options that are mandatory:

Table E–56 commadmin user search Mandatory Options

Option	Description
-D *login*	The user ID of the user with permission to execute this command.
-n *domain*	The domain of the user specified with the **-D** option.

Table E–57 displays the options that are non-mandatory:

Table E–57 commadmin user search Non-Mandatory Options

Option	Description
-d *domain*	The domain of the user. The user is searched only in the specified domain. If **-d** is not specified, all domains are considered for the search.
-E *string*	Searches for user's mail address. The wildcard operator (*) may be used within any part of *string*.
-F *string*	Searches for user's first name. The wildcard operator (*) may be used within any part of *string*.
-h, **-?**	Prints command usage syntax.
-i *inputfile*	Reads the command information from a file instead of the command line. An option value specified in the input file overrides any value for the same option set in the command line.
-L *string*	Searches for user's last name. The wildcard operator (*) may be used within any part of *string*.
-l *string*	Searches for user's login name. The wildcard operator (*) may be used within any part of *string*.
-p *DA server port*	Use this option to specify an alternate TCP port where Delegated Administrator is listening. If not specified, the default *DA server port* is used, or Port 80 is used if no default was configured at install time.
-s	Use SSL (Secure Socket Layer) to connect to the directory.
-S *service*	Specifies the services to match in the user search. *services* can have the value of a single service or multiple services. The valid *service* values are **mail**, **cal**, **im**, and **contacts**. These values are case-insensitive. The list of services is separated by the comma (,) delimiter. For example: `-S mail,cal,im,contacts`
-t *Search template*	Specifies the name of the search templates to be used instead of the default search templates. This is an entry in the directory that defines the filter for the search. Only active users are searched for.
-v	Enables debugging output.
-V	Prints information about the command and its version.
-X *DA server host name*	Specifies the host on which Delegated Administrator is running. If not specified, the default *DA server host name* is used, or the localhost if no default was configured at install time.

Example

The following example searches for users in the **example.com** domain:

```
commadmin user search -D chris -d example.com -n example.com
```

Permission to Run Commands

Table E–58 shows who has permission to run the various **commadmin** commands.

Table E–58 Permission to Run commadmin Commands

Command	Description	Permission to Run*
commadmin admin add	Grants Organization Administrator privileges to a user	Top-Level Administrator
commadmin admin remove	Revokes Organization Administrator privileges from a user	Top-Level Administrator
commadmin search	Searches and displays users who have Organization Administrator privileges	Top-Level Administrator Organization Administrator
commadmin debug log	Creates a debug log	Top-Level Administrator
commadmin domain create	Creates a domain	Top-Level Administrator
commadmin domain delete	Deletes a domain	Top-Level Administrator
commadmin domain modify	Modifies a domain	Top-Level Administrator
commadmin domain purge	Purges a domain	Top-Level Administrator
commadmin domain search	Searches for a domain	Top-Level Administrator
commadmin group create	Creates a group	Top-Level Administrator Organization Administrator
commadmin group delete	Deletes a group	Top-Level Administrator Organization Administrator
commadmin group modify	Modifies a group	Top-Level Administrator Organization Administrator
commadmin group search	Searches for a group	Anyone
commadmin resource create	Creates a resource	Top-Level Administrator Organization Administrator
commadmin resource modify	Modifies a resource	Top-Level Administrator Organization Administrator
commadmin resource delete	Deletes a resource	Top-Level Administrator Organization Administrator
commadmin resource search	Searches for a resource	Anyone
commadmin user create	Creates a user	Top-Level Administrator Organization Administrator
commadmin user delete	Deletes a user	Top-Level Administrator Organization Administrator
commadmin user search	Searches for a user	Anyone
commadmin user modify	Modifies a user	Top-Level Administrator Organization Administrator

Note: Delegated Administrator does not support the Service Provide Administrator's use of the **commadmin** command.