

# Sun Desktop Manager 1.0 Administration Guide



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

Part No: 819-2726  
January, 2006

Copyright 2006 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

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

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

Sun, Sun Microsystems, the Sun logo, docs.sun.com, AnswerBook, AnswerBook2, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

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

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

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

---

Copyright 2006 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

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

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

Sun, Sun Microsystems, le logo Sun, docs.sun.com, AnswerBook, AnswerBook2, et Solaris sont des marques de fabrique ou des marques déposées, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

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

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

# Contents

---

<b>Preface</b> .....	7
<b>1 Concepts and Architecture</b> .....	9
Scope of the Desktop Manager .....	9
Architecture .....	10
Configuration Repositories .....	10
Available Configuration Repositories .....	12
Management Tools .....	13
Templates .....	13
Configuration Agent .....	14
Configuration Adapters .....	14
From Configuration Profiles to Application Settings .....	14
Configuration Data Sources .....	14
Construction of the Profile Configuration Data .....	14
Configuration Data Source Combination .....	16
<b>2 Using the Desktop Manager GUI</b> .....	17
Introduction .....	17
Configuration Repositories Window .....	17
Configuration Repositories Table .....	18
▼ Creating a New Configuration Repository .....	19
▼ Removing a Configuration Repository .....	20
▼ Renaming a Configuration Repository .....	20
▼ Synchronizing Configuration Repositories .....	20
Profiles — Tree View .....	21
Navigation Area .....	21
▼ Searching for an Element .....	22
▼ Using the Find Function .....	23

Content Area .....	23
▼ Creating a new Profile .....	24
▼ Deleting a Profile .....	24
▼ Assigning a Profile .....	25
▼ Unassigning a Profile .....	25
▼ Renaming a Profile .....	25
▼ Copying and Moving a Profile .....	25
▼ Importing Profiles .....	26
▼ Exporting Profiles .....	26
▼ Viewing Effective Settings .....	26
Profiles — All .....	27
▼ Creating a new Profile .....	27
▼ Deleting a Profile .....	28
▼ Renaming a Profile .....	28
▼ Copying and Moving a Profile .....	28
▼ Importing Profiles .....	28
▼ Exporting Profiles .....	29
Profile Editor .....	29
General Properties .....	29
▼ Specifying General Properties for a Profile .....	30
Configuration Settings .....	30
▼ Viewing and Editing a Profile's Configuration Settings .....	30
Assigned Elements .....	31
▼ Assigning an Element to a Profile .....	31
▼ Unassigning an Element from a Profile .....	32
Settings Summary .....	32
Advanced Options .....	32
▼ Changing the Application from Location .....	33
▼ Assigning a Merge Order to a Profile .....	33
<b>3 Using the Command Line Interface .....</b>	<b>35</b>
Overview of the Desktop Manager CLI .....	35
Working with the CLI .....	35
Invoking CLI Commands .....	35
Bootstrapping Information Required by the CLI .....	36
Bootstrapping file .....	36

Bootstrapping options .....	36
Authentication by Username and Password .....	36
Authentication by login command .....	36
Authentication for the other commands .....	37
Running a Command .....	37
Representing Elements .....	37
CLI Commands .....	38
Help .....	38
Version Information .....	38
Add .....	38
Create .....	39
Delete .....	39
Export .....	40
Import .....	40
List .....	41
Login .....	42
Modify .....	43
Remove .....	43
Rename .....	44
Command Summary .....	44
<b>4 Migration .....</b>	<b>47</b>
Object Model and Terminology .....	47
Profile Management .....	47
Profile Format .....	48
Delegated Administration .....	48



# Preface

---

The *Sun Desktop Manager 1.0 Administration Guide* provides information about the concepts and usage of the Sun™ Desktop Manager 1.0. It contains a detailed description of the Graphical User Interface and its functionality, as well as a description of the Command Line Interface.

## How This Book Is Organized

[Chapter 1](#) provides an overview of the Sun Desktop Manager.

[Chapter 2](#) provides information about how to use the Sun Desktop Manager GUI.

[Chapter 3](#) describes the commands used in the Sun Desktop Manager CLI.

[Chapter 4](#) describes how to migrate from the Java Desktop System Configuration Manager 1.1 to the Sun Desktop Manager 1.0.

## Related Books

The following books provide additional information about the Desktop Manager:

- *Sun Desktop Manager 1.0 Installation Guide*
- *Sun Desktop Manager 1.0 Developer Guide*

## Accessing Sun Documentation Online

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



# Concepts and Architecture

---

The Sun Desktop Manager provides a framework to store configuration settings for applications on a network in a central location for users, organizations, and host machines that run the application.

This chapter describes the general architecture and the key concepts of the Desktop Manager.

## Scope of the Desktop Manager

The Desktop Manager directly supports the following configuration settings:

- Gconf (Gnome configuration framework)
- StarOffice™ Registry
- Mozilla Preferences
- Java Preferences
- Desktop launchers
- Menu entries
- Startup applications

---

**Note** – The Desktop Manager only supports applications that use these settings.

---

By default, only the settings that are relevant to a system administrator can be configured with the Desktop Manager. However, you can use templates that are included with the installation to extend the functionality of the Desktop Manager to include the configuration settings that you want to control. Furthermore, desktop applications that do not use the supported configuration systems can access central configuration data through the legacy data framework.

# Architecture

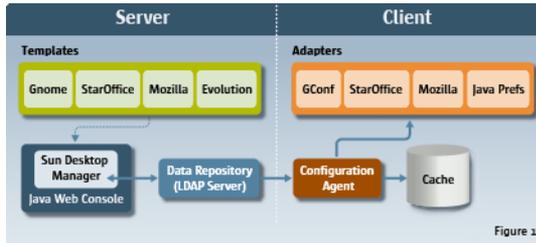


FIGURE 1-1 High-level architecture

The Desktop Manager contains the following components:

- **Configuration Repositories:** Configuration repositories store the configuration profiles and organizational structures.
- **Management Tools:** Management tools refers to both the Desktop Manager web-based administration GUI and the command line interface (CLI). With these tools, you can create, modify, delete, and assign configuration profiles.

The management tools access the data in a configuration repository and use templates to display the data in a browser window.

- **Configuration Agent and adapters:** The agent retrieves and caches the configuration settings on the configuration repositories for the user applications. The adapters apply the settings. The agent and the adapters must be installed on every client.
- **Templates:** Templates render the configuration data in a web browser window.

## Configuration Repositories

The Desktop Manager stores configuration data in a configuration repository. A configuration repository stores the following three types of configuration data:

- **Organization structure:** describes the structure of an organization. A single object of an organization structure is called an *element*. The configuration data for an organization structure provides the following information:
  - A tree that represents the Organization/Sub-Organization's structure. This includes a list of users that are part of the organization, as well as the location of the users in the organization structure.
  - A tree that represents the host Domain/Sub-Domain structure.
- **Configuration profiles:** define sets of configuration data, called *configuration profiles*, for applications or modules. Profiles can be assigned to organizations, domains, hosts, and users. A profile provides a default value for a configuration setting or enforces a value for a configuration key. A profile can also contain profiles that cover multiple applications.

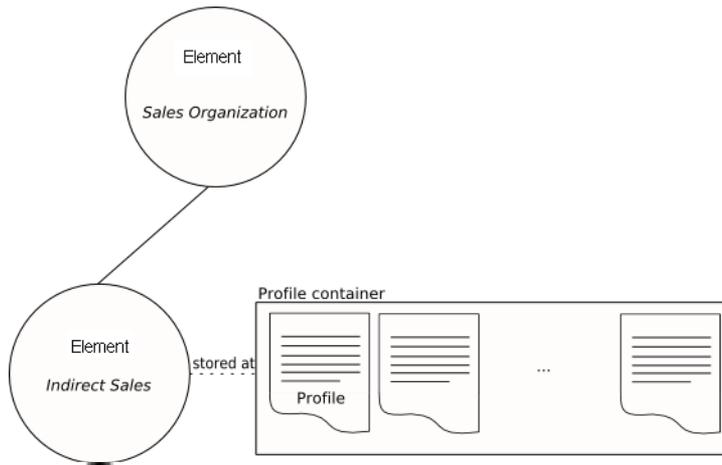
You can use configuration profiles to define default configuration settings for an organization, such as the default unit of measure for rulers in StarOffice Writer. However, a user can set preferences manually in an application to override such defaults.

---

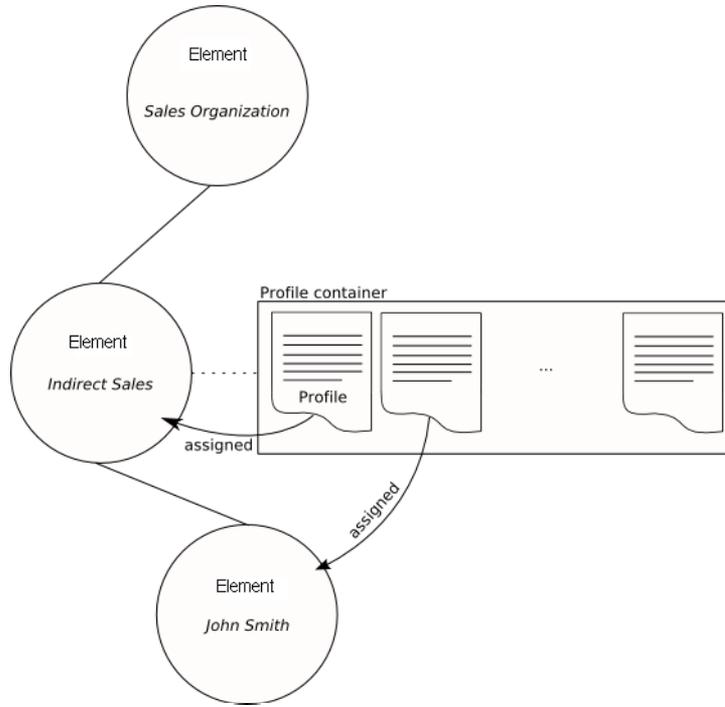
**Note** – You can change the value of a configuration setting to enforce a default setting in a profile, so that a user cannot change the setting manually.

---

Configuration profiles are stored at an element node in an organization or domain structure.



The assignment of profiles associates the configuration data that is contained in a profile to an element. You can only assign a profile to the element that stores the profile, or to elements that occur below the storage element in the hierarchy.



The Desktop Manager also stores a priority with configuration profiles. Priorities determine the order that the profiles are assigned when you create the Profile Configuration Data (see [“Construction of the Profile Configuration Data” on page 14](#)). You cannot store more than one configuration profile with the same priority in an element.

You can also use the Desktop Manager to store local, user-specific configuration profiles directly on desktop machines.

- **Assignments:** Assignments define the relationship between one or many elements of an organization and a profile. Assignments define the elements of the organization or domain that the configuration data can be applied to.

Child elements inherit the assignments of the parent element in the organization hierarchy.

## Available Configuration Repositories

There are three types of configuration repositories that can be implemented:

- **LDAP:** stores the configuration data on an LDAP directory server as additional entries. Desktop Manager supports the following LDAP directory servers:
  - Sun Java Systems Directory Server
  - OpenLDAP
  - Microsoft Active Directory

The access protocol to query this type of repository is LDAP. However, any other LDAPv3 compliant directory can be used as a repository.

- **File:** stores the configuration data in a filesystem. The Desktop Manager accesses this type of repository directly from the filesystem or through HTTP/HTTPS. For HTTP/HTTPS access, you need to configure a web server so that the Agent can access the configuration repository; the management tools require read/write access to the file system that stores the profiles and assignments.
- **Hybrid:** a hybrid repository reads the organizational structure of the company from an LDAP server, and then read/writes the configuration settings into a file system.

---

**Note** – The LDAP Configuration Repository provides the best overall performance. The hybrid repository is best for when you do not have write access to the LDAP directory. The file-based repository is only useful for evaluation purposes.

---

## Management Tools

The management tools provide a web-based graphical user interface and a command-line interface where you can manage the configuration data. The tools only operate on the configuration repository and do not require the agents to run.

If you use an LDAP configuration repository, you can deploy the management tools in a separate system from the one that holds the LDAP service. If you use the file-based repository, the management tools require direct access as well as read/write permissions to the repository for the noaccess user, or the user under which the Java Web Console is executed. That is, the tools must be in the same system as the repository, or the repository must be an NFS mount with read/write access for the tools. The noaccess user runs the Desktop Manager GUI, and must be created when you install the tools.

You can use the management tools to create, delete, modify, assign, and unassign profiles. You cannot use the tools to add, delete, and modify elements in the hierarchy, for example, to add users.

## Templates

Desktop Manager uses templates to view, define, and enforce configuration settings in the configuration repository and to render the GUI for displaying these configuration settings. The templates are deployed by the web-based Management tools.

For more information on templates, see the *Sun Desktop Manager 1.0 Developer Guide*.

## Configuration Agent

To access the configuration data from the Desktop Manager, a desktop client requires the Desktop Manager Configuration Agent. The Configuration Agent communicates with the remote configuration data repository and the adapters, as well as integrates data into specific configuration systems. The configuration systems that are currently supported are GConf, Java Preferences, Mozilla Preferences, and StarOffice Registry.

## Configuration Adapters

Configuration adapters query the configuration agent for configuration data and provide the data to the applications. The adapters must be installed on every client that you want to manage centrally.

# From Configuration Profiles to Application Settings

This section describes how the configuration data is processed to end up with user settings for a specific application running in a specific host.

## Configuration Data Sources

Each user application receives configuration data from the following sources:

- Default configuration data source: stores the default settings for an application. This configuration data source is deployed with the application and remains mostly unchanged throughout its lifetime. The query mechanisms and the format for this data source are defined by each application. Some applications use a "proprietary" method to store configuration data, while other applications use a shared subsystem for the configuration data, for example, GConf.
- User configuration data source: stores user settings for an application. The query/update mechanisms and format for this data source are defined by each application.
- Profile Configuration data source: this data source is provided by the Sun Desktop Manager 1.0. The configuration data is stored in the configuration repositories. The mechanisms to access that data are provided by the Configuration Agent and the Configuration Adapters.

The application settings for a user on a host are calculated in two steps. The profile configuration tree is constructed, and then the configuration data sources are combined.

## Construction of the Profile Configuration Data

The profile configuration data holds the configuration profile for a user application that runs on a specific host.

The organizational units of an organization, along with the users, are stored in the configuration repository hierarchically. The same applies to the domain components.

Configuration profiles are assigned to elements in the hierarchies. Configuration profiles that are assigned to an element are inherited by the children of that element.

The configuration data of an application depends on the user who runs the application and the host where the application runs.

The configuration settings that affect a user depend on the configuration profiles that are assigned to the elements in the path from the user element to the root of the tree. These profiles must be merged together to build the set of configuration settings for the user.

Since it is possible to define profiles based on the host where the application of the user is running, the profiles assigned to the host, or to any of the elements that are in the path from the host to the root of the tree should also be merged together with the configuration profiles that affects the user.

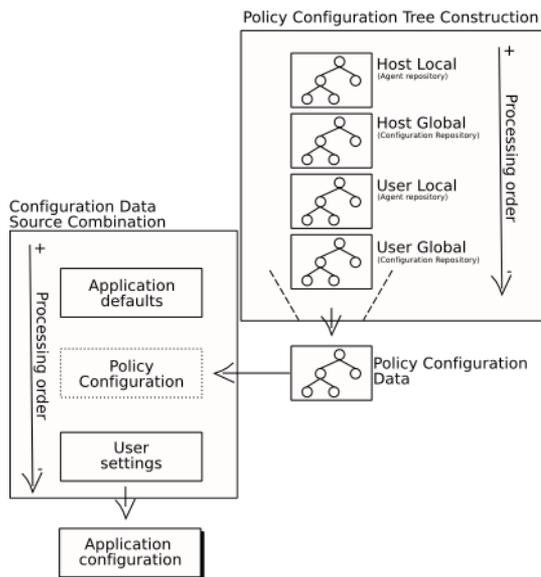


FIGURE 1-2 Configuration Process

The following rules are used to construct the profile configuration:

- **Order of processing:** The order in which profiles are merged is important. The Host Local set of configuration profiles are processed first, followed by the Host Global set, the User Local set, and then the User Global set.

If there is more than one configuration profile in a set, the priority number that is associated with a profile determines the processing order for the profile, that is, the lowest number is processed first.

- **Properties and Values:** A property is an element in the configuration that can be defined, for example, "background color". A value is the value that you assign to a property, for example, the value for the property "background color" is "white".

If a new property is introduced by one profile, the property and its value are added to the merged configuration tree.

If a property has a value that is marked as enforced, the new value is stored in the merged tree. New values for the property that occur in other profiles are ignored.

If a property already has a value when a new value is defined in a profile that is being processed, the new value is assigned to the property.

## Configuration Data Source Combination

The configuration data provided by the three different configuration data sources must be combined to produce a single set of settings for the user application to use at runtime.

1. The configuration data provided by the default configuration provider is read and a configuration tree is constructed.
2. A profile configuration data is constructed based on the user and host of the client application.
3. The user settings are read and a configuration tree is constructed.
4. The three trees are combined into one to get the configuration settings that the application will use. The rules followed in this process are the same that were used to construct the profile configuration data.

The resulting tree will be used by the application adapters to provide the configuration settings.

# Using the Desktop Manager GUI

---

This chapter provides information about the Desktop Manager functionality and usage.

## Introduction

The following are some terms you should know in order to work with the Desktop Manager:

- *Configuration Repository*: The organization and domain tree container, together with the profiles stored in it.
- *Domain*: An element in the domain tree. Represents a realm in a computer network. May contain domains and hosts.
- *Element*: A logical object to which configuration data can be assigned. Users, roles/groups, and organizations are examples of entities known to the Desktop Manager
- *LDAP*: Lightweight Directory Access Protocol. LDAP is a directory service protocol that runs over TCP/IP. The details of LDAP are defined in RFC 1777 'The Lightweight Directory Access Protocol'.
- *Organization*: An element in the organization tree. Represents an organization or a sub-organization, for example, "Call Center". May contain organizations, users, or roles.
- *Profile*: A named container for configuration settings. It is stored in a configuration repository and can be assigned to elements.

## Configuration Repositories Window

The configuration repository is the storage location for the configuration data that you need to configure your applications and where organizational hierarchy information is stored, along with profile and assignment data for each element. In the Configuration Repositories window, you can create new repositories, as well as work with and synchronize existing repositories.

The Configuration Repositories Window consists of a masthead and the Configuration Repositories list.

The masthead provides a number of general links. The upper part of the masthead contains the Utility Bar, which contains four links (from left to right):

- The Console button opens the Java Web Console launch page.
- The Version button opens a window that displays version information about the Desktop Manager.
- The Log Out button logs you out of the Java Web Console, and thus the Configuration Manager, returning you to the Login page.
- The Help button opens the online help pages.

The lower section of the masthead contains the product name, the server name, and the name of the administrator currently logged in.

## Configuration Repositories Table

The Configuration Repositories table consists of five columns:

- The Selection column, which contains check boxes to select repositories  
By selecting one or more repositories, the Remove, Rename, and Synchronize functions are enabled.

---

**Note** – Selecting a repository disables the New button.

---

- Name: contains the name of the repositories  
The contents of the Name column are links. If you click on one of these links, the Configuration Repositories view is replaced by the Profile Editor page for the selected repository.
- Type: can be one of three types, LDAP, file, or hybrid
  - LDAP: an LDAP repository reads the organizational structure (users, organization, hosts, domains, and so on) of the company from an LDAP server and reads/writes the configuration settings using that same LDAP server.
  - File: a file repository reads the organizational structure of the company from the file system, and reads/writes the configuration settings using that same file system.
  - Hybrid: a hybrid repository reads the organizational structure of the company from an LDAP server, and then read/writes the configuration settings into a file system.
- Location of Organization Data: either a file, if a file repository, or LDAP URL, if an LDAP or hybrid repository. Designates the storage location of the organizational data.
- Location of Profile Data: either a file, if a file repository, or LDAP URL, if an LDAP or hybrid repository. Designates the storage location of the configuration settings. A profile is a named collection of configuration settings that is assigned to users, organizations, hosts, and domains.

You can perform the following actions within the Configuration Repositories table:

- New: creates a new configuration repository

- Remove: removes an existing configuration repository
  - Rename: renames a configuration repository
  - Synchronize: synchronizes configuration repositories
- This function is only enabled when one or two repositories are selected

## ▼ Creating a New Configuration Repository

A configuration repository is a place where organizational hierarchy information is stored, along with profile and assignment data for each element.

### 1 Click the New button in the Configuration Repositories table

The New Configuration Repository Wizard opens. The wizard is used to create a configuration repository for use with the Desktop Manager.

### 2 In the Repository Name field, type a name for the new configuration repository, and then click Next.

### 3 Select the repository type from the Repository Type list.

The wizard can be used to configure three types of repositories:

- LDAP: The hierarchy is taken from LDAP, and all data is stored in LDAP.
- File-Based: The hierarchy is taken from a file, and all data is stored in directories and files.
- Hybrid: The hierarchy is taken from LDAP, but all data is stored in directories and files.

### 4 Click the Next button.

### 5 Enter the details of the LDAP server that you want to configure for this repository.

The Hostname and Port on which the server runs are required. You can also choose whether or not to connect to this server using SSL.

---

**Note** – Note To connect to the server using SSL, the proper certificate needs to be present in the Desktop Manager key store. This key store is located in `/etc/opt/webconsole/keystore`, and either the Certification Authority or the LDAP server certificate needs to be present in that store. A certificate can be added to that location by running the command:

```
keytool -import -file <certificate file> -keystore /etc/opt/webconsole/keystore
```

The default password for that key store is **changeit**. The Java Web Console must be restarted using the `smcwebserver restart` command for the new certificate to be recognized by the Desktop Manager.

---

### 6 Further steps are determined by the choices made in the preceding steps. Follow the instructions provided by the wizard.

---

**Note** – The directory whose URL is provided for the creation of file-based or hybrid repositories must be owned by user noaccess, group noaccess, and have permissions 755 set. This results in read and write permissions for user noaccess, and read only permissions for all others.

---

## ▼ **Removing a Configuration Repository**

- 1 Select the check box corresponding to the configuration repository that you want to remove.**
- 2 Click the Remove button.**  
A confirmation dialog appears.
- 3 If you really want to remove the configuration repository, click the OK button in the confirmation dialog.**

## ▼ **Renaming a Configuration Repository**

- 1 Select the check box corresponding to the configuration repository that you want to rename.**
- 2 Click the Rename button.**  
A Rename dialog opens.
- 3 Type the new name of the configuration repository in the text field of the dialog, and click OK.**

## ▼ **Synchronizing Configuration Repositories**

With the Desktop Manager you can manage multiple back ends in parallel. This allows you to define one back end for testing and one back end for the final production. Configuration changes can now be defined and evaluated first in the test back end. Thereafter you can *synchronize* parts or the complete test back end with the production back end to apply the required changes quickly and safely to the production system. This method can also be used for a simple backup and restore mechanism of the production back end. The Synchronization window allows you to perform this synchronization.

- 1 In the Configuration Repositories window, select the repository that you want to synchronize.**  
The Synchronization window opens.
- 2 Select the source repository from the Source Configuration Repository list.**  
You can change the source repository which serves as source for the changes. This repository is *not* changed.

- 3 (Optional) You can restrict the source repository by clicking the Change button next to Source Starting Point.**

This specifies another element than the root as the source starting point for the synchronization.

A dialog opens, which allows you to navigate to the desired organization or domain element. Once an element is selected, the new path is reflected in the Source Starting Point.
- 4 Select a target repository from the Target Configuration Repository list.**

You can change the target repository which serves as target for the changes. This repository *is* changed.
- 5 (Optional) You can restrict the target repository by clicking the Change button next to Target Starting Point.**

This specifies another element than the root as the target for the synchronization.

A dialog opens, which allows you to navigate to the desired organization or domain element. Once an element is selected, the new path is reflected in the Target Starting Point.
- 6 Click the Compare button.**

The two trees are compared and all differences are listed in a table.
- 7 Select the check boxes next to the table rows containing the profiles you want to synchronize and click the Synchronize button.**

When the synchronization is finished the target repository matches the source repository regarding the selected profiles.

## Profiles — Tree View

The Profiles — Tree View page displays all available profiles that are currently assigned to an element. On the left side of the page is a navigation area, where you can browse through the tree hierarchies to find profiles. On the right side of the page is the Content area, which displays information about any profile selected in the navigation area.

### Navigation Area

The navigation tree is used to browse through the organization and domain trees to select an element. The tree reflects the hierarchy of the elements. If a node has subnodes, a blue triangle is displayed to the left of the node name. Clicking the blue triangle, the node is expanded, and some subnodes are shown.

---

**Note** – By default, only 10 subnodes are displayed for each expanded node. If there are more than 10 subnodes, the 11th “node” does not refer to an element, but indicates that there are nodes that are not displayed: “... some elements not displayed”. Clicking this “node” opens the Find function.

---

To select an element, navigate the tree and click on the desired element. Once an element has been selected, the element is highlighted, and the content area displays the element’s assigned and inherited profiles.

The navigation also provides a mechanism for a recursive search (see [“Searching for an Element” on page 22](#)) and a non-recursive find (see [“Using the Find Function” on page 23](#)), in order to locate elements in the tree hierarchies. Clicking the Search button opens the Search window.

## ▼ Searching for an Element

The Search function allows you to recursively search for an element in the domain or organization hierarchy. It searches for not only the direct children of an element, but also all descendants. To perform a search of only the direct children of an element, use the Find function.

A search opens the Search window, which also allows you to perform a more advanced search.

---

**Note** – The Search feature only supports LDAP and hybrid repositories, but does not support file-based repositories. If you perform a search on a file-based repository, the result is always reported as Elements Found (0).

---

### 1 Type the name or partial name of the element that you wish to search for into the search field.

---

**Note** – The search is not case-sensitive and not strict, for example, the search string “bc” will match “ABC” and “bcd”. An asterisk (\*) denotes an arbitrary sequence of characters within a search string. You can use more than one asterisk at any location in the string. An empty field is equivalent to a field with the search string “\*”.

---

### 2 Click the Search button.

The Search window opens. Any string that you typed into the search field of the Navigation area is moved from that search field to the search field of the Search window, and a search for that string is started automatically. Any results are then displayed in a results table, which shows the name, type and path of any found elements.

---

**Note** – The maximum number of results is capped at 100.

---

If no results were found, or if you left the search field in the Navigation area blank, you can type in the name of an element in the search field of the Search window.

---

**Tip** – Use the type modifier list, located next to the search field, to search for a specific element type, such as “Search domains”.

---

**3 If necessary, add advanced options to the search by clicking the Show Advanced Options button.**

The advanced options consist of the following:

- Restrict to: you can determine the element at which the search algorithm should start to traverse through the tree.
- Results displayed per page: allows you to determine the number of results displayed on every page of the results table.

**4 Click the Search button again to view results for a new or modified search.**

---

**Note** – Clicking the Reset button changes all search parameters to the default and clears the Search field.

---

## ▼ Using the Find Function

The Find function allows you to perform a qualified find for all elements that are *direct* children of the expanded element. This is in contrast to the search function, which provides a recursive search for elements.

**1 Click the Some Elements are not Displayed node in the navigation tree.**

The Find window opens.

**2 Type a string or partial string into the Find field.**

The search is not case-sensitive and not strict, for example, the search string “bc” will match “ABC” and “bcd”. An asterisk (\*) denotes an arbitrary sequence of characters within a search string. You can use more than one asterisk at any location in the search string. An empty field is equivalent to a field with the search string “\*”.

**3 Click the Find button.**

A list of results is displayed. The number of results is capped at 100.

**4 To add a found element to the navigation tree, click the desired element in the results list.**

The Find window closes, and the element is added to the list of subnodes in the navigation tree.

## Content Area

When you select an element, information about that element is displayed in the content area of the Profiles — Tree View page. The content area can contain up to three tables:

- **Assigned Profiles table:** this table is always displayed. This table shows the profiles that are assigned to the currently selected element. It contains three columns, Name, Author, and Last Modified. The value of the Last Modified column changes only if a setting of the profile is changed, but not if the profile itself is renamed, moved or re-prioritized.
- **Inherited Profiles table:** this table is displayed if any element above the selected element in the hierarchy has an assigned profile. This profile is listed in the table. It contains two columns, Name and Assigned To. Assigned To shows the path and name of the element that the profile is assigned to.
- **Users or Roles table:** this table appears when an element of “user” or “role” type is selected. If the element is a user who is a member of at least one role, then those roles are shown in the Roles table. If the element is a role that has members, these members are displayed in the Users table.

## ▼ Creating a new Profile

### 1 Click the New button.

The Profile Editor opens.

### 2 Follow the instructions provided by the Profile Editor.

For more information about how to use the Profile Editor to create a new profile, see [“Profile Editor” on page 29](#).

---

**Note** – When you have finished creating the new profile, it is automatically assigned to the currently selected element.

---

## ▼ Deleting a Profile

### 1 Select the check box corresponding to the profile that you want to delete.

A confirmation dialog appears.



---

**Caution** – It is important to understand that Delete removes all assignments and then physically deletes the profile. You cannot retrieve it, it is gone. Deleting a profile also causes assignments to other elements besides the current element to be removed. To simply remove a profile assignment from an element, use Unassign.

---

### 2 Click the Delete button.

### 3 If you really want to delete the profile, click the OK button in the confirmation dialog.

## ▼ Assigning a Profile

### 1 Click the Assign Profile button.

A dialog opens, listing all available profiles that can be assigned to the selected element.

---

**Note** – Profiles already assigned to the selected element, as well as profiles that are not stored at or above the currently selected element, are not listed.

---

### 2 Select one or more of the listed profiles, and click OK.

The newly assigned profiles now appear in the Assigned Profiles table.

## ▼ Unassigning a Profile

### 1 Select the check box corresponding to the profile that you want to unassign.

### 2 Click the Unassign Profile button.

The profile is now unassigned, and disappears from the Assigned Profiles table.



---

**Caution** – There is no warning dialog, so if you clicked the Unassign Profile button by mistake, you can reverse this by using the Assign Profile function.

---

## ▼ Renaming a Profile

### 1 Select the check box corresponding to the profile that you want to rename.

### 2 Choose Rename from the actions list.

A Rename dialog opens.

### 3 Type the new name of the profile in the text field of the dialog, and click OK.

## ▼ Copying and Moving a Profile

### 1 Select the check box corresponding to the profile that you want to copy or move.

### 2 Choose Copy and Move from the actions list.

The Copy and Move wizard opens.

### 3 Follow the instructions provided by the Copy and Move wizard to perform your task.

## ▼ Importing Profiles

- 1 **Make sure all check boxes are deselected.**
- 2 **Choose Import from the actions list.**

The Import Profile dialog opens.
- 3 **Enter the path of the file that you want to import, or click the Browse button to choose a file from the default file chooser dialog of your operating system.**
- 4 **Click the Import button.**

## ▼ Exporting Profiles

- 1 **Select the check box corresponding to the profile that you want to export.**
- 2 **Choose Export from the actions list.**

A dialog opens, which allows you to specify the location to which you want to export the file.
- 3 **Select the Save to Disk option in the dialog and click OK.**
- 4 **Choose the location where you want to save the exported file and click OK.**

## ▼ Viewing Effective Settings

Effective Settings is a read-only view of the effective configuration settings for a given element, such as an organization, domain or host.

The effective settings of a specific user are the result of a merge process, including the configuration settings stored at the involved organization level(s), role(s), and the user. You can display the merged configuration settings of a selected element in a separate Effective Settings window. This allows for easy verification and also error debugging. An Effective Settings report can also be printed and stored for archival purposes.

- 1 **Click the Effective Settings button in the Assigned Profiles table.**

When first opened, the effective settings for the selected element in the navigation tree are displayed.

The Effective Settings window consists of the following two sections:

- In the Modify Effective Settings Parameters section, the Create the Effective Settings Using area lets you choose which tree to take into account. The Select an element from the Domain Tree option lets you choose which entities to generate a Effective Settings report for. Use the Browse button to select an element in the organization or domain tree.
- The Effective Settings section displays the following:

- The location of the selected element in the hierarchy and the hostname.
- The time and date when the current effective settings report was generated.
- In the Profiles Involved section, the list of profiles involved in generating the effective settings report for the current element. They are listed in merge order from most relevant to least relevant. Each profile is a link that opens the Profile Editor window for the profile.
- The Settings Summary area displays the configuration settings for the current effective settings report, grouped into categories. The Settings Summary contains Name, Value, and Status fields.

The Status column contains entries of the type “Defined In <profileName>”, where <profileName> is a link that opens the Profile Editor for that profile on the Configuration Settings tab, and navigates to the particular setting within this profile.

- 2 **You can modify a generated effective settings report by changing any relevant effective settings parameters, and then clicking the Generate Effective Settings button at the top of the Effective Settings window.**

A new report is then displayed in the Effective Settings window.

- 3 **To print an effective settings report, click the Print button at the top or bottom of the Effective Settings Window.**

## Profiles — All

The Profiles — All page displays all available profiles, including those that are not assigned to an element. The profiles are shown in two tables, one for domain profiles and one for organization profiles.

The profile tables let you perform the following actions:

- Creating a new Profile
- Deleting a Profile
- Renaming a Profile
- Copying and Moving a Profile
- Importing Profiles
- Exporting Profiles

### ▼ Creating a new Profile

- 1 **Click the New button in the desired profile table, either organization or domain.**

The Profile Editor wizard opens.

**2 Follow the instructions provided by the wizard.**

For more information about creating a new profile using the Profile Editor, see [“Profile Editor” on page 29](#).

## ▼ **Deleting a Profile**

- 1 Select the check box corresponding to the profile that you want to delete.
- 2 Click the Delete button.

## ▼ **Renaming a Profile**

- 1 Select the check box corresponding to the profile that you want to rename.
- 2 Choose Rename from the actions list.  
A dialog box appears.
- 3 Type the new name of the profile in the text field of the dialog, and click OK.

## ▼ **Copying and Moving a Profile**

- 1 Select the check box corresponding to the profile that you want to copy or move.
- 2 Choose Copy and Move from the actions list.  
The Copy and Move wizard opens.
- 3 Follow the instructions provided by the Copy and Move wizard to perform your task.

## ▼ **Importing Profiles**

- 1 Make sure all check boxes are deselected.
- 2 Choose Import from the actions list.  
The Import Profile dialog opens.
- 3 Enter the path of the file that you want to import, or click the Browse button to choose a file from the default file chooser dialog of your operating system.
- 4 Click the Import button.

## ▼ Exporting Profiles

- 1 Select the check box corresponding to the profile that you want to export.
- 2 Choose **Export** from the actions list.  
A dialog opens, which allows you to specify the location to which you want to export the file.
- 3 Select the **Save to Disk** option in the dialog and click **OK**.
- 4 Choose the location where you want to save the exported file and click **OK**.

## Profile Editor

The Profile Editor is used to create new profiles and edit existing profiles. You can also view the current settings and options for the currently selected profile.

You can open the Profile Editor by either clicking the name of a profile in one of the tables in the Profiles — Tree View content area or in the Profiles — All page.

---

**Tip** – To create a new profile, see [“Creating a new Profile” on page 24](#).

---

The Profile Editor contains the following tab pages:

- General Properties
- Configuration Settings
- Assigned Elements
- Settings Summary
- Advanced Options

## General Properties

The General Properties page allows you to specify general information about a profile. This page is the default view of the Profile Editor window.

---

**Note** – The Profile Editor appears when the user creates a new profile by clicking the New button.

---

## ▼ **Specifying General Properties for a Profile**

**1** In the **Name** field, type the desired name of the profile.

**2** Add any comments into the **Comments** field.

You can leave the **Comments** field blank.

---

**Note** – The **Author** and **Last Modified** fields are read-only, and cannot be modified. The **Author** field displays the profiles author, and the **Last Modified** field shows the date and time of the last modification.

---

**3** Click the **Save** button.

If you close the window before saving any new information, and warning dialog appears that prompts you to save the settings.

## **Configuration Settings**

The **Configuration Settings** page allows you to view and edit the configuration settings that are stored in the selected profile.

## ▼ **Viewing and Editing a Profile's Configuration Settings**

**1** To view a profile's configuration setting, navigate to the setting from the list of categories on the right side of the page.

The configuration settings are organized hierarchically into categories that directly correspond to the configuration options of desktop applications, such as StarOffice™, Mozilla, Evolution or Gnome.

**2** Click the desired configuration setting.

The selected profile's settings appear, which displays the available settings for that profile. If necessary, additional context information is provided, such as the effects of individual settings or the allowed value range.

---

**Note** – If you select a category rather than a configuration setting, a table appears that lists that categories settings or sub-categories.

---

**3 For each configuration setting, you can either specify a concrete value, or you can leave the setting undefined.**

If you leave the setting undefined, the application will not be affected by this profile setting, and instead takes advantage of its own factory built-in default value. In general, the Profile Editor lists and applications default value through additional help text or, if possible, through a marker directly in the setting's options list.

---

**Note** – It is important to understand that several profiles might affect the same user. For example, you can create a profile with settings for a specific organization. These settings are automatically available for all users that are part of this organization. Similarly, you can define additional profiles with configuration settings for sub-organizations. These may overwrite settings of the parent organization, for example, different proxy settings for sub-organizations would be defined, but leave other settings unchanged. The effective settings for a specific user would then be the result of a merge process including all profiles assigned to the organization, sub-organizations and probably the user.

---

---

**Tip** – In some cases you might want to control this merge process and want to enforce a configuration setting, meaning that it should not be possible for other profiles to overwrite your setting and to define a different value. The Profile Editor allows to specify for each setting the desired merging behavior. The default is that other "profiles may overwrite" a setting. Deselecting this option will enforce the setting and other profiles can no longer overwrite your value.

---

## Assigned Elements

The Assigned Elements page lists the elements that are assigned to the current profile, and allows you to assign or unassign elements.

The current profiles elements are displayed in the Assigned Elements table, which contains the following information:

- Name: the name of the element to which the current profile is assigned.
- Path: the location of the element to which the current profile is assigned.

### ▼ Assigning an Element to a Profile

**1 Click the Assign Element button.**

A window containing a navigation tree appears.

**2 Navigate the tree, and select the desired element.**

---

**Tip** – If you can't find the desired element in the tree, you can click the Search button to open the Search window and find the element.

---

The element then appears in the Assigned Elements table, now assigned to the current profile.

## ▼ Unassigning an Element from a Profile

**1** Select the check box corresponding to the element that you want to unassign.

**2** Click the Unassign Element button.

The element is now not assigned to the current profile, and disappears from the Assigned Elements table.

## Settings Summary

The Settings Summary is a read-only view of all configuration settings that contain data for the selected profile.

The Settings Summary page contains the Settings Summary table. This table shows the configuration settings that contain data for the current profile, grouped into categories. The table has the following fields:

- Name: the setting name
- Value: the setting value
- Status: the status column contains entries of the type “Defined in <profileName>”, where <profileName> is a link that switches the Profile Editor to the Configuration Settings page, and navigates to the particular setting within this profile.

## Advanced Options

The Advanced Options page allows you to specify advanced options for the current profile. There are two main options available on this page:

- Applicable from: allows you to move the profile to a different storage location in the organization or domain tree. By moving the storage location, you are also changing the set of elements that the profile can be assigned or applied to. This is because a profile can only be assigned to the element it is stored at, or assigned to the elements that are in the subtree below the element at which it is stored.



---

**Caution** – You can not change the “Application from” parameter in a way that the resulting layout would violate the rule mentioned above. For example, this would happen if you move a profile down the tree hierarchy, and an existing assignment connecting the profile to an element below the profile’s storage location would, after the movement of the profile, connect the profile to the same element. But now, the element is above the storage location of the profile. If the you try this, the action is cancelled, and a warning message is displayed.

---

- **Merge Order:** specifies the order in which all profiles stored at the same element are merged. Because more than one profile can be assigned to a single element and more than one profile can define the same setting, the order in which the profiles are merged for a single element must be defined. This resolves the conflicts of one setting being defined in two or more profiles assigned to the same element.

The storage location is used first to determine the merge order. The profile stored at an element that is located closer to the root of the tree is merged first. As a consequence, the profile that is stored deeper in the tree hierarchy is merged secondly. The settings of a profile merged later overwrite the settings of a profile merged earlier.

## ▼ **Changing the Application from Location**

- 1 **Click the Change button.**  
A window containing a navigation tree appears.
- 2 **Navigate the tree, and select the desired element.**

---

**Tip** – If you can’t find the desired element in the tree, you can click the Search button to open the Search window and find the element.

---

- 3 **Click the OK button.**

## ▼ **Assigning a Merge Order to a Profile**

- 1 **Select the desired profile in the Selected list.**
- 2 **Use the buttons on the right of the list to move the profile up or down within the list.**



# Using the Command Line Interface

---

This chapter provides information about the Sun Desktop Manager command line interface.

## Overview of the Desktop Manager CLI

The CLI is used to create, manipulate, export, import and delete configuration profiles. As with the Desktop Manager GUI, the CLI allows the following actions:

- Profiles can be assigned and unassigned to or from elements.
- Profiles can be exported and imported in XML format to or from zip files. The profile settings in these zip files can then be created, edited or deleted prior to importing the profiles.

---

**Note** – The CLI does not provide the equivalent of the GUI functionality for the following functions:

- Navigation of element hierarchy.
  - Viewing merged profile settings for elements.
  - Generating reports.
  - Access to file based or hybrid back end, provides access to LDAP back end only.
- 

## Working with the CLI

### Invoking CLI Commands

The CLI consists of the command `pgtool`, which operates in a single-line command mode that executes one command at a time. `pgtool` contains a number of sub-commands, options and operands, which are described in “[Command Summary](#)” on page 44. The options can be specified using either a full or a short keyword. In the following commands descriptions, the full keywords are used, but the shortcuts that correspond to the commands can be found in [Table 3–2](#).

## Bootstrapping Information Required by the CLI

Bootstrapping information is required in order to locate and interrogate the datastore containing the elements and profiles. The bootstrapping information that is required is server, port number, base distinguished name (DN) and username of the administrator. This information can be specified at the command line or in a bootstrapping file.

### Bootstrapping file

The bootstrapping information can be read from a properties file. The location of this file can be specified in the command line by the `--file` option.

`--file=<bootstrap file>` : fully qualified path to a bootstrapping file. The default file is `$HOME/pgtool.properties`.

Example: `--file=/var/opt/apoc/cli.properties`.

The format of the file is described in the Appendix A, “Configuration Parameters,” in *Sun Desktop Manager 1.0 Installation Guide*.

### Bootstrapping options

The bootstrapping can be specified at the command line using the `--url` and `--username` options.

`--url=<url>` : the URL identifying the datastore. The URL format is `ldap://<hostname>:<port>/<base name>` with `<hostname>` the name of the server (default is `localhost`), `<port>` the port number on the server (default is 389) and `<base name>` the distinguished name of the base entry.

Example: `--url=ldap://server1.sun.com:399/o=apoc`.

`--username=<username>` : the username of the administrator in the format used by the storage backend. The administrator is then prompted for a password. If this option is not used, and the administrator has not used the `pgtool login` sub-command, then the administrator is prompted to enter a username and password.

Example: `--username="jmonroe"`.

## Authentication by Username and Password

A username and password are required for each execution of a command.

### Authentication by login command

The CLI provides a login command to allow username/password pairs to be stored in a credentials file in the administrator’s home directory. This file is named `.apocpass`. The `.apocpass` file has restricted access.

If authentication is successful, a username/password pair entry is added to the `.apocpass` file. The key for this pair is made up of the server/port/base DN and the username so that username/password pairs can be stored for other back ends in the same file.

Once the login command has successfully completed, other CLI commands can be executed without the necessity of specifying a username or password.

For more details on how to use the `login` command, see [“Login” on page 42](#).

## Authentication for the other commands

For other commands, the CLI first checks to see if an `.apocpass` file exists for the current user.

If the file does not exist, the user is prompted for a username and password. If this username and password is successfully authenticated, the command is executed.

If the credentials file does exist and a username has been specified at the command line, the CLI looks for an entry for the host, port, base DN and username. If an entry exists, the stored user DN and password is used to execute the command, otherwise the user is prompted for a password.

If a username is not specified at the command line, the `.apocpass` file is searched for keys using the host/port and base DN combination. If there is a unique entry for this combination, the stored user DN and password is used to execute the command. If the entry is not unique, the user is prompted for a username. If this matches an entry, the stored user DN and password is used to execute the command. If this does not match, then the user is prompted for a password.

Where the user is prompted for a password, an entry from the `.apocpass` file for this host/port/baseDN combination is used to authenticate the username and password. If such an entry does not exist, anonymous access is used for the authentication.

## Running a Command

Each use of a command creates and initializes a connection to the datastore and then exits once the command has been executed. If the command exits with an error, no changes were applied to the configuration profiles.

## Representing Elements

An element is represented using the LDAP full Distinguished Name (DN).

Example: `uid=jmonroe,ou=People,o=apoc`.

# CLI Commands

This section describes the sub-commands of Desktop Manager CLI functionality.

---

**Note** – For backward compatibility reasons, the CLI uses the `--entity` option to specify what is now called an “element” in the Desktop Manager user interface. The terms are interchangeable. There is no `--element` option.

---

## Help

To obtain a list of all available CLI commands, type `pgtool --help`.

## Version Information

To display version information, type `pgtool --version`.

## Add

Assigns a profile to an element.

### Syntax

```
add [--scope=<user|host>] [--entity=<entity>] <profile name> <target entity>
```

`--scope=<user|host>` : the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope.

`--entity=<entity>` : the element where the profile is located. If not specified, then defaults to root organization or domain element, depending on the `--scope` option.

`<profile name>` : the name of the profile to be assigned to the element. If no profile or more than one profile with this name and this scope can be found at this layer, then the command exits with an error.

`<target entity>` : the element to which the profile is assigned.

**EXAMPLE 3-1** Adding a profile to an element

```
% pgtool add UserProfile1 cn=Role1,o=staff,o=apoc
```

The profile “UserProfile1” located in root organization was assigned to the element “cn=Role1,o=staff,o=apoc”.

## Create

Creates a new, empty profile.

### Syntax

```
create [--name=<profile name>] [--scope=<user|host>] [--entity=<entity>]  
[--priority=<priority>]
```

`--name=<profile name>` : the name of the profile to be created. If a profile with this name and this scope already exists at this layer, then the command exits with an error. If not specified, then the command defaults to the first profile name available in the series, for example, “NewProfile”, “NewProfile2”, “NewProfile3”, ...

`--scope=<user|host>` : the scope for the profile, which can be either user or host. If not specified, then the command defaults to the user scope.

`--entity=<entity>` : the element where the profile is created. If not specified, then the command defaults to root organization or domain element, depending on the `--scope` option.

`--priority=<priority>` : a strictly positive integer specifying the priority of the profile. If the priority specified is the same as that of an existing profile of this scope at this layer, then the `priority` option is ignored. If not specified, then the created profile will have the highest priority at this layer.

**EXAMPLE 3-2** Creating a new profile

```
% pgtool create --scope=host --name=NewHostProfile1
```

Creates a new profile called “NewHostProfile1” in the domain root and whose scope is “host”.

## Delete

Deletes a profile.

### Syntax

```
delete --name=<profile name> [--scope=<user|host>] [--entity=<entity>]
```

`--name=<profile name>` : the name of the profile to be deleted. This option is mandatory. If no profile or more than one profile with this name and this scope can be found at this layer, then the command exits with an error.

`--scope=<user|host>` : the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope.

`--entity=<entity>` : the element where the profile is located. If not specified, then defaults to root organization or domain element, depending on the `--scope` option.

**EXAMPLE 3-3** Deleting a profile

```
% pgtool delete --scope=host --name=renamedNewHostProfile1
```

Deleted the “renamedNewHostProfile1” profile located in root domain.

## Export

Exports the profiles contained in a profile to the specified target, using a zip file format.

### Syntax

```
export --name=<profile name> [--scope=<user|host>] [--entity=<entity>] <target>
```

`--name=<profile name>` : the name of the profile to export. This option is mandatory. If no profile or more than one profile with this name and this scope can be found at this layer, then the command exits with an error.

`--scope=<user|host>` : the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope.

`--entity=<entity>` : the element where the profile is located. If not specified, then defaults to root organization or domain element, depending on the `--scope` option.

`<target>` : the full path to the zip file where the profile is to be exported. If the path does not specify any filename, then the filename defaults to `<profile name>.zip`. If the target is not writable, then the command exits with an error.

**EXAMPLE 3-4** Exporting a profile

```
% pgtool export --scope=host --name=HostProfile1 /tmp/newdir
```

Exported “HostProfile1” (located in root domain) to `HostProfile1.zip`, which was created in new directory `/tmp/newdir`.

## Import

Imports a profile stored in zip file format from the specified source. The source contains the profiles to be imported into the profile.

### Syntax

```
import [--name=<profile name>] [--scope=<user|host>] [--entity=<entity>]  
[--priority=<priority>] <source>
```

`--name=<profile name>` : the name of the profile to be imported. If not specified, then defaults to the name of the `.zip` source file. If a profile of this name and scope already exists at this layer, then it is overwritten.

- `--scope=<user|host>` : the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope.
  - `--entity=<entity>` : the element where the profile is located. If not specified, then defaults to root organization or domain element, depending on the `--scope` option.
  - `--priority=<priority>` : a strictly positive integer specifying the priority of the profile. If the priority specified is the same as that of an existing profile of this scope at this layer, then this option is ignored. If not specified, then the imported profile will have the highest priority at this layer.
- `<source>` : the full path to the zip file containing the profiles to import.

EXAMPLE 3-5 Importing a profile

```
% pgtool import --scope=host --name=NewHostProfile1 --priority=7
/tmp/HostProfile1.zip
```

Profile with name “NewHostProfile1”, scope “host”, priority “7” and located in root domain, was imported from HostProfile1.zip.

## List

Lists all profiles or element/profile assignments.

If neither `--name` option nor `--entity` option is specified, then all the user or host profiles are listed, depending on the `--scope` option.

If the `--name` option is specified, then all the elements assigned to this profile are listed.

If the `--entity` option is specified, then all the profiles assigned to the element specified are listed.

### Syntax

```
list [--scope=<user|host>] [--entity=<entity>] [--name =<profile name>]
```

- `--scope=<user|host>` : the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope.

- `--entity=<entity>` : the element where the profile is located, if the `--name` option is specified. Otherwise, the element to which profiles to be listed are assigned. If not specified in the former case, then defaults to the root organization or domain element, depending on the `--scope` option.

- `--name=<profile name>` : the name of the profile whose assigned elements are to be listed. If no profile or more than one profile with this name and this scope can be found at this layer, then the command exits with an error.

EXAMPLE 3-6 Listing assignments

```
% pgtool list
```

Lists the profiles assigned to root organization.

EXAMPLE 3-6 Listing assignments (Continued)

```
% pgtool list --scope=host --name=HostProfile1
```

Lists elements assigned to profile “HostProfile1” located in root domain.

## Login

Stores the username and password for the datastore back end after a successful authentication to this datastore. This username and password can then be used in future invocations of `pgtool`.

The credentials are stored in a file named `.apocpass` in the administrator’s home directory. If this file already exists and it does not have the correct permissions, then the command exits with an error.

If a username is specified, the administrator is prompted for a password, otherwise the administrator is prompted for a username and a password. The username and password are authenticated using anonymous access to the datastore. If anonymous access is not supported, then the administrator is prompted to enter an authorized username and a password. If authentication by the authorized username fails, then the command exits with an error.

Once authenticated, the username/password pair entry is stored in the administrator’s `.apocpass` file using a key made up of a combination of the host/port/base DN and the username so that username/password pairs can be stored for other back ends in the same file.

The bootstrapping information can be specified in the command using the `--file` option to indicate the file where this information is found, or using the `--url` option to directly specify this information. For more information about bootstrapping, see [“Bootstrapping Information Required by the CLI” on page 36](#)

If bootstrapping information is not available or the credentials file cannot be created, then the command exits with an error.

If the credentials file is successfully created, then it is not necessary to specify username and password for subsequent `pgtool` commands using this storage back end. The username and password details stored in the credentials file are used.

### Syntax

```
login [--username=<username>] [--file=<bootstrap file>] [--url=<url>]
```

`--username=<username>`: the username of the administrator in the format used by the storage back end, for example “jmonroe”.

`--file=<bootstrap file>`: fully qualified path to a bootstrapping file. The default file is `$HOME/pgtool.properties`. This option is not compatible with the `--url` option.

`--url=<url>`: the URL identifying the datastore. The URL format is `ldap://<hostname>:<port>/<base name>` with `<hostname>` the name of the server (default is `localhost`), `<port>` the port number on the server (default is 389) and `<base name>` the distinguished name of the base entry.

**EXAMPLE 3-7 Login**

```
% pgtool login --username=jmonroe [Enter the correct password when prompted]
```

A file called `~/ .apocpass` created with entry for “jmonroe” and “*password*”, file has permissions 600. The information about the back end to connect to is found in the default bootstrapping file `~/pgtool.properties`

## Modify

Changes the priority of a profile.

### Syntax

```
modify [--scope=<user|host>] [--entity=<entity>] <profile name> <priority>
```

`--scope=<user|host>`: the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope.

`--entity=<entity>`: the element where the profile is located. If not specified, then defaults to root organization or domain element, depending on the `--scope` option.

`<profile name>`: the name of the profile to change priority to. If no profile or more than one profile with this name and this scope can be found at this layer, then the command exits with an error.

`<priority>`: a strictly positive integer specifying the priority of the profile. If the priority specified is the same as that of an existing profile of this scope at this layer, then the command exits with an error.

**EXAMPLE 3-8 Changing the priority of a profile**

```
% pgtool modify UserProfile1 15
```

Changed the priority of “UserProfile1”, located in root organization, to 15.

## Remove

Unassigns a profile from an element.

### Syntax

```
remove [--scope=<user|host>] [--entity=<entity>] <profile name> <target entity>
```

`--scope=<user|host>`: the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope.

`--entity=<entity>`: the element where the profile is located. If not specified, then defaults to root organization or domain element, depending on the `--scope` option.

<profile name> : the name of the profile to unassign from the target element.

<target entity>: the element to from which to unassign the profile.

**EXAMPLE 3-9** Unassigning a profile from an element

```
% pgtool remove UserProfile1 cn=Role1,o=staff,o=apoc
```

The profile “UserProfile1” located in root organization was unassigned from the element “cn=Role1,o=staff,o=apoc”.

## Rename

Renames a profile.

### Syntax

```
rename [ --scope=<user|host> ] [ --entity=<entity> ] <profile name> <newname>
```

--scope=<user|host> : the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope.

--entity=<entity> : the element where the profile is located. If not specified, then defaults to root organization or domain element, depending on the --scope option.

<profile name> : the current name of the profile to rename.

<newname> : the new name for the profile. If a profile of this name and scope already exists at the same layer, then the command exits with an error.

**EXAMPLE 3-10** Renaming a profile

```
% pgtool rename NewUserProfile2 renamedNewUserProfile2
```

Renamed “NewUserProfile2” located in root organization to “renamedNewUserProfile2”.

## Command Summary

**TABLE 3-1** Commands

Command	Description
add	Assigns a profile to an element.
create	Creates a new, empty profile.

TABLE 3-1 Commands (Continued)

Command	Description
delete	Deletes a profile.
export	Exports the profiles contained in a profile to the specified target, using a zip file format.
import	Imports a profile stored in zip file format from the specified source. The source contains the profiles to be imported into the profile.
list	Lists all the profiles or the profiles assigned to an element or the elements a profile has been assigned to.
login	Stores the username and password for this datastore back end in a file in the administrator's home directory. This username and password can then be used in future invocations of <code>pgtool</code> .
modify	Changes the priority of a profile.
remove	Unassigns a profile from an element.
rename	Renames a profile.

TABLE 3-2 Options

Option	Description
-b <base name>, --base=<base name>, <i>DEPRECATED</i>	The root entry of the storage back end. The format for this base entry is determined by the storage back end used. For example, an LDAP storage back end with a root entry of <code>o=apoc</code> : <code>--base o=apoc</code> . <i>DEPRECATED</i> : use the <code>--url</code> option instead
-e <entity>, --entity=<entity>	Represents a user, role, organization, host or domain element. The entry format for the element is determined by the storage back end used. For example, an LDAP storage back end with a user "jmonroe": <code>-e uid=jmonroe,ou=People,o=Staff,o=apoc</code> .
-f <file>, --file=<file>	The fully qualified path to the file detailing the bootstrapping information to be used in the execution of this command. For example: <code>-f /var/opt/apoc/cli.properties</code> .
-h <hostname>, --hostname=<hostname>, <i>DEPRECATED</i>	The name of the host for the storage back end. For example: <code>--host=server1.sun.com</code> . <i>DEPRECATED</i> : use the <code>--url</code> option instead

TABLE 3-2 Options (Continued)

Option	Description
-i <priority>, --priority=<priority>	A strictly positive integer specifying the priority of the profile. For example: -i 12.
-l <url>, --url=<url>	the URL identifying the datastore. The URL format is ldap://<hostname>:<port>/<base name> with <hostname> the name of the server (default is localhost), <port> the port number on the server (default is 389) and <base name> the distinguished name of the base entry. For example: --url=ldap://server1.sun.com:399/o=apoc.
-m <name>, --name=<name>	The name of the profile. For example: --name=UserProfile1.
-p <port>, --port=<port>, <i>DEPRECATED</i>	The port number for the storage back end. For example: -p 399. <i>DEPRECATED</i> : use the --url option instead
-s <scope>, --scope=<scope>	the scope for the profile, which can be either user or host. If not specified, then defaults to the user scope. For example: --scope=host.
-t <type>, --type=<type>, <i>DEPRECATED</i>	The type of storage back end. This defaults to LDAP. For example: -t LDAP. <i>DEPRECATED</i> : use the --url option instead
-u <username>, --username=<username>	The username of the administrator in the format used by the storage back end. The administrator is then prompted for a password. If this option is not used, and the administrator has not used the pgtool login sub-command, then the administrator is prompted to enter a username and password. For example: --username=jmonroe.
-?, --help	Displays the help information
-V, --version	Displays the version information.

# ◆ ◆ ◆ CHAPTER 4

## Migration

---

The Desktop Manager contains a number of changes and new features to simplify the management of configuration settings. If you want to migrate a Java Desktop System Configuration Manager 1.1 installation to a Sun Desktop Manager 1.0 installation, you need to migrate your system piece by piece. For example, migrate the server component first, followed by the client component (the Configuration Agent). If you want, you can continue to use the Java Desktop System Configuration Manager 1.1 Configuration Agent with the Desktop Manager but without any of the Desktop Manager features, such as delegated administration or the file-based profile repository.

### **Object Model and Terminology**

The underlying object model for the Desktop Manager has been simplified, but remains compatible with Java Desktop System Configuration Manager. The Desktop Manager manages configuration settings, which are collected and stored within profiles. You can assign profiles to hosts and groups of a domain hierarchy, to users and groups of an organizational hierarchy. You can also assign more than one profile to a single element. Profiles that are assigned to higher level elements of a hierarchy are inherited by their children.

### **Profile Management**

Profile groups that you created in Java Desktop System Configuration Manager 1.1 appear as "profiles" in the Desktop Manager. You can only assign settings to an element after you create a corresponding profile. Configuration settings that you assigned to entities in Java Desktop System Configuration Manager automatically appear as profiles with the label "Settings for ...." in the Desktop Manager.

## Profile Format

When you use the Configuration Wizard in the Desktop Manager to configure an LDAP server, the wizard also detects existing Java Desktop System Configuration Manager 1.1 profiles, and offers to migrate the profiles to the new format. This migration is optional and only improves the read performance of the Desktop Manager agents. If you still need to support Configuration Manager 1.1 agents, do not migrate the profiles. You can still view and edit these profiles without losing compatibility.

## Delegated Administration

With the Java Desktop System Configuration Manager 1.1, you can only store profiles under the root of the organization and domain hierarchy in the LDAP server. With the Desktop Manager, you can store profiles in different locations in the organization and domain hierarchy. This delegated administration feature is especially useful if different administrators are responsible for the configuration of different branches of the organization or domain hierarchy, which should not affect each other.

When you migrate to the Desktop Manager, existing Configuration Manager 1.1 profiles on the LDAP server remain under the root of the organization and domain hierarchy. If you want, you can move these with the Copy/Move Wizard or in the advanced options settings of the profile to a new location on the LDAP server. However, when you move an existing profile to a new location, the profile is no longer compatible with the Java Desktop System Configuration Manager 1.1. In other words, only use the delegated administration feature if you have completely migrated your Configuration Manager 1.1 agents to the Desktop Manager.