



# Sun Java™ System Directory Editor 1 2004Q4 Installation and Configuration Guide

---

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

Part No: 819-1078-10

Copyright © 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

THIS PRODUCT CONTAINS CONFIDENTIAL INFORMATION AND TRADE SECRETS OF SUN MICROSYSTEMS, INC. USE, DISCLOSURE OR REPRODUCTION IS PROHIBITED WITHOUT THE PRIOR EXPRESS WRITTEN PERMISSION OF SUN MICROSYSTEMS, INC.

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.

This distribution may include materials developed by third parties.

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 in other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Java, Solaris, JDK, Java Naming and Directory Interface, JavaMail, JavaHelp, J2SE, iPlanet, the Duke logo, the Java Coffee Cup logo, the Solaris logo, the SunTone Certified logo and the Sun ONE logo 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 architecture developed by Sun Microsystems, Inc.

Legato and the Legato logo are registered trademarks, and Legato NetWorker, are trademarks or registered trademarks of Legato Systems, Inc. The Netscape Communications Corp logo is a trademark or registered trademark of Netscape Communications Corporation.

The OPEN LOOK and Sun(TM) 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.

Products covered by and information contained in this service manual are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

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 © 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuels relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs des brevets américains listés à l'adresse <http://www.sun.com/patents> et un ou des brevets supplémentaires ou des applications de brevet en attente aux Etats - Unis et dans les autres pays.

CE PRODUIT CONTIENT DES INFORMATIONS CONFIDENTIELLES ET DES SECRETS COMMERCIAUX DE SUN MICROSYSTEMS, INC. SON UTILISATION, SA DIVULGATION ET SA REPRODUCTION SONT INTERDITES SANS L'AUTORISATION EXPRESSE, ECRITE ET PREALABLE DE SUN MICROSYSTEMS, INC.

Cette distribution peut comprendre des composants développés par des tierces parties.

Des parties de ce produit peuvent être dérivées des systèmes 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, Java, Solaris, JDK, Java Naming and Directory Interface, JavaMail, JavaHelp, J2SE, iPlanet, le logo Duke, le logo Java Coffee Cup, le logo Solaris, le logo SunTone Certified et le logo Sun[tm] ONE 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.

Legato, le logo Legato, et Legato NetWorker sont des marques de fabrique ou des marques déposées de Legato Systems, Inc. Le logo Netscape Communications Corp est une marque de fabrique ou une marque déposée de Netscape Communications Corporation.

L'interface d'utilisation graphique OPEN LOOK et Sun(TM) 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.

Les produits qui font l'objet de ce manuel d'entretien et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes biologiques et chimiques ou du nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

# Contents

Who Should Use This Book . . . . .	viii
Before You Read This Book . . . . .	viii
How This Book Is Organized . . . . .	ix
Conventions Used in This Book . . . . .	x
Typographic Conventions . . . . .	x
Symbols . . . . .	xi
Mnemonics . . . . .	xi
Default Paths and File Names . . . . .	xi
Books in This Documentation Set . . . . .	xii
Related Documentation . . . . .	xii
Accessing Sun Resources Online . . . . .	xii
Contacting Sun Technical Support . . . . .	xiii
Related Third-Party Web Site References . . . . .	xiii
Sun Welcomes Your Comments . . . . .	xiv
 <b>Chapter 1 Before You Install . . . . .</b>	 <b>1</b>
Supported Software Environments . . . . .	1
Operating Systems . . . . .	2
Application Servers . . . . .	2
Browsers . . . . .	3
Directory Servers . . . . .	3
Recommended Directory Editor Configuration . . . . .	3
Memory Requirements . . . . .	4
Setup Task Flow . . . . .	4
Prerequisite Tasks . . . . .	5
What's Next? . . . . .	5

<b>Chapter 2 Installing Directory Editor</b>	<b>7</b>
Before You Begin	7
Installation Steps	8
Step 1: Install an Application Server	8
Step 2: Install Directory Editor	10
Step 3: Install Directory Editor on an Application Server	13
Step 4: Specify the Startup Properties	28
Step 5: Specify the Managed Directory Properties	33
What's Next?	34
 <b>Chapter 3 Getting Started</b>	 <b>35</b>
Product Features	35
Logging In	36
Changing Your LDAP Password	37
Navigating Directory Editor	38
Browsing Directories	40
Using the Browse Button	42
Running in Non-Production Mode	43
Using Online Help	43
 <b>Chapter 4 Creating and Editing Objects</b>	 <b>45</b>
Creating Objects	45
Accessing the Create Page	46
Creating Users	46
Creating Groups	60
Creating Organizational Units	72
Creating Domains	77
Creating Localities	80
Editing Objects	82
Accessing an Edit Page	82
Enabling or Disabling Objects	87
Renaming Objects	89
Deleting Objects	91
 <b>Chapter 5 Working With Forms</b>	 <b>93</b>
Overview	93
How Do Forms Work?	95
Why Edit the Forms?	95
Accessing the Forms Page	97
Customizing Forms	98
Opening a Form	99

Working with the Forms Page .....	100
Performing Common Tasks .....	103
Creating New Forms .....	111
Testing Your Form .....	113
 <b>Chapter 6 Searching Directories .....</b>	<b>115</b>
Executing a Basic Search .....	115
Executing an Advanced Search .....	118
Executing a Filtered Search .....	124
Editing the Search Page Configuration .....	126
Accessing the Search Configuration Page .....	126
Specifying Default Settings .....	127
Configuring the Results Table .....	128
Specifying Search Types .....	130
 <b>Chapter 7 Configuring Directory Editor .....</b>	<b>133</b>
Controlling User Access .....	133
Understanding Roles .....	134
Understanding Principals .....	135
Understanding Accessible Components .....	135
Accessing the Authorization Page .....	136
Defining Directory Editor Roles .....	137
Editing Roles .....	140
Deleting Roles .....	141
Working with Naming Attributes .....	142
Accessing the Naming Attributes Page .....	142
Creating New Object Class Naming Attribute Mappings .....	144
Editing Naming Attributes .....	146
Deleting Selected Naming Attributes .....	147
 <b>Chapter 8 Backing Up and Restoring Configurations .....</b>	<b>149</b>
Accessing the Backup/Restore Page .....	150
Backing Up Your Configuration .....	150
Restoring Your Configuration .....	152
 <b>Chapter 9 Removing the Software .....</b>	<b>153</b>
Removing the Software .....	153
Using the Directory Editor Uninstaller .....	154
Uninstalling Manually From a Windows Machine .....	154
Uninstalling Manually From a Unix Machine .....	155
Undeploying Directory Editor from Your Application Server .....	156
From Sun Application Servers .....	156

From Tomcat Application Servers .....	156
From WebLogic Application Servers .....	157
From WebSphere Application Servers .....	157
<b>Chapter 10 Error Logging and Troubleshooting .....</b>	<b>159</b>
Before You Call Technical Support .....	159
Obtaining Information About the Problem .....	160
Information in the Page .....	160
Information in the Logs .....	161
Common Problems and Solutions .....	165
java.lang.OutOfMemoryError .....	165
Failed to Save Startup Configuration Properties .....	165
java.security.AccessControlException .....	167
WebSphere is Extremely Slow During Startup .....	168
Changing the Object Class and Attributes Used for Configuration Objects .....	169
<b>Appendix A HTML Components Used to Define Directory Editor Forms .....</b>	<b>171</b>
Overview .....	171
What Are HTML Components? .....	172
Component Classes .....	172
Basic Component Classes .....	172
Container Classes .....	173
Page Processor Requirements for HTML Components .....	173
Hidden Parameters .....	173
Component Subclasses .....	174
Naming Conventions .....	174
Data Types .....	175
Base Component Class .....	176
Basic Components .....	182
<b>Appendix B Resources for Role Configuration .....</b>	<b>193</b>
<b>Index .....</b>	<b>1</b>

# List of Figures

Figure 2-1	Directory Editor License Agreement .....	10
Figure 2-2	Select Installation Directory Screen .....	11
Figure 2-3	Ready to Install Screen .....	11
Figure 2-4	Installation Summary Screen .....	12
Figure 2-5	Startup Properties Page .....	28
Figure 2-6	Managed Directories Page .....	32
Figure 3-1	Directory Editor Log In Page .....	36
Figure 3-2	Edit Password Page .....	37
Figure 3-3	Directory Editor Home Page .....	38
Figure 3-4	Browse Tab .....	41
Figure 3-5	Example Browse Dialog Box .....	42
Figure 3-6	Completed Parent Entry Text Box .....	42
Figure 3-7	Online Help Link .....	43
Figure 3-8	Online Help Window .....	44
Figure 4-1	Create Page .....	46
Figure 4-2	Quick Create Tab .....	47
Figure 4-3	Mike Miller's Common Attributes .....	49
Figure 4-4	Example Group Membership Tab .....	50
Figure 4-5	Example Location Information .....	51
Figure 4-6	Example Misc Attributes Tab .....	52
Figure 4-7	Example Extensions Tab .....	54
Figure 4-8	Example Mail User Tab .....	56
Figure 4-9	Example NT User Tab .....	58
Figure 4-10	Example UNIX User Tab .....	60

Figure 4-11	Create Page .....	60
Figure 4-12	Quick Create Tab .....	61
Figure 4-13	Static Group Tab .....	62
Figure 4-14	Search For Members to Add to the Group Search Page .....	63
Figure 4-15	Removing A Member from the Group .....	65
Figure 4-16	Importing Members .....	67
Figure 4-17	Group Extensions Tab .....	68
Figure 4-18	Default Dynamic Group Form Tab .....	69
Figure 4-19	Example Results Table .....	71
Figure 4-20	Quick Create Tab .....	73
Figure 4-21	Common Attributes Page .....	74
Figure 4-22	Locality Page .....	75
Figure 4-23	Quick Create Page .....	77
Figure 4-24	Quick Create Tab .....	80
Figure 4-25	Locality Common Attributes Page .....	81
Figure 4-26	Enable the Checkbox in the Browse Tree .....	83
Figure 4-27	Enable the Checkbox in the Browse Tree .....	83
Figure 4-28	Click the Object Name in the Results Table .....	83
Figure 4-29	Example Edit an Object Page .....	84
Figure 4-30	Import/Export Tab .....	85
Figure 4-31	Specifying How Your Browser Should Open the File .....	86
Figure 4-32	Enable the Checkbox in the Browse Tree .....	88
Figure 4-33	Click the Object Name in the Results Table .....	88
Figure 4-34	Example Enable/Disable Pages .....	88
Figure 4-35	Enable the Checkbox in the Browse Tree .....	89
Figure 4-36	Click the Object Name in the Results Table .....	89
Figure 4-37	Rename Page .....	90
Figure 4-38	Rename Page .....	90
Figure 4-39	Deleting Objects .....	91
Figure 5-1	Forms Tab .....	97
Figure 5-2	Default Create Form .....	99
Figure 5-3	Tree View Display (Expanded) .....	100
Figure 5-4	Initial Property View: Create Form .....	101
Figure 5-5	Property View: Editing New Object Type Element .....	102
Figure 5-6	Identifying a Default Create Forms .....	103
Figure 5-7	New Form Page .....	104
Figure 5-8	Customizing Field Properties .....	105



Figure 5-9	Move Field Tab .....	108
Figure 5-10	Move Field Menu .....	109
Figure 5-11	Creating a New Field .....	110
Figure 5-12	Creating a New Form .....	111
Figure 5-13	Adding Classes to a Form .....	112
Figure 5-14	Specify Properties for Each Node in the Tree .....	113
Figure 6-1	Search Results .....	115
Figure 6-2	Search Results .....	116
Figure 6-3	Advanced Search Tab .....	118
Figure 6-4	Selecting Object Classes .....	119
Figure 6-5	Example 1 Search Results .....	122
Figure 6-6	Example 2 Search Results .....	123
Figure 6-7	Filtered Search tab .....	124
Figure 6-8	Example Filtered Search Results .....	125
Figure 6-9	Search Configuration Page .....	126
Figure 6-10	Results Table Page .....	128
Figure 6-11	Search Types Page .....	130
Figure 7-1	Authorization Page .....	136
Figure 7-2	Role Properties Tab .....	137
Figure 7-3	Principals Tab .....	138
Figure 7-4	Accessible Resources Tab .....	139
Figure 7-5	New Role Added to the Roles Table .....	140
Figure 7-6	Click the Checkbox .....	140
Figure 7-7	Edit Role Page .....	141
Figure 7-8	Click the Checkbox .....	141
Figure 7-9	Naming Attributes Page .....	143
Figure 7-10	New Naming Attributes Page .....	144
Figure 7-11	New Object Class and Naming Attribute Added to the Table .....	145
Figure 7-12	Updated Table .....	145
Figure 7-13	Click the Checkbox .....	146
Figure 7-14	Editing the Naming Attributes .....	146
Figure 7-15	Click the Checkbox .....	147
Figure 8-1	Backup/Restore Page .....	150
Figure 8-2	Specifying How Your Browser Should Open the File .....	151
Figure 10-1	Enabling Audit Logging .....	164



# List of Tables

Table 1	Typographic Conventions . . . . .	x
Table 2	Symbol Conventions . . . . .	xi
Table 3	Default Paths and File Names . . . . .	xi
Table 4	Books in This Documentation Set . . . . .	xii
Table 4-1	Mail User Attributes . . . . .	55
Table 4-2	NT User Attributes . . . . .	57
Table 4-3	Unix User Attributes . . . . .	59
Table 5-1	Directory Editor Forms . . . . .	93
Table 5-2	Field Properties . . . . .	106
Table 6-1	Example Search Parameters . . . . .	121
Table 6-2	Example Search Parameters . . . . .	123
Table A-1	Hidden Parameter Names . . . . .	173
Table A-2	Component Properties Data Types . . . . .	175
Table A-3	Command Property Values . . . . .	181
Table B-1	Accessible Resources for Role Configuration . . . . .	193



# Preface

This publication explains how to install and configure Sun Java System Directory Editor for use in a production environment.

For the latest information about the features and functionality in this release of Directory Editor, please see the online release notes at:

<http://docs.sun.com/db/prod/sjs.diredit>

---

<b>NOTE</b>	User interfaces depicted in this document are subject to change in future versions of the product.
-------------	--

---

This Preface contains the following information:

- “Who Should Use This Book” on page viii
- “Before You Read This Book” on page viii
- “How This Book Is Organized” on page ix
- “Conventions Used in This Book” on page x
- “Books in This Documentation Set” on page xii
- “Related Documentation” on page xii
- “Accessing Sun Resources Online” on page xii
- “Contacting Sun Technical Support” on page xiii
- “Related Third-Party Web Site References” on page xiii
- “Sun Welcomes Your Comments” on page xiv

# Who Should Use This Book

This *Installation and Configuration Guide* is intended for use by administrators, systems engineers, and professional services engineers who will use Directory Editor to create and manage directory data.

You should already be familiar with

- Creating and managing directory, organization, group, user, and container data
- Configuring and operating Sun Java™ System Directory Server
- Lightweight Directory Access Protocol (LDAP)
- Java technology
- Extensible Markup Language (XML)

# Before You Read This Book

The *Sun™ Java System Directory Editor Release Notes* contain the latest information about the product — including information that may supersede instructions provided in this book. Be sure you read these Release Notes before attempting any procedures described in this book.

Because Sun Java™ System Directory Server is used as the data store in an Directory Editor deployment, you should be familiar with the documentation provided with that product. Directory Server documentation can be accessed online at

[http://docs.sun.com/coll/DirectoryServer\\_04q2](http://docs.sun.com/coll/DirectoryServer_04q2).

# How This Book Is Organized

The *Sun Java™ System Directory Editor Installation and Configuration Guide* is organized into the following chapters:

- Chapter 1, “Before You Install” provides important information to help you prepare to install and configure Directory Editor.
- Chapter 2, “Installing Directory Editor” provides instructions for installing Directory Editor on your system.
- Chapter 3, “Getting Started” provides an overview of the Directory Editor’s graphical user interface and major features.
- Chapter 4, “Creating and Editing Objects” explains how to create and edit objects in your directories.
- Chapter 5, “Working With Forms” explains how to customize the default Directory Editor forms and how to create new forms for your site.
- Chapter 6, “Searching Directories” explains how to search for objects in the directory.
- Chapter 7, “Configuring Directory Editor” explains how to configure authorization roles and create a secure site.
- Chapter 8, “Backing Up and Restoring Configurations” explains how to backup and restore your configuration data.
- Chapter 9, “Removing the Software” explains how to uninstall Directory Editor.
- Chapter 10, “Error Logging and Troubleshooting” explains how to work with Directory Editor error logs and how to troubleshoot problems you may encounter as you work.
- Appendix A, “HTML Components Used to Define Directory Editor Forms” describes the HTML components you use to customize Directory Editor forms.
- Appendix B, “Resources for Role Configuration” describes the resources that are available for configuring roles.

# Conventions Used in This Book

The tables in this section describe the conventions used in this book. The information is organized as follows:

- “Typographic Conventions” on page x
- “Symbols” on page xi
- “Mnemonics” on page xi
- “Default Paths and File Names” on page xi

## Typographic Conventions

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

**Table 1**   Typographic Conventions

Typeface	Meaning	Examples
AaBbCc123 (Monospace)	API and language elements, HTML tags, web site URLs, command names, file names, directory path names, on-screen computer output, sample code.	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files.
<b>AaBbCc123</b> (Monospace bold)	What you type, when contrasted with on-screen computer output.	% <b>su</b> Password:
<i>AaBbCc123</i> (Italic)	Book titles, new terms, words to be emphasized. A placeholder in a command or path name to be replaced with a real name or value.	Refer to the <i>Sun Java™ System Directory Editor User's Guide</i> . <i>Do not</i> save the file. The file is located in the <i>install-dir/bin</i> directory.



# Symbols

The following table describes the symbol conventions used in this book.

**Table 2** Symbol Conventions

Symbol	Description	Example	Meaning
[ ]	Contains optional command options.	ls [-1]	The -1 option is not required.
{   }	Contains a set of choices for a required command option.	-d {y n}	The -d option requires that you use either the <i>y</i> argument or the <i>n</i> argument.
-	Joins simultaneous multiple keystrokes.	Control-A	Press the Control key while you press the A key.
+	Joins consecutive multiple keystrokes.	Ctrl+A+N	Press the Control key, release it, and then press the subsequent keys.
>	Indicates menu item selection in a graphical user interface.	File > New > Templates	From the File menu, choose New. From the New submenu, choose Templates.

# Mnemonics

Directory Editor uses *mnemonics* (underlined letters) throughout the user interface to give you quicker options for performing certain tasks. You simply type the unlined letter to perform the task. Mnemonics are not case sensitive. To access them press the Alt key simultaneously.

# Default Paths and File Names

The following table describes the default paths and file names used in this book.

**Table 3** Default Paths and File Names

Term	Description
<installation_root>/<machine_name>/logs	Default path to the Directory Editor central logs (on Unix)
C:<installation_root>-<version_number>\webapps	Default installation directory for the dml.war file (on Windows)

# Books in This Documentation Set

The following table summarizes the books included in the Directory Editor documentation set.

**Table 4** Books in This Documentation Set

Book Title	Description
<i>Sun Java System Directory Editor Installation and Configuration Guide</i>	Describes how to install and configure Directory Editor for use in a production environment.
<i>Sun Java System Directory Editor Release Notes</i>	Available after the product is released. Contains last-minute information, including a description of what is new in this current release, known problems and limitations, installation notes, and how to report issues with the software or the documentation.

## Related Documentation

Because you will be working with Sun™ Java System Directory Server, you may need to refer to their product documentation. You can access the documentation from the following location:

[http://docs.sun.com/coll/DirectoryServer\\_04q2](http://docs.sun.com/coll/DirectoryServer_04q2)

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

## Accessing Sun Resources Online

For product downloads, professional services, patches and support, and additional developer information, go to the following:

- **Developer Information**  
<http://developers.sun.com/prodtech/index.html>
- **Download Center**  
<http://www.sun.com/software/download/>
- **Product Data Sheets**  
<http://www.sun.com/software/>

- **Product Documentation Online**  
<http://docs.sun.com>
- **Product Support and Status**  
<http://www.sun.com/service/support/software/>
- **Professional Services**  
<http://www.sun.com/service/sunps/sunone/index.html>
- **Sun Enterprise Services, Solaris Patches, and Support**  
<http://sunsolve.sun.com>
- **Support and Training**  
<http://www.sun.com/supporttraining/>

## Contacting Sun Technical Support

If you have technical questions about this product that are not answered in the product documentation, go to:

<http://www.sun.com/service/contacting>

## Related Third-Party Web Site References

The following third-party web sites are referenced in this publication:

- For information about configuring LDAP over SSL:  
<http://support.microsoft.com/default.aspx?scid=kb;en-us;321051>
- For information about installing the Apache Tomcat software:  
<http://jakarta.apache.org/tomcat/>

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

# Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions.

To share your comments, go to <http://docs.sun.com> and click Send Comments. In the online form, provide the document title and part number. The part number is a seven-digit or nine-digit number that can be found on the title page of the book or at the top of the document.

For example, the title of this books is *Sun<sup>TM</sup> Java System Directory Editor Installation and Configuration Guide*, and the part number is 819-1078-10.

# Before You Install

Use the information provided in this chapter to prepare for the installation of Sun™ Java System Directory Editor (Directory Editor). This information is organized as follows:

- “Supported Software Environments” on page 1
- “Recommended Directory Editor Configuration” on page 3
- “Memory Requirements” on page 4
- “Setup Task Flow” on page 4
- “Prerequisite Tasks” on page 5
- “What’s Next?” on page 5

## Supported Software Environments

This section lists the software and environments that are compatible with Directory Editor:

- “Operating Systems” on page 2
- “Application Servers” on page 2
- “Browsers” on page 3
- “Directory Servers” on page 3

---

<b>NOTE</b>	Because software product developers frequently ship new versions, updates, and fixes to their software, the information published here changes often. Review the product release notes for updates before proceeding with installation.
-------------	---

---

## Operating Systems

Directory Editor is compatible with the following operating systems:

- SPARC/x86
- Solaris 8 or later
- AIX 4.3.3 or later
- Microsoft Windows 2000 SP3 or later
- Microsoft Windows 2003
- Red Hat Linux

## Application Servers

Directory Editor supports any application server that supports Servlet 2.2 or later, but Directory Editor operates most efficiently with the following application servers:

- Sun Java™ System Application Server 8 or later
- Sun ONE Application Server 7 or later
- Apache Tomcat 4.1.x or later
- BEA WebLogic 8.1 or later
- IBM WebSphere 4 or later

---

<b>NOTE</b>	You must configure your application server's policy file to give Directory Editor permission to access certain system resources. Instructions for editing the policy file are provided in Chapter 2, the "Step 3: Install Directory Editor on an Application Server" section.
-------------	---

---

## Browsers

Directory Editor is compatible with the following browsers:

- Firefox 1.0
- Netscape 7.0 and later
- Mozilla 1.4 and later
- Microsoft Internet Explorer (IE) 5.5 and later

## Directory Servers

Using Directory Editor with Sun Java™ System Directory Server 5.1 or Sun ONE Directory Server provides you with enhanced functionality and optimal performance.

# Recommended Directory Editor Configuration

When configuring Directory Editor, the following configurations are recommended for optimum performance and support:

- Sun Solaris 9, running one of the following application servers:
  - Sun ONE Application Server 7.0
  - WebLogic 7.2
- Microsoft Windows 2000 SP4, running one of the following application servers:
  - Sun ONE Application Server 7.0
  - Apache Tomcat 5.0
  - WebLogic 7.2
  - WebLogic 8.1
  - WebSphere 5.0
- IBM AIX 4.3.3, running WebSphere 5.0
- IBM AIX 5.2, running WebSphere 5.1

This configuration list is based on input from customer configurations, support, professional services, and pre-sales.

# Memory Requirements

You can determine memory requirements and set values in your application server's JVM (Java Virtual Machine). Add the maximum and minimum heap size to the Java command line, for example:

```
java -Xmx128M -Xms128M
```

---

<b>NOTE</b>	For the best performance, set these values to the same size.
-------------	--

---

## Setup Task Flow

Your installation and configuration task flow will depend somewhat on which application server and database server you decide to use. In general, you will perform the following tasks:

- Perform the prerequisite tasks (described in the next section)
- Install the Directory Editor `dml.war` file
- Configure Directory Editor

Depending on your server types; you may combine steps, perform them in a different order, or eliminate them altogether.



# Prerequisite Tasks

Before you install the software, you must perform the following tasks:

- Install and configure a Java Development Kit (JDK or J2ESDK) version 1.4.2 or later if a JDK is not provided by your application server
- Install and configure an application server
- Install Directory Server

---

<b>NOTE</b>	For detailed information about installing and configuring Sun Java™ System Directory Server, consult the <i>Sun Java™ System Directory Server 5 2004Q2 Installation and Migration Guide</i> .
-------------	---

---

- Select a *configuration directory* and a *managed directory*
  - The Directory Editor configuration directory is a directory where you store the product's configuration information.

This directory does not have to be a configuration directory as defined by the *Sun ONE Server Console 5.x Server Management Guide*.
  - The managed directory is the directory server serving as the source of data being managed by Directory Editor.

---

<b>NOTE</b>	You can use the same directory for both the configuration and the managed directories.
-------------	--

---

## What's Next?

Continue to Chapter 2, "Installing Directory Editor" to install and configure Directory Editor.

What's Next?

# Installing Directory Editor

Use the information and procedures described in this chapter to install Directory Editor for use with different application servers. This chapter is organized as follows:

- “Before You Begin” on page 7
- “Installation Steps” on page 8
- “What’s Next?” on page 34

## Before You Begin

Before starting the Directory Editor installation process,

- Ensure that your application server or servlet container uses the Java 1.4.2 J2SDK (or later)
- Ensure you have the following information at-hand:
  - If your application server is already installed, note the installation location
  - Your login and password for the Directory Server

---

<b>NOTE</b>	You must have an account with read and write access for the directory server you will use to store the Directory Editor configuration.
-------------	--

---

# Installation Steps

Use the following steps to install Directory Editor:

1. Install an application server (if it is not already installed on your machine).
2. Install the Directory Editor software.
3. Install and deploy Directory Editor on your application server.
4. Set the Startup Properties.
5. Set the Managed Directory.
6. Log-in.

## Step 1: Install an Application Server

You must have a Java compiler and a Java Virtual Machine installed on your machine to run the Java classes that perform actions in Directory Editor. Both items are available with the Java 2 Software Developer's Kit (SDK). (The JRE packages do not provide a compiler.)

You can download the J2SDK from <http://java.sun.com> and follow the installation instructions provided with the product *or* many application servers bundle a JDK with their installation.

---

**CAUTION** Directory Editor requires access to some privileged operations that the application server may not allow by default. Edit the security policy file for your application server and grant access to certain actions, as follows:

1. Open the `server.policy` file.
2. Add the following lines to the end of the block granting access to “all remaining code”:

```
permission javax.security.auth.AuthPermission
"getLoginConfiguration";

permission javax.security.auth.AuthPermission
"setLoginConfiguration";

permission javax.security.auth.AuthPermission
"createLoginContext.SunDirectoryLogin";

permission javax.security.auth.AuthPermission
"modifyPrincipals";
```

3. Save the file.
  4. Restart the application server.
- 

---

**NOTE**

- The J2SDK version that is shipped with an application server is always preferred to any other J2SDKs installed on your server.
- You can conserve space on your machine when installing the J2SDK, if you do not install the Java documentation, examples, or header files.
- Be aware that adding `JAVA_HOME` to your list of system environment variables benefits the application server, but it may affect other applications.

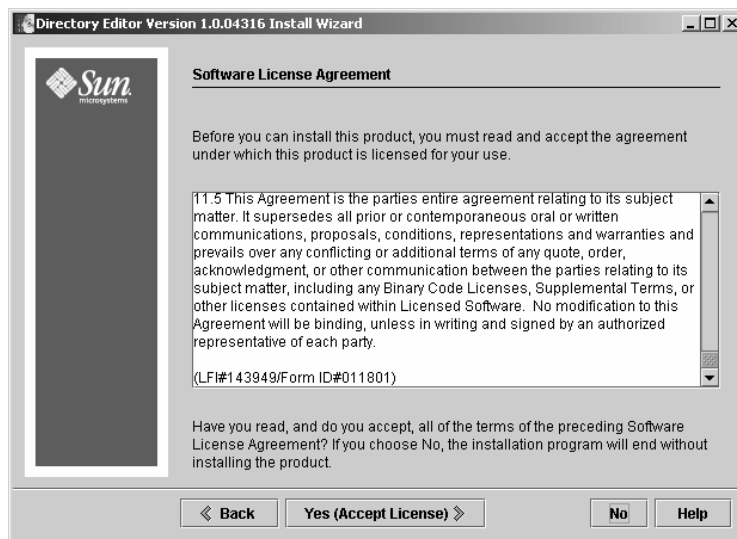
---

## Step 2: Install Directory Editor

Use the following instructions to download and install the Directory Editor `dml.war` file.

1. Download the `DE12004Q4.zip` file from the Sun website to a safe location in your deployment directory. This zip file contains the following files:
  - Readme file
  - Third-party license file
  - Directory Editor license file
  - Directory Editor installer
2. Extract these files from the `DE12004Q4.zip` file.
3. From a command window, type `java.dml` to launch the Directory Editor installer.
4. When the Welcome screen is displayed, click Next to open the Directory Editor Software License Agreement.

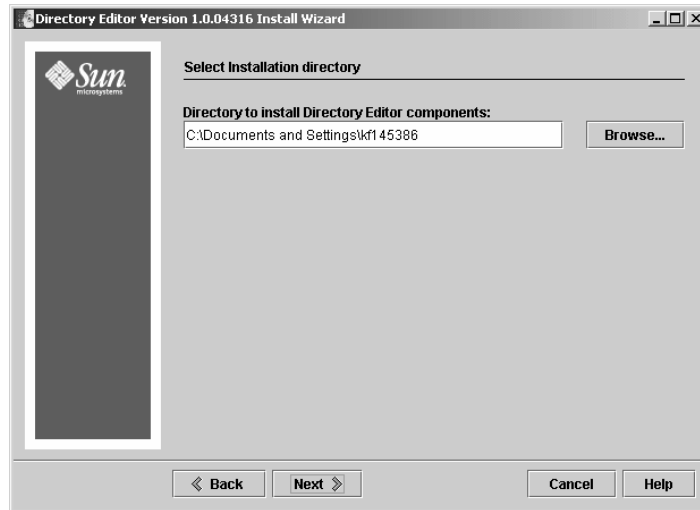
**Figure 2-1** Directory Editor License Agreement



- Click Yes (Accept License) if you agree to the terms.
- Click No to cancel the installation.

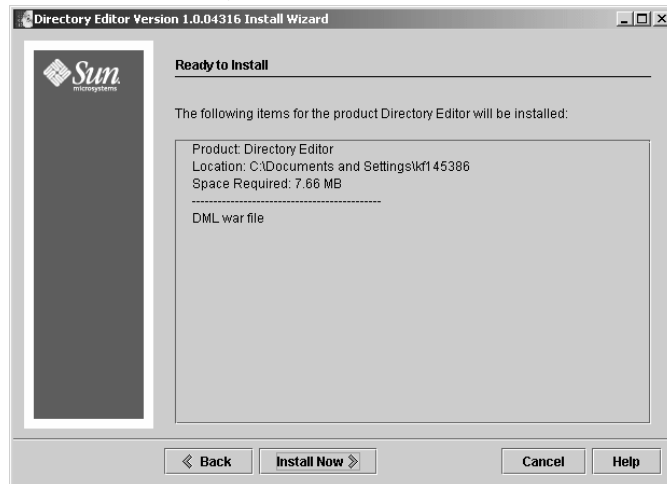
5. If you accept the license agreement, the Select Installation Directory screen displays. Enter the directory and path where you want to install the product components (or click Browse to locate and select a directory).

**Figure 2-2** Select Installation Directory Screen



6. Click Next and the Ready to Install Screen is displayed to provide information about the product.

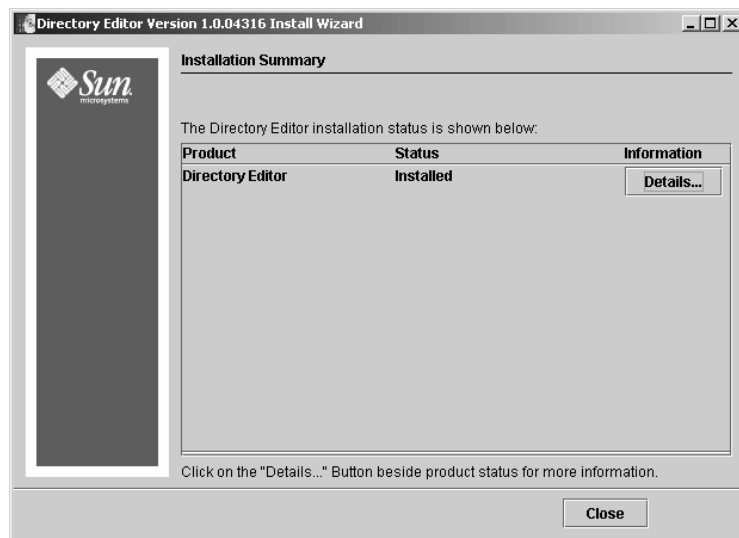
**Figure 2-3** Ready to Install Screen



7. When you're ready, click **Install Now**.

An **Installing...** screen displays briefly during the installation process, and then the **Installation Summary** screen is displayed to report the installation status.

**Figure 2-4** Installation Summary Screen



If necessary, you can click the **Details** button for additional information.

8. Click **Close** to exit the installer.

Next, you must install and deploy the `dml.war` file on your application server. Continue to "Step 3: Install Directory Editor on an Application Server" on page 13 for instructions:



## Step 3: Install Directory Editor on an Application Server

After downloading and installing the Directory Editor `dml.war` file, use the instructions provided in this section to install and deploy the software on your application server.

The information is organized as follows:

- “Installing Directory Editor on a Sun Application Server” on page 13
- “Installing Directory Editor for Tomcat 5.0.x” on page 17
- “Installing Directory Editor for WebLogic” on page 24
- “Installing Directory Editor for WebSphere 5.1” on page 26

### Installing Directory Editor on a Sun Application Server

Use the procedures described in this section to install Directory Editor for use with a Sun Java™ System Application Server or a Sun ONE Application Server.

The information is organized as follows:

- “Installing the Sun Application Server Software” on page 13
- “Deploying Directory Editor into a Sun Applications Server” on page 16

### *Installing the Sun Application Server Software*

---

<b>NOTE</b>	The following information is provided for general reference only. For detailed installation instructions, consult the installation instructions provided with your application server product or download the Sun Java System Application Server or Sun ONE Application Server product documentation from <a href="http://docs.sun.com">http://docs.sun.com</a> .
-------------	---

---

You may need to perform one or more of these general steps when installing the software:

- Use the Sun Application Server’s Typical installation.
- Specify a location for the installation directory.
- Specify an administrator name and password for Application Server administration.

### *Configuring the Sun Application Server's Policy File*

You must configure the application server's policy file to give Directory Editor permission to access the application server.

1. Define a `dml.home` variable as follows:
  - a. Launch and log into the Sun ONE Admin Console.
  - b. On the left side of the Console, click the folder name for the Application Server instance on which Directory Editor will be installed.

---

**NOTE**      The default folder name is usually *server1*.

---

- c. On the left side of the Console, select the JVM settings tab, and then select the JVM Options link.
  - d. Add the following JVM option for `dml.home`:

```
-Ddml.home=<SunONEHome>\<domain_name>\<App_Server_Instance>\
applications\j2ee-modules\<idm_1>
```

For example:

```
-Ddml.home=-Ddml.home=/opt/SUNWappserver7/domain1/server1/
applications/j2ee-modules/dml_1
```

2. Add the following line to the `<App Server Home>/domains/<Domain Name>/config/server.policy` file (see Code Example 2-1):

```
grant codeBase "file:${dml.home}/-" {
permission java.util.PropertyPermission "*", "read,write";
permission java.io.FilePermission "<<ALL FILES>>", "execute";
};
```

**Code Example 2-1** Example Sun Application Server Policy File

```

/* Copyright (c) 2002 Sun Microsystems, Inc. All rights reserved. */
/* Use is subject to license terms. */

//System Code Permissions
grant codeBase "file:${waveset.home}/-" {
    permission java.util.PropertyPermission "waveset.home", "read,write";
    permission java.security.SecurityPermission "putProviderProperty.CryptixCrypto";
    permission java.security.SecurityPermission "insertProvider.CryptixCrypto";
    permission java.util.PropertyPermission "security.provider", "read,write";
    permission java.io.FilePermission "${waveset.home}/${/*}", "read,write,execute";
    permission java.util.PropertyPermission "/*", "read,write";
    permission java.lang.RuntimePermission "accessClassInPackage.sun.io";
    permission java.net.SocketPermission "/*", "connect,resolve";
};

grant { permission java.security.AllPermission; };

grant codeBase "file:${dml.home}/-" {
    permission java.util.PropertyPermission "/*","read,write";
    permission java.io.FilePermission"<<ALL FILES>>","execute";
};

//Web Application Permissions
grant {
    permission java.lang.RuntimePermission "accessClassInPackage.sun.io";
    permission java.lang.RuntimePermission "createClassLoader";
    permission java.lang.RuntimePermission "accessDeclaredMembers";
    permission com.waveset.repository.test.testConcurrentLocking "read";
    permission java.net.SocketPermission "/*", "connect,resolve";
    permission java.io.FilePermission "/export/logs/trace.log", "read,write";
    permission java.util.PropertyPermission "trace.file", "read";
    permission java.util.PropertyPermission "trace.destination", "read";
    permission java.util.PropertyPermission "trace.enabled", "read";
};

// Core server classes get all permissions by default
grant codeBase "file:${com.sun.aas.installRoot}/lib/-" {
    permission java.security.AllPermission;
};

// iMQ classes get all permissions by default
grant codeBase "file:${com.sun.aas.imqLib}/-" {
    permission java.security.AllPermission;
};

```

```
// Web Services classes get all permissions by default
grant codeBase "file:${com.sun.aas.webServicesLib}/-" {
    permission java.security.AllPermission;
};

// Basic set of required permissions granted to all remaining code
grant {
    permission java.lang.RuntimePermission "loadLibrary.*";
    permission java.lang.RuntimePermission "queuePrintJob";
    permission java.net.SocketPermission ".*", "connect";
    permission java.io.FilePermission "<<ALL FILES>>", "read,write";
    permission java.util.PropertyPermission ".*", "read";

    permission java.lang.RuntimePermission "modifyThreadGroup";
};

// Following grant block is only required by Connectors. If Connectors
// are not in use the recommendation is to remove this grant.
grant {
    permission javax.security.auth.PrivateCredentialPermission
"javax.resource.spi.security.PasswordCredential com.sun.enterprise.security.PrincipalImpl
\"ANONYMOUS\".", "read";
};

// Following grant block is only required for Reflection. If Reflection
// is not in use the recommendation is to remove this section.
grant {
    permission java.lang.RuntimePermission "accessDeclaredMembers";
};
```

### *Deploying Directory Editor into a Sun Applications Server*

Use the following steps to deploy Directory Editor into the Sun Application Server:

1. Launch the Sun Admin Console and log in.
2. Navigate to the Web Apps folder icon in the left panel (for example, select App Server Instance->server1->Applications->Web Apps) and click the folder icon.
3. In the right panel, select the Deploy tab.
4. Enter the file path for the dml.war file, and then click OK.
5. When prompted, set both the Web Application Name and the Context Root to dml, and then click OK.

6. Open your browser and type `http://localhost:<port_number>/dml/` into the URL field. (If you used the defaults at install time and you are not using a Web Server. Also, the port number will vary.)

---

**NOTE** If you are running your web browser on a host other than the application server you may have to change or adjust the host name.

---

The Directory Editor Startup Properties page is displayed. Continue to “Step 4: Specify the Startup Properties” on page 28 for instructions.

## Installing Directory Editor for Tomcat 5.0.x

This section provides general instructions for installing an Apache Tomcat application server, and then explains two methods for installing Directory Editor on an Apache Tomcat application server. The information is organized as follows:

- “Installing Tomcat” on page 17
- “Installing Directory Editor Using Tomcat Manager” on page 22
- “Installing Directory Editor Manually” on page 23

### *Installing Tomcat*

---

**NOTE** The instructions provided in this section are provided for general reference only. For detailed installation instructions, consult the Apache website (<http://jakarta.apache.org/tomcat/>) or the reference information provided for the application server software.

---

If you are installing the application server from the Tomcat installer,

1. Download and unpack the Tomcat installation bundle.
2. Decide where to install your Tomcat installation.
3. Specify to start Tomcat as a service (*on Windows only*).
4. Select a port (default is *8080*).
5. Modify the Tomcat start-up script as follows:

- **On UNIX:** Open the `$CATALINA_HOME/bin` directory and add the following lines to the top of the `setclasspath.sh` file:

```
JAVA_HOME=<JDK location>
```

```
export JAVA_HOME
```

- **On Windows:** Open the `$CATALINA_HOME/bin` directory and add the following lines to the top of the `setclasspath.bat` file:

```
SET JAVA_HOME=<JDK location>
```

### *Configuring the Policy File*

If you will be running Tomcat with the Security Manager turned on, you must configure the application server's `catalina.policy` file (located in `$Appserver_home\conf\catalina.policy`) to give Directory Editor permission to access the application server. Add the following line to the bottom of the file (see Code Example 2-2):

```
grant codeBase "file:${catalina.home}/webapps/dml/-" {
  permission java.security.AllPermission;
};
```

#### **Code Example 2-2**     Example `catalina.policy` File

```
/ =====
// catalina.corepolicy - Security Policy Permissions for Tomcat 5
//
// This file contains a default set of security policies to be enforced (by the JVM)
// when Catalina is executed with the "-security" option. In addition to the permissions
// granted here, the following additional permissions are granted to the codebase
// specific
// to each web application:
//
// * Read access to the document root directory
//
// $Id: catalina.policy,v 1.11 2004/03/02 12:36:22 remm Exp $
// =====
```

```

// These permissions apply to javac grant codeBase "file:${java.home}/lib/-"
{permission java.security.AllPermission;
};

// These permissions apply to all shared system extensions grant codeBase
"file:${java.home}/jre/lib/ext/-" {permission java.security.AllPermission;
};

// These permissions apply to javac when ${java.home} points at $JAVA_HOME/jre grant
codeBase "file:${java.home}/../lib/-" {permission java.security.AllPermission;
};

// These permissions apply to all shared system extensions when
// ${java.home} points at $JAVA_HOME/jre grant codeBase "file:${java.home}/lib/ext/-"
{permission java.security.AllPermission;
};

// ===== SYSTEM CODE PERMISSIONS =====

// ===== CATALINA CODE PERMISSIONS =====

// These permissions apply to the launcher code grant codeBase
"file:${catalina.home}/bin/commons-launcher.jar" {permission
java.security.AllPermission;
};

// These permissions apply to the daemon code grant codeBase
"file:${catalina.home}/bin/commons-daemon.jar" {permission java.security.AllPermission;
};

// These permissions apply to the commons-logging API grant codeBase
"file:${catalina.home}/bin/commons-logging-api.jar"
{permission java.security.AllPermission;
};

// These permissions apply to the server startup code grant codeBase
"file:${catalina.home}/bin/bootstrap.jar" {permission java.security.AllPermission;
};

// These permissions apply to the JMX server grant codeBase
"file:${catalina.home}/bin/jmx.jar" {permission java.security.AllPermission;
};

// These permissions apply to the servlet API classes and those that are shared across
all // class loaders located in the "common" directory grant codeBase
"file:${catalina.home}/common/-" {permission java.security.AllPermission;
};

```

```
// These permissions apply to the container's core code, plus any additional
// libraries installed in the "server" directory
grant codeBase "file:${catalina.home}/server/-" {permission
java.security.AllPermission;
};

// ===== WEB APPLICATION PERMISSIONS =====

// These permissions are granted by default to all web applications
// In addition, a web application will be given a read FilePermission and JndiPermission
// for all files and directories in its document root.grant {
// Required for JNDI lookup of named JDBC DataSource's and
//javamail named MimePart DataSource used to send mail

permission java.util.PropertyPermission "java.home", "read";
permission java.util.PropertyPermission "java.naming.*", "read";
permission java.util.PropertyPermission "javax.sql.*", "read";

// OS Specific properties to allow read access
permission java.util.PropertyPermission "os.name", "read";
permission java.util.PropertyPermission "os.version", "read";
permission java.util.PropertyPermission "os.arch", "read";
permission java.util.PropertyPermission "file.separator", "read";
permission java.util.PropertyPermission "path.separator", "read";
permission java.util.PropertyPermission "line.separator", "read";

// JVM properties to allow read access
permission java.util.PropertyPermission "java.version", "read";
permission java.util.PropertyPermission "java.vendor", "read";
permission java.util.PropertyPermission "java.vendor.url", "read";
permission java.util.PropertyPermission "java.class.version", "read";
permission java.util.PropertyPermission "java.specification.version", "read";
permission java.util.PropertyPermission "java.specification.vendor", "read";
permission java.util.PropertyPermission "java.specification.name", "read";
```



```

permission java.util.PropertyPermission "java.vm.specification.version", "read";
permission java.util.PropertyPermission "java.vm.specification.vendor", "read";
permission java.util.PropertyPermission "java.vm.specification.name", "read";
permission java.util.PropertyPermission "java.vm.version", "read";
permission java.util.PropertyPermission "java.vm.vendor", "read";
permission java.util.PropertyPermission "java.vm.name", "read";

// Required for OpenJMX
permission java.lang.RuntimePermission "getAttribute";

// Allow read of JAXP compliant XML parser debug
permission java.util.PropertyPermission "jaxp.debug", "read";

// Precompiled JSPs need access to this package.
permission java.lang.RuntimePermission "accessClassInPackage.org.apache.jasper.runtime";
permission java.lang.RuntimePermission "accessClassInPackage.org.apache.jasper.runtime.*";
};

// You can assign additional permissions to particular web applications by
// adding additional "grant" entries here, based on the code base for that
// application, /WEB-INF/classes/, or /WEB-INF/lib/ jar files.
//

// Different permissions can be granted to JSP pages, classes loaded from
// the /WEB-INF/classes/ directory, all jar files in the /WEB-INF/lib/
// directory, or even to individual jar files in the /WEB-INF/lib/ directory.
//

// For instance, assume that the standard "examples" application
// included a JDBC driver that needed to establish a network connection to the
// corresponding database and used the scrape taglib to get the weather from
// the NOAA web server. You might create a "grant" entries like this:
//

```

```
// The permissions granted to the context root directory apply to JSP pages.
// grant codeBase "file:${catalina.home}/webapps/examples/-" {
//     permission java.net.SocketPermission "dbhost.mycompany.com:5432", "connect";
//     permission java.net.SocketPermission "*.noaa.gov:80", "connect";
// };
//

// The permissions granted to the context WEB-INF/classes directory
// grant codeBase "file:${catalina.home}/webapps/examples/WEB-INF/classes/-" {
// };
//

// The permission granted to your JDBC driver
// grant codeBase
"jar:file:${catalina.home}/webapps/examples/WEB-INF/lib/driver.jar!/-" {
//     permission java.net.SocketPermission "dbhost.mycompany.com:5432", "connect";
// };

// The permission granted to the scrape taglib
// grant codeBase "jar:file:${catalina.home}/webapps/examples/WEB-INF/lib/scrape.jar!/-" {
//     permission java.net.SocketPermission "*.noaa.gov:80", "connect";
// };
grant codeBase "file:${catalina.home}/webapps/dml/-" {permission
java.security.AllPermission;
};
```

### *Installing Directory Editor Using Tomcat Manager*

This section explains how to install Directory Editor using Tomcat Manager:

---

**NOTE** You must have a manager role in Tomcat to use this installation method. To verify your status, check the following file:  
file <Tomcat\_base\_directory>/conf/tomcat-users.xml

---

Use the following steps to install Directory Editor using the Tomcat Manager:

1. Navigate to the Tomcat bin directory and use one of the following methods to start Tomcat:
  - **On Unix:** From the command line, invoke `startup.sh`.
  - **On Windows:** Double-click `startup.bat`.

2. Open your browser and type `localhost:<port_number>/manager/html` into the URL field.
3. You will be prompted for your Tomcat Manager Application user name and password. Enter the information and press OK.

The Tomcat Manager page is displayed.

4. Scroll down until you locate the Deploy section called “War file to deploy.”
5. Click Browse to locate the Directory Editor `dml.war` file.
6. Select the `dml.war` file and then click Deploy.

### *Installing Directory Editor Manually*

Use the following steps to install Directory Editor manually:

1. Copy the Directory Editor `dml.war` file from its current location into the Tomcat webapps directory. For example,  
`C:\Tomcat\jakarta-tomcat-5.0.28\webapps\`
2. Navigate to the Tomcat bin directory and use one of the following methods to stop Tomcat:
  - **On Unix:** From the command line, invoke `shutdown.sh`.
  - **On Windows:** Double-click `shutdown.bat`.
3. From the same directory, use one of the following methods to restart Tomcat:
  - **On Unix:** From the command line, invoke `startup.sh`.
  - **On Windows:** Double-click `startup.bat`.

The Tomcat application will automatically detect the `dml.war` file, and will install the Directory Editor program.

---

<b>NOTE</b>	Be sure to wait for Tomcat to finish initializing before proceeding to the next step.
-------------	---

---

4. Open your browser and type `localhost:<port_number>/dml/` into the URL field.

The Directory Editor Startup Properties page is displayed. Continue to “Step 4: Specify the Startup Properties” on page 28 for instructions.

## Installing Directory Editor for WebLogic

Use the procedures described in this section to install Directory Editor for use with the BEA WebLogic application server. The information is organized as follows:

- “Configuring the WebLogic Software” on page 24
- “Installing Directory Editor” on page 24
- “Configuring a WebLogic 7x Server” on page 25
- “Configuring a WebLogic 8.1 SP1Server” on page 25

### *Configuring the WebLogic Software*

If necessary, install WebLogic (using that product’s installation instructions) and select the domain that will be referenced when you install Directory Editor.

### *Installing Directory Editor*

Use the following steps to install Directory Editor:

1. Copy the Directory Editor `dml.war` file from its current location into the folder where you want to install Directory Editor.

---

<b>NOTE</b>	<p>The WebLogic Web application home directory is:</p> <ul style="list-style-type: none"> <li>• <b>For Version 7x:</b>  <code>&lt;ServerHome&gt;/user_projects/&lt;DomainName&gt;/applications</code></li> <li>• <b>For Version 8.1 SP1:</b>  <code>&lt;ServerHome&gt;/user_projects/domains/&lt;DomainName&gt;/applications</code></li> </ul>
-------------	--

---

2. The Application Home panel will display the location where Directory Editor will be installed. Click Next to begin installation.
3. Navigate to the WebLogic `bin` directory and use one of the following methods to stop WebLogic:
  - **On Unix:** From the command line, invoke `shutdown.sh`.
  - **On Windows:** Double-click `shutdown.bat`.
4. From the same directory, restart WebLogic using one of the following methods:
  - **On Unix:** From the command line, invoke `startup.sh`.
  - **On Windows:** Double-click `startup.bat`.

The WebLogic application will automatically detect the `dml.war` file, and will install the Directory Editor program.

---

<b>NOTE</b>	Be sure to wait for WebLogic to finish initializing before proceeding to the next step.
-------------	---

---

Next you must configure the WebLogic server for use with Directory Editor. Continue to one of the following sections for instructions:

- “Configuring a WebLogic 7x Server” on page 25
- “Configuring a WebLogic 8.1 SP1Server” on page 25

### *Configuring a WebLogic 7x Server*

Use these steps to configure a WebLogic 7x server:

1. Start the WebLogic server:
2. Start the BEA WebLogic Administration Console.
3. In the left panel, expand deployments and then click Web Applications.  
The console displays the Web Applications panel.
4. Click Configure a new Web Application.
5. Using the links, locate the `dml` folder and select it.
6. Select the target server. To do this, select the server from the Available Servers list and move it to the Target Servers area, and then click Configure and Deploy.
7. Click Deploy to deploy Directory Editor.
8. Open your browser and type `localhost:<port_number>/dml` into the URL field. (The port number will vary.)

The Directory Editor Startup Properties page is displayed. Continue to “Step 4: Specify the Startup Properties” on page 28 for instructions.

### *Configuring a WebLogic 8.1 SP1Server*

Use these steps to configure a WebLogic 8.1 SP1 server:

1. Start the WebLogic server:
2. Start the BEA WebLogic Administration Console.

3. In the left panel, expand Deployments, and then choose Web Application Modules.

The console displays the Web Applications panel.

4. Click Deploy a new Web Application Module.
5. Using the links under applications, locate the `dm1` folder where you put the `dm1.war` file, and select it.
6. Click Target Module.
7. Review the Targets, Accessibility and Identity configuration, and make any necessary changes.
8. Click Deploy to deploy Directory Editor.
9. Open your browser and type `localhost:<port_number>/dm1` into the URL field. (The port number will vary.)

The Directory Editor Startup Properties page is displayed. Continue to “Step 4: Specify the Startup Properties” on page 28 for instructions.

## Installing Directory Editor for WebSphere 5.1

Use the following steps to install Directory Editor for use with the IBM WebSphere 5.1 (or later) application server.

1. Start the application server.
2. Start the WebSphere administration console, and then select Applications → Install New Application.
3. Add the `dm1.war` file name in the Path:Local Path field.
4. Add the path to the Context Root for the Directory Editor installation (for example, `/dm1`), and then click Next.
5. Select the Generate Default Bindings option (using the default selections for Override and Virtual Host), and then click Next.
6. Install a new applications page. If you do not want to install the application in WebSphere's default location, enter the path to a different location into the Directory to Install Application field. For example:  

```
c:\Program Files\WebSphere\AppServer\installedApps\Hostname
```
7. Be sure the Distribute Application and Use Binary Configuration options are selected.

8. Be sure the Create Mbeans for Resources and Deploy EJBs options are *not* selected.
9. Enter the name of the application in the Application Name field (the default is *dml*).
10. Selecting the Enable class reloading option is optional. Click Next.
11. To prepare for the new application's installation, make sure the panel displays a line for the current release of Directory Editor, and that it maps to the appropriate virtual host. Click Next.
12. Be sure the panel displays a line for the current release of Directory Editor, and that it maps to the appropriate server. Click Next and then click Finish.
13. Click Save to Master Configuration to save the configuration.
14. Click Save, and then wait for the page to clear.
15. Select Applications ->Enterprise Applications, and then click the application name (the name you specified in the Application Name field).
16. Be sure the Use Metadata From Binaries option is selected.
17. Select PARENT\_LAST in the Classloader Mode field.
18. Select Application in the WAR Classloader Policy field.
19. Click Apply, and then click OK.
20. From the menu bar, click Save.
21. Click Save to save the changes to the Master Configuration.
22. Stop and restart the application server.
23. Open your browser and type `localhost:<port_number>/dml` into the URL field. (The port number will vary.)

---

<b>NOTE</b>	With some platforms, there is a performance impact if you use the JCE provided with that platform. If you experience a long start-up time, see page 168 in Chapter 10, "Error Logging and Troubleshooting".
-------------	---

---

The Directory Editor Startup Properties page is displayed. Continue to "Step 4: Specify the Startup Properties" on page 28 for instructions.

## Step 4: Specify the Startup Properties

The first time you open Directory Editor, a Startup Properties page is displayed, similar to the following:

**Figure 2-5** Startup Properties Page

**Startup Properties** Save Reset

You will have to log in again after successfully saving these properties. \* Indicates required field

**Configuration Directory Server**

\* Host:

\* Port:  ☐ Secure Port

\* Bind DN:

\* Password:

\* Configuration Suffix:  Refresh

Specify the host, port, and credentials and press Refresh.

**Startup Options**

☒ Allow users to see this page during startup.

☒ Allow users to log in anonymously.

☐ Show user detailed message for failed log in attempts.

Save Reset

You will be prompted to specify these Startup properties and Managed Directory properties (described in the next section).

---

**NOTE** Directory Editor may automatically complete some of the properties fields, but you can change the information if necessary.

---



Use the following information to complete the Startup Properties page:

1. Specify the following Configuration Directory Server parameters:

- **Host:** Enter the name of the host where your configuration Directory Server is located.

---

**NOTE** The Directory Editor configuration directory is just a directory where you store the product's configuration information. This directory server does not have to be a configuration directory as defined by the *Sun ONE Server Console 5.x Server Management Guide*.

---

- **Port:** Enter the port number on which the Directory Server is listening.

---

**CAUTION** If you use SSL to connect to a server on a port that is not using SSL, will cause the connection to hang — which is a characteristic of the SSL protocol.

Similarly, using a plain socket to connect to a server's SSL socket, will also cause the connection to hang.

---

- **Bind DN:** Enter the bind distinguished name used to authenticate to Directory Server in the bind request.
- **Password:** Enter the password you use to access the configuration directory.

---

**NOTE** After completing the Configuration Directory Server Properties section, click the Refresh button and Directory Editor will automatically display the naming contexts available for storing the configuration.

---

- **Configuration Suffix:** Specify the base suffix of the naming context where the Directory Editor configuration is stored.

## 2. Specify the optional Startup Options, as follows:

- **Allow users to see this page during startup:** Enable or disable the checkbox to control whether the Startup Properties page is displayed.

---

**NOTE**

The Startup Properties page is displayed if Directory Editor cannot connect to the configuration directory when you first start the application.

For security purposes, disable this checkbox after the initial Directory Editor installation. After disabling this checkbox, you can edit the configuration directory settings by selecting Configuration -> Startup Page or by editing the `startup.properties` file in `WEB-INF/classes`.

---

- **Allow users to log in anonymously:** Enable or disable the checkbox to control whether your users can log in to Directory Editor anonymously.

If users log in anonymously, they can access Directory Editor's Home, Browse, and Search pages only. They will not have access to the Create or Configure pages.

- **Show user detailed message for failed log in attempts:** Enable or disable the checkbox to control whether the end-user will see more-detailed failed log-in messages.

For example, if the user enters an invalid password

- The following message displays if this option is disabled:

Authentication Failed: Invalid Credentials

- The following message displays if this option is enabled:

Authentication Failed: Invalid Password

---

**NOTE For Advanced Users Only:**

The following properties are configurable, but they are not represented in the Directory Editor user interface:

- **datastore.objectClass:** Specify the LDAP object class that Directory Editor can use to store configuration objects in the directory.
- **datastore.dmlIdAttribute:** Specify an attribute from the specified object class that Directory Editor can use to store configuration object IDs.
- **datastore.xmlObjectAttribute:** Specify an attribute from the specified object class that Directory Editor can use to store configuration content in XML format.
- **datastore.location:** Specify the configuration location when it is prepended with the `datastore.configSuffix`.

You can configure these properties in the `startup.properties` file located in `<application_home>\WEB-INF`; however, changing these attributes, may cause issues with the program and can make support more difficult.

---

3. When you are finished with this page, click the Save and Continue button to save the information.

---

**NOTE** Your application server must allow write access to Directory Editor's `WEB-INF/classes` directory to persist `startup.properties`.

If your application server does not allow write access, you will be allowed to continue, but Directory Editor will remain in *non-production* mode.

---

A Managed Directories page displays (similar to the following figure).

**Figure 2-6** Managed Directories Page

The screenshot shows a web-based configuration page titled "Managed Directory". At the top right are "Save" and "Cancel" buttons. Below the title, a note states: "These settings are used by this application during initialization." A legend on the right indicates that an asterisk (\*) denotes a required field.

The configuration fields are as follows:

- \* Host:
- \* Port:
- Secure Port: ☐
- \* Base Context:
- \* Manager Group:
- User search authentication: ☒ Anonymously ☐ Simple Bind
- Bind DN:
- Password:
- \* Naming Attributes: A list of attributes with "Add" and "Remove" controls.
  - 
  - 
  -

At the bottom right, there are "Save" and "Cancel" buttons.

Instructions for completing this page are provided in the next section.

## Step 5: Specify the Managed Directory Properties

Use the following information to complete the Managed Directory Properties page (Figure 2-6):

1. Specify the following parameters:
  - **Host:** Enter the name of the host where your managed directory is located.
  - **Port:** Enter the port number on which the managed directory is listening, and then enable or disable the Secure Port checkbox to control whether this directory must communicate using a secure connection.

---

**CAUTION** If you use SSL to connect to a server on a port that is not using SSL, will cause the connection to hang — which is a characteristic of the SSL protocol.

Similarly, using a plain socket to connect to a server's SSL socket, will also cause the connection to hang.

---

- **Base Context:** Enter the base context of the naming context to be managed.


For example, `ou=People,dc=example,dc=com`

---

**NOTE** You must set the base context high enough in the tree to ensure that you have access to all the information you need.

---

- **Manager Group:** Enter the name or distinguished name (DN) of an LDAP group under the base context whose members are considered Directory Editor directory managers (administrators). These directory managers will have access rights to all Directory Editor functionality. (For more information, see “Configuring Directory Editor” on page 133.)

- **User search authentication:** Enter the method by which Directory Editor will search the directory for authenticating users.
  - **Anonymously:** Enable this button if your user objects are visible to anonymous search queries. (The Bind DN and Password text fields will become inactive and you cannot type in those fields.)
  - **Simple Bind:** If your user objects are not visible to anonymous search queries, you must enable this button and provide a Bind DN and Password in the text boxes provided.
- **Bind DN:** Enter the bind distinguished name used to authenticate to the managed directory in the bind request (*not required for anonymous user search authentication*). This option is used only to enable Directory Editor to search for users during the login process.
- **Password:** Enter the password you use to access your managed directory to search for user object DNs (*not required for anonymous user search authentication*).
- **Naming Attributes:** Enter the attributes used in the directory tree. For additional attributes, click the Add button. To remove attributes, click the Rem button. 

When a user tries to log in, Directory Editor uses the Account ID field on the Log In form to search for a user object that matches one of these naming attributes exactly.

2. When you are finished, click Save to save the information and to open the Directory Editor Log In page.

## What's Next?

Continue to Chapter 3, “Getting Started” to log-in and begin customizing applications with Directory Editor.

# Getting Started

This chapter provides an overview of the Directory Editor's graphical user interface and major features. The information in this chapter is organized as follows:

- “Product Features” on page 35
- “Logging In” on page 36
- “Changing Your LDAP Password” on page 37
- “Navigating Directory Editor” on page 38
- “Browsing Directories” on page 40
- “Using the Browse Button” on page 42
- “Running in Non-Production Mode” on page 43
- “Using Online Help” on page 43

## Product Features

Directory Editor is J2EE-based web application that enables you to quickly and easily manage directory data. Directory Editor:

- Provides an easy-to-use, directory editing tool that is appropriate for administrators and end-users
- Provides ability to create/edit users, groups, containers, and more
- Supports a many concurrent users and large directory installations
- Supports extensive customizations, branding, and embedding

- Supports private data transmissions with the client browser and the directory
- Provides built-in authorization to limit visible menus to users roles

## Logging In

When the Directory Editor Log In page is displayed (see Figure 3-1), enter your credentials (user name and password) for full access.

**Figure 3-1** Directory Editor Log In Page

Sun Java™ System Directory Editor

Account Id:

Password:

☐ Log In Anonymously

**Log In**

Copyright ©2004 Sun Microsystems, Inc. All rights reserved. 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. Use is subject to license terms. This distribution may include materials developed by third parties. Sun, Sun Microsystems, the Sun logo and Java are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

---

**NOTE**

- If you enable the Log In Anonymously checkbox, you will have access to Directory Editor's Home, Browse, and Search pages only. You cannot access the Create or Configure pages.
  - After your initial log in, the log in screen (Figure 3-1) will display every time you open Directory Editor.
- 

When you log in successfully, the Directory Editor Home page is displayed. (See "Navigating Directory Editor" on page 38 for an overview of this, and other, Directory Editor pages.)



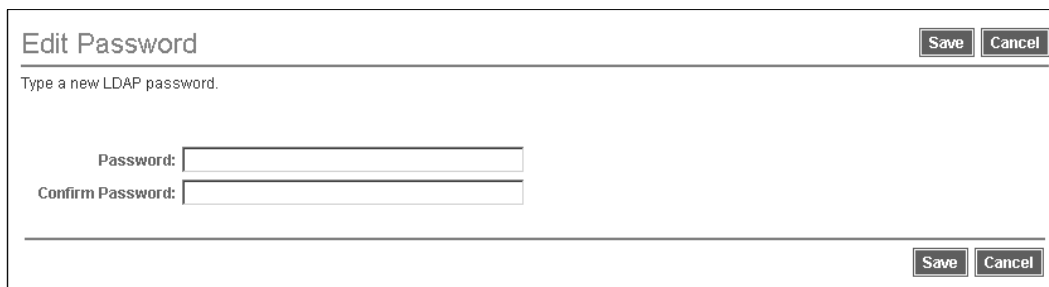
# Changing Your LDAP Password

**NOTE** If your password is about to expire, Directory Editor will display a warning message and allow you to change your password. However; if your password has already expired, you cannot log into Directory Editor until your system administrator resets your password.

To change your LDAP password, use the following steps:

1. From the Home page, click the Change My Directory Password link.
2. When the Edit Password page is displayed (Figure 3-2), type a password in the Password text box.

**Figure 3-2** Edit Password Page



**Edit Password** Save Cancel

Type a new LDAP password.

Password:

Confirm Password:

Save Cancel

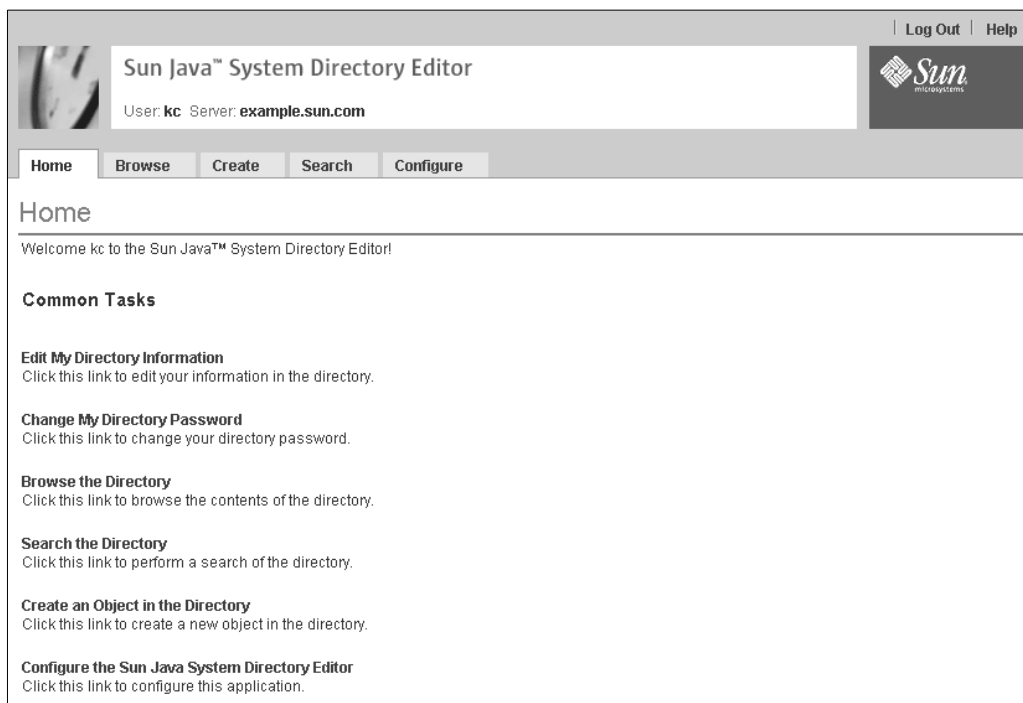
3. Retype the password in the Confirm Password text box.
4. Click the Save button when you are finished to save the password (or click Cancel to return to the Home page without saving the password).

When you log in successfully, the Directory Editor Home page is displayed.

# Navigating Directory Editor

After logging into Directory Editor as an administrator, the Home page displays as shown in the following figure:

**Figure 3-3** Directory Editor Home Page




---

**NOTE** If your page looks different than the one shown in the preceding figure, you may have specified credentials for an account that is not a member of the Manager Group you specified in “Step 5: Specify the Managed Directory Properties” on page 33.

---

The Home page contains a list of tasks commonly performed with Directory Editor. Click any of the links on this page to perform the task described, as follows:

- **Edit My Directory Information:** Click this link to open the Edit an Object page, where you can edit information in the directory. (Instructions for editing directory information are provided in “Editing Objects” on page 82.)
- **Change My Directory Password:** Click this link to open the Edit Password page, where you can change your LDAP password. (Instructions for changing your password are provided in “Changing Your LDAP Password” on page 37.)
- **Browse the Directory:** Click this link or select the Browse tab to open the Browse page where you can view the contents of the directory in tree format. (Instructions for browsing the directory are provided in “Browsing Directories” on page 40.)
- **Search the Directory:** Click this link or select the Search tab to open the Search page, where you can to locate objects in the directory. (Instructions for searching the directory are provided in Chapter 6, “Searching Directories.”)

You can execute three different types of searches from the Search page:

- **Basic Search** (*default*): Search for a user, a group, or an organizational unit.
- **Advanced Search:** Specify conditions that can be combined with and operations to search for objects.
- **Filtered Search:** Use a free-form query string to search for objects.
- **Create an Object in the Directory:** Click this link or select the Create tab open the Create page where you can create user, group, organizational unit, domain, and locality objects. (Instructions for editing objects in the directory are provided in Chapter 4, “Creating and Editing Objects.”)
- **Configure the Sun Java™ System Directory Editor** (*only available to administrative users*): Click this link or select the Configure tab to configure the Directory Editor application. You can customize forms, configure the Startup and Managed directories, configure searches, configure naming attributes, specify roles, and backup or restore configurations. (Instructions for configuring Directory Editor are provided in Chapter 7, “Configuring Directory Editor.”)






The following options (located in the upper-right corner of the application window) are also available from all Directory Editor web pages:

- **Log Out:** Click the link to log out of the Directory Editor application. (The Directory Editor log in page is displayed immediately.)
- **Help:** Click the link to open the online Help when you need assistance with the interface or a procedure. (Instructions for using the Help tool are provided in “Using Online Help” on page 43.)

## Browsing Directories

Select the Browse tab to browse the contents of the directory.

Initially, only the top-level directory objects (based on object class) are displayed in tree view, as follows,

-  OU, O, domain
-  Groups
-  Person
-  Role
-  All other


---

<b>NOTE</b>	You specify the directory context using the Managed Directory parameters stored in the configuration directory.
-------------	---

---

**Figure 3-4** Browse Tab



Click the node key  to view the contents of that folder. If there are more than 100 objects in that folder, then a link to the Search page will be displayed instead.

**NOTE**

- If you click the folder icon or directory name, the Edit page is displayed so you can edit that object. For example, if you select `cn=Directory Administrators` you can edit the object name, the availability of people in the group, add or remove members, and view statistics. Instructions for editing objects are provided in Chapter 4, “Creating and Editing Objects.”
- You can change the default limit for the number of objects displayed in a folder, by selecting `Configure > Search Options`. When the Search Configuration page is displayed, change the Number of Results value.

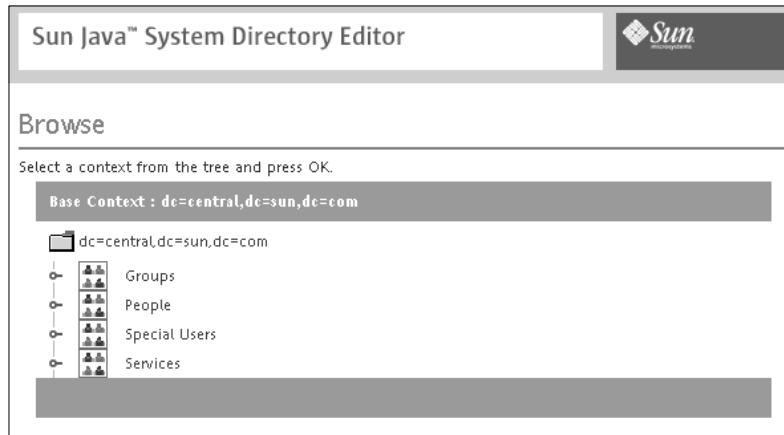
# Using the Browse Button

Many Directory Editor pages provide a Browse button to help you locate information you must provide to complete a task.

For example, when creating a new object, you must provide a parent entry (the name of the context) where you are going to add the object.

1. Click the Browse button to open a Browse dialog box, as follows:

**Figure 3-5** Example Browse Dialog Box



2. In this dialog box, navigate within the tree to locate the appropriate parent entry (or context name).
3. Click the entry to select it.

For example, you might want to add a new user entry under the People node.

4. When you have made your selection, click OK to save your selection and return to the Create page.

Directory Editor then enters the context information into the Parent Entry text box.

**Figure 3-6** Completed Parent Entry Text Box

# Running in Non-Production Mode

You must give Directory Editor permission to write to the application server file system, or Directory Editor can only run in non-production mode, which means the product cannot save its startup properties to the file system.

If you are just evaluating Directory Editor, you may want to run the product in non-production mode.

To run Directory Editor in non-production mode (evaluation mode):

1. Temporarily remove the Directory Editor permissions from your `catalina.policy` file.
2. Start Tomcat (or your application server) with the `-security` flag.
3. Re-deploy the Directory Editor application and go through the initial configuration steps described in Chapter 2, “Installing Directory Editor.”

A message prompt is displayed to warn you that Directory Editor cannot write its Startup properties, and asking if you want to continue in *non-production* mode. If you respond **Yes**, Mode: NON-PRODUCTION will display in the title bar.

4. Click the Continue button to proceed with your evaluation.

## Using Online Help

Directory Editor provides an online help tool that you can use to quickly and easily locate information about the product

To use the online help,

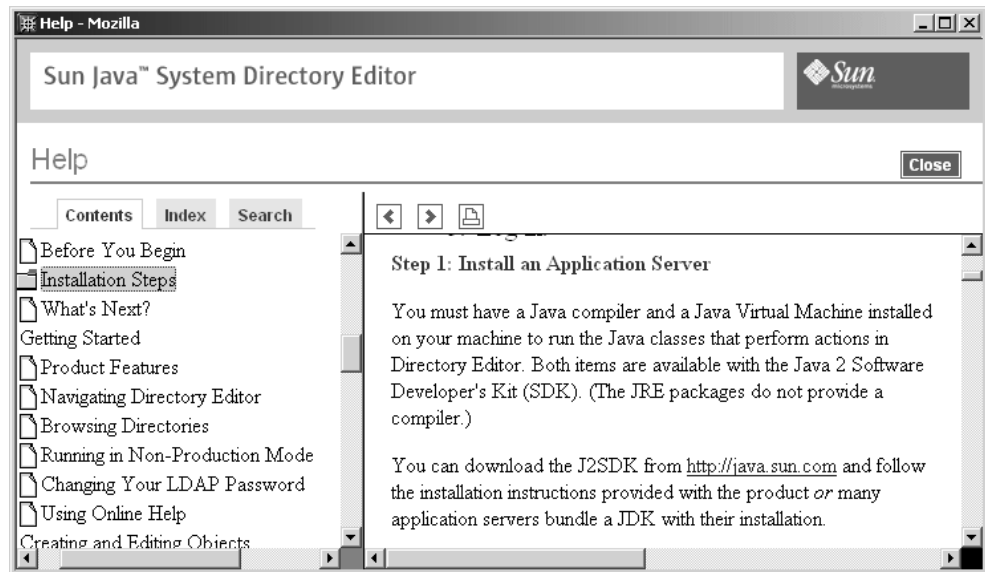
1. Click the Help link located in the upper right corner of the Directory Editor browser window.




**Figure 3-7** Online Help Link



A new, separate window is displayed as follows:

**Figure 3-8** Online Help Window



2. Use the following features to navigate in the Help window and to locate information about Directory Editor:
  - **Contents tab:** Lists all available help topics. Click a topic name to display the help content in the information pane (located on the right side of the Help window).
  - **Index tab:** Contains an index of terms and topics. Click an index entry to display the help content in the information pane.
  - **Search tab:** Enter a text string into the text box and press the Return (Enter) key to search for help content associated with that string.
  - **Back and Forward buttons:** Click the Back button  to display the previous help topic in the information pane or click the Forward button  to display the next help topic.
  - **Print button:** Click this button  to print the content displayed in the right pane. A standard Print dialog will display so you can specify print settings and select a printer.
3. Click the Close button to close the Help window.



# Creating and Editing Objects

You can use the Create tab to create new users, groups, organizational units, domains, or localities. This chapter explains how to create each object, and the information is organized as follows:

- “Creating Objects” on page 45
- “Editing Objects” on page 82

## Creating Objects

This section explains how to access the Create page and create objects using Directory Editor. The information is organized as follows:

- “Accessing the Create Page” on page 46
- “Creating Users” on page 46
- “Creating Groups” on page 60
- “Creating Organizational Units” on page 72
- “Creating Domains” on page 77
- “Creating Localities” on page 80

## Accessing the Create Page

You can access the Create Page using one of the following methods:

- Select the Create Tab
- Select the Browse tab, click a node in the Browse tree, and then click the Create button.
- Select the Search tab, execute a search from any of the Search tabs (Basic, Advanced, or Filtered), and click the Create button located just above the Results table. (See Chapter 6, “Searching Directories.”)

Each method opens the Create page shown in Figure 4-1.

**Figure 4-1** Create Page

The screenshot shows the 'Create' page of the Sun Java™ System Directory Editor. At the top, there are five tabs: 'Home', 'Browse', 'Create', 'Search', and 'Configure'. The 'Create' tab is selected. Below the tabs, the word 'Create' is displayed on the left, and 'Continue' and 'Cancel' buttons are on the right. A legend indicates that an asterisk (\*) denotes a required field. The 'New Object Type' is set to 'User' in a dropdown menu. The '\* Parent Entry' field is empty, with a 'Browse' button next to it.

## Creating Users

Use the following steps to create a new user:

1. Select the Create tab.
2. On the Create page (Figure 4-1), select User from the New Object Type menu.
3. Indicate where to add the new user by typing the parent's context information into the Parent Entry text field (or click Browse to locate the information). For example, to create the new user under People, you might enter:  
`ou=People,dc=example,dc=com`
4. When you are finished, click Continue (or press your Enter/Return key).

---

**NOTE** The Cancel button clears the Organization text box.

---

A new page is displayed, and defaults to the Quick Create tab (see Figure 4-2).

**Figure 4-2** Quick Create Tab

The screenshot shows a web form titled "Create". At the top right are "Save" and "Cancel" buttons. Below the title is a row of tabs: "Quick Create", "Common Attributes", "Group Membership", "Location Info", "Misc Attributes", and "Extensions". The "Quick Create" tab is selected. Inside this tab, there is a legend: "\* Indicates required field". Below the legend, under the heading "Common Attributes", are three text input fields, each preceded by an asterisk: "\* First Name:", "\* Last Name:", and "\* Common Name:". To the right of the "Common Name" field is a button with a plus sign and the word "Add".

---

**NOTE**

- The Quick Create tab contains all of the attributes you must provide to create the object. For example, to create an User object, you must enter a first name, last name, and at least one common name for that user.
  - The other tabs on this page enable you to specify additional, but optional, information about the new user.
- 

**5.** You can proceed in one of two ways:

- Enter the required information in the text boxes provided on the Quick Create tab to create the new user. Click Save to apply your changes.
- Select one or more of the tabs on the Create page to provide additional, optional information about the new user.

The following sections provide instructions for completing each tab on the Create page:

- “Completing the Common Attributes Tab” on page 48
- “Completing the Group Membership Tab” on page 50
- “Completing the Location Info Tab” on page 51
- “Completing the Misc Attributes Tab” on page 52
- “Completing the Extensions Tab” on page 53

6. When you are finished entering the new information on the Create page (being sure to provide all *required* information), click Save to add the new user to the directory.

## Completing the Common Attributes Tab

To provide general information about the new user, use the following steps:

1. Select the Common Attributes tab and complete one or more of the following text boxes:

---

### NOTE

- The program will automatically insert the First, Last, and Common Names if you entered them on the Quick Create tab (page 47).
  - Use the Add and Rem buttons located to the right of some attributes on this page to enter multiple values for those attributes or to remove values. For example, you might want to enter multiple telephone numbers (work, mobile, and home).
- 

- **User Name:** Enter the user name (uid).
- **First Name** (*required*): Enter the user's first name.
- **Last Name** (*required*): Enter the user's last name.
- **Common Name:** Enter a common name if it is different from the user's first name. For example, Kimberly Jones may prefer being called "Kim Jones."
- **Password:** Type a password for the new user.
- **Confirm Password:** Re-type the password to confirm it.
- **Email:** Enter the user's email address.
- **Telephone Number:** Enter the user's telephone number.
- **Mobile Phone:** Enter one or more mobile telephone numbers.
- **Pager:** Enter one or more pager numbers.
- **Facsimile Telephone Number:** Enter one or more fax numbers.
- **Department #:** Enter the user's department number.

- **Employee #:** Enter the user's employee number.
- **Employee Type:** Enter information about the user's employment status (such as salaried, contract, part-time, and so forth).

For example, if you are creating a new user named *Mike Miller*, your page might look something like the one in the following figure.

**Figure 4-3** Mike Miller's Common Attributes

**Create** Save Cancel

**Quick Create** **Common Attributes** **Group Membership** **Location Info** **Misc Attributes** **Extensions**

\* Indicates required field

User Name:

\* First Name:

\* Last Name:

\* Common Name:  ⊕ Add

Password:

Confirm Password:

Email:  ⊕ Add

Telephone Number:  ⊕ Add ⊖ Rem  
 ⊕ Add ⊖ Rem

Mobile Phone:  ⊕ Add

Pager:  ⊕ Add

Facsimile Telephone Number:  ⊕ Add

Department #:

Employee #:

Employee Type:

2. When you are finished with this tab, you can
  - Select another tab to provide additional information.
  - Click Save to save the new information to the directory.
  - Click Cancel to clear the Common Attributes text boxes.

## Completing the Group Membership Tab





To assign a new user to specific groups, use the following steps:

1. Select the Group Membership tab.
2. Use the Groups selection tool as follows:

---

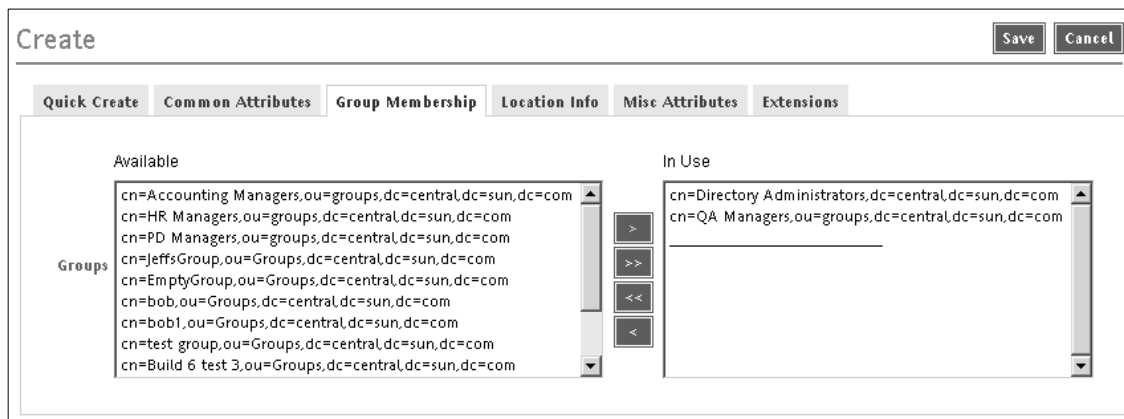
**TIP** Press the Shift key on your keyboard, and then click on group names to select multiple names in either list.

---

- To add to one or more groups to the In Use list, select the group names in the Available list and click the  button.
- To move all available groups to the In Use list, click the  button.
- To move all groups in the In Use list back to the Available list, click the  button.
- To move one or more groups back to the Available list, select the group names in the In Use list and click the  button.

For example, to assign Mike Miller to the Directory Administrators group and the QA Managers group, select those two groups from the Available list and add them to the In Use List, as shown in the following figure:

**Figure 4-4** Example Group Membership Tab



The screenshot shows the 'Create' dialog box with the 'Group Membership' tab selected. The 'Available' list on the left contains the following groups:

- cn=Accounting Managers,ou=groups,dc=central,dc=sun,dc=com
- cn=HR Managers,ou=groups,dc=central,dc=sun,dc=com
- cn=PD Managers,ou=groups,dc=central,dc=sun,dc=com
- cn=JeffsGroup,ou=Groups,dc=central,dc=sun,dc=com
- cn=EmptyGroup,ou=Groups,dc=central,dc=sun,dc=com
- cn=bob,ou=Groups,dc=central,dc=sun,dc=com
- cn=bob1,ou=Groups,dc=central,dc=sun,dc=com
- cn=test group,ou=Groups,dc=central,dc=sun,dc=com
- cn=Build 6 test 3,ou=Groups,dc=central,dc=sun,dc=com

The 'In Use' list on the right contains the following groups:

- cn=Directory Administrators,dc=central,dc=sun,dc=com
- cn=QA Managers,ou=groups,dc=central,dc=sun,dc=com

Buttons between the lists include '>', '>>', '<<', and '<'.

3. When you are finished with this tab, you can
  - Select another tab to provide additional information.
  - Click Save to save the new information to the directory.
  - Click Cancel to clear your changes and return to the initial Create page.

## Completing the Location Info Tab

To specify the new user's physical location in the company, use the following steps:

1. Select the Location Info tab.
2. Type the following information in the text boxes provided:
  - **Postal Address:** Enter the user's mailing address.
  - **Mail stop:** Enter the user's company mail stop.
  - **Postal Code:** Enter the user's mailing address postal code (zip code).

For example, you might enter the following information for Mike Miller:

**Figure 4-5** Example Location Information

Postal Address:	111 A Street, Suite 1, Anywhere, CA
Mailstop:	B1111
Postal Code:	11111-0000

3. When you are finished with this tab, you can
  - Select another tab to provide additional information.
  - Click Save to save the new information to the directory.
  - Click Cancel to clear the text boxes.

## Completing the Misc Attributes Tab

To provide additional information about the new user, use the following steps:

1. Select the Misc Attributes tab.
2. Type the following information in the text boxes provided:
  - **Photo URL:** Enter a URL location where the user's photograph can be viewed.
  - **Home Phone:** Enter one or more home telephone numbers.
  - **Home Address:** Enter the user's home address.
  - **Auto License:** Enter the user's automobile license number.

For example, you might enter the following information for Mike Miller:

**Figure 4-6** Example Misc Attributes Tab

The screenshot shows a 'Create' dialog box with the 'Misc Attributes' tab selected. The fields and their values are:

- Photo URL:
- Home Phone:  ⊕ Add
- Home Address:
- Auto License #:  ⊕ Add

### NOTE

If the new user has multiple home phones or cars, click the Add button to create new, blank text boxes so you can enter the additional numbers. Repeat as many times as necessary.

If you add new fields, Rem buttons will also display next to each field (as shown in the following figure). Click the Rem button to remove extra fields.

This close-up shows the 'Home Phone' label followed by two text input boxes. Each box has a '⊕ Add' button to its right, and the second box also has a '⊖ Rem' button to its right.



3. When you are finished with this tab, you can
  - Select another tab to provide additional information.
  - Click Save to save the new information to the directory.
  - Click Cancel to clear the text boxes.

## Completing the Extensions Tab

Use the Extensions tab to specify additional attributes to be made available for the new user.

---

**NOTE** The `inetOrgPerson`, `organizationalPerson`, `person`, and `top` extensions are in use by default.

---





To specify extensions, use the following steps:

1. Select the Extensions tab.
2. Use the Extensions selection tool as follows:

---

**TIP** Press the Shift key on your keyboard, and then click on group names to select multiple names in either list.

---

- To add to one or more extensions to the In Use list, select the extension name(s) in the Available list and click the  button.
- To move all available extensions to the In Use list, click the  button.
- To move all extensions in the In Use list back to the Available list, click the  button.
- To move one or more extensions back to the Available list, select the extension name(s) in the In Use list and click the  button.

For example, you might want to select Mail User:

**Figure 4-7** Example Extensions Tab




---

**NOTE** When you add Mail User, NT User, and/or UNIX User extensions to the In Use List, Directory Editor adds tabs to the Create page so you can provide additional information related to those extensions. See the following sections for instructions:

---

3. When you are finished with this tab, you can
  - Select another tab to provide additional information.
  - Click Save to save the new information to the directory.
  - Click Cancel to clear the text boxes.

## Completing the Mail User Tab

Use the Mail User tab to provide detailed information about the new user's Mail account, as follows:

- 
- NOTE**
- The program will automatically insert `cn`, `mail`, `uid`, and `userPassword` values based on the information you entered on the Quick Create tab (page 4-35), and on the Common Attributes tab (page 4-36).
  - For detailed information about specifying the attributes on this page, consult the *Sun Java™ System Directory Server 5 2004Q2 Administration Reference*.
- 

1. Select the Mail User tab.
2. Enter the following user information:

**Table 4-1** Mail User Attributes

Attribute Name	Description
<code>cn</code>	User's common name
<code>mail</code>	User's primary email address
<code>mailAlternateAddress</code>	Alternate email addresses
<code>mailHost</code>	Fully qualified DNS hostname of the user's email host
<code>mailRoutingAddress</code>	Forwarding email address for the user
<code>mailAccessDomain</code>	One or more domains, server names, or IP address the user can use to log on via POP or IMAP
<code>mailAutoReplyMode</code>	Auto-reply mode (For example, <i>vacation</i> )
<code>mailAutoReplyText</code>	An automatic email reply message
<code>mailDeliveryOption</code>	How the user's email should be delivered
<code>mailForwardingAddress</code>	Forwarding email address
<code>mailMessageStore</code>	Absolute path to the email host directory that contains the message store where the user's mailbox resides
<code>mailProgramDeliveryInfo</code>	One or more commands to specify programmed mail delivery for the user
<code>mailQuota</code>	Value to indicate the maximum disk space (in bytes) available to the user's mailbox
<code>multiLineDescription</code>	Description of the user

---

**Table 4-1** Mail User Attributes *(Continued)*

<b>uid</b>	User's user ID
<b>userPassword</b>	Temporary email password
<b>userPassword Confirmation</b>	Confirm password.

**NOTE** To add multiple email addresses, click the Add button located next to the mail text field. To remove an email address, click the Rem button.

For example, you might enter the following information for Mike Miller:

**Figure 4-8** Example Mail User Tab

Quick CreateCommon AttributesGroup MembershipLocation InfoMisc AttributesExtensionsMail User

cnMichael Miller⊕ Add

mailmmiller@example.com⊕ Add

mailAlternateAddressmm11111@newaddress.net

mailHostlorre.example.com

mailRoutingAddress

mailAccessDomainexample.com

mailAutoReplyModevacation

mailAutoReplyTextOn vacation until Jan. 5, 2005.\$See you next year

mailDeliveryOptionmailbox

mailForwardingAddress

mailMessageStore

mailProgramDeliveryInfo

mailQuota5000000

multiLineDescription

uidmmiller

userPassword\*\*\*\*\*

userPassword Confirmation\*\*\*\*\*

- 
- NOTE**
- If the new user has multiple email addresses, click the Add button to create new blank fields so you can enter the additional addresses. Repeat as many times as necessary.
  - When you add new fields, Rem buttons are also displayed next to each field. Click the Rem button to remove extra fields.
- 

3. When you are finished with this tab, you can
  - Select another tab to provide additional information.
  - Click Save to save the new user information to the directory.
  - Click Cancel to clear the text boxes.

### *Completing the Default NT User Tab*

Use the Default NT User tab to define entries for the new user that describe the user's Windows NT account.

To specify these entries, use the following steps:

1. Select the Default NT User tab.
2. Enter the following user information:

- 
- NOTE**
- NT User Id is the only required attribute on this tab.
  - For detailed information about specifying the attributes on this page, consult the *Sun Java™ System Directory Server 5 2004Q2 Administration Reference*.
- 

**Table 4-2** NT User Attributes

Attribute Name	Description
NT User ID	User's NT domain name
Comment	Description about user's NT account
User Workstation List	NT workstations user can log into within your NT domain
Home Drive	Drive letter assigned to user's home directory
Home Directory	User's NT home directory

---

**Table 4-2** NT User Attributes *(Continued)*

Server	Server to which user's NT logon requests are sent
User Profile Path	Path to user's Windows NT profile
Logon Script	Path to user's NT login script

For example, you might enter the following information for Mike Miller:

**Figure 4-9** Example NT User Tab

Create

SaveCancel

Quick CreateCommon AttributesGroup MembershipLocation InfoMisc AttributesExtensionsDefault NT User Form

\* Indicates required field

\* NT User ID:workgroup:mmiller

Comment:QA manager for Bogart 1.0 project

User Workstation List:bacall

Home Drive:c:

Home Directory:c:\home\example\mmiller

Server:\\bacall

User Profile Path:

Logon Script:c:\home\example\mmiller\lscript.bat

3. When you are finished with this tab, you can
- Select another tab to provide additional information.

Click Save to save the new user information to the directory.

Click Cancel to clear the text boxes.

### Completing the UNIX User Tab

Use the UNIX User tab to provide detailed information about the new user's UNIX account, as follows:

- 
- NOTE**
- The program automatically inserts values you entered on the Quick Create tab (page 4-35) and the temporary user password you entered on the Common Attributes tab (page 4-36).
  - For detailed information about specifying the attributes on this page, consult the *Sun Java™ System Directory Server 5 2004Q2 Administration Reference*.
- 

**Table 4-3** Unix User Attributes

Attribute Name	Description
<b>cn</b>	User's common name
<b>uid</b>	User's user ID
<b>uidNumber</b>	Value that uniquely identifies the user in your administrative domain
<b>gidNumber</b>	Value that uniquely identifies the user's group in your administrative domain
<b>homeDirectory</b>	Absolute path to user's home directory
<b>userPassword</b>	Temporary password for new user
<b>userPassword Confirmation</b>	Confirm password
<b>loginShell</b>	Path to user's login shell
<b>gecos</b>	GECOS field (or common name)
<b>description</b>	Information about user, such as work assignment

---

For example, you might enter the following information for Mike Miller:

**Figure 4-10** Example UNIX User Tab

Common AttributesGroup MembershipLocation InfoMisc AttributesExtensionsUNIX User

\* Indicates required field

\* cnMichael Miller⊕ Add

\* uidmmiller

\* uidNumber658

\* gidNumber650

\* homeDirectory/home/mmiller111

userPassword\*\*\*\*\*

userPassword Confirmation\*\*\*\*\*

loginShell

gecos

descriptionQA Manager for Bogart 1.0 project

## Creating Groups

Use the following steps to create a new group object in the directory:

1. From the Create tab, select Group from the New Object Type menu.
2. Indicate where to add the new group by typing the parent's context information into the Parent Entry text field (or click Browse to locate the information). For example, you might enter:

ou=Groups,dc=example,dc=sun,dc=com

**Figure 4-11** Create Page

CreateContinueCancel

\* Indicates required field

New Object Type:Group

\* Parent Entry:dc=central,dc=sun,dc=comBrowse



3. When you are finished, click Continue.

---

**NOTE** The Cancel button clears the Organization text box.

---

The Quick Create page is displayed (see Figure 4-12).

**Figure 4-12** Quick Create Tab

Quick Create Static Group Import/Export Extensions

\* Indicates required field

Static Group

\* Name: Contractors

- 
- NOTE**
- By default, you will be creating a static group object (an object with the `groupOfNames` or `groupOfUniqueNames` object class). However, you can also create dynamic group objects (a group that determines membership at any given moment using a set of LDAP search filters). For instructions, see “Completing the Extensions Tab” on page 68.
  - The Quick Create tab contains all of the attributes you must provide to create the selected object type. For example, to create a Group, you must enter a Static Group name.
  - The other tabs on this page enable you to specify additional, but optional, information about the object.
- 

4. You can proceed in one of two ways:
  - Type a name in the Name text box provided on the Quick Create tab to create the new Group. Click Save.

Directory Editor saves the new group name. Now, if you click the Groups node on the Browse page, the new group name is displayed in the tree.



- Select one or more of the tabs on the Create page to provide additional, optional information about the new Group.

The following sections provide instructions for completing each tab on the Create page:

- “Completing the Static Group Tab” on page 62
  - “Completing the Import Tab” on page 67
  - “Completing the Extensions Tab” on page 68
5. When you are finished entering the new information on the Create page (being sure to provide all *required* information), click Save to add the new group to the directory.

## Completing the Static Group Tab

You can use the Static Group tab to perform the following tasks:

- Add members to the group
- Remove members from the group (see page 64)
- Show members in the group (see page 66)

### *Adding Members to the Group*

To perform any of these tasks, use the following steps:

1. Select the Static Group tab.

**Figure 4-13** Static Group Tab

The screenshot shows the 'Static Group' tab selected. At the top are four tabs: 'Quick Create', 'Static Group' (active), 'Import/Export', and 'Extensions'. Below the tabs is a form area. In the top right of the form area, it says '\* Indicates required field'. On the left, there is a label '\* Name:' followed by a text box containing the word 'Contractors'. Below this is a label 'Number of Members:' followed by a text box containing the number '0'. At the bottom of the form area are three buttons: 'Add Members', 'Remove Members', and 'Show Members'.

2. If you specified a group name on the Quick Create tab, Directory Editor automatically completes the Name text box.

If you have not yet provided a name (or want to change the current name) enter a name in the text box.

3. Click the Add Members button and a Search page is displayed.

**Figure 4-14** Search For Members to Add to the Group Search Page

4. You can use the Basic Search, Advanced Search, or Filtered Search tabs to search the directory for members to add to this new group.

---

**NOTE** Detailed instructions for using these tabs are provided in Chapter 6, “Searching Directories.”

---

To illustrate the procedure using the Basic Search tab, use one of the following methods:

- **To search for a particular member:** Type the member’s name into the Enter the name of a User, Group, or Organizational Unit text box, use the Search Type menu to select the object type, and then click Search.

For example, enter Jones in the text box and select User (or All). Directory Editor will return all members named Jones.

- **To search for all the members:** Leave the text box blank, use the Search Type menu to select the object type, and then click Search.

For example, if you do not know the name of the member to search for, leave the text box empty and select User (or All). Directory Editor will return all User members.

5. When the Results table displays, enable the checkbox(es) next to the member(s) you want added to the new group.

6. Click the Add Selected To Group button.

Directory Editor will add the selected member(s) to the new group and return you to the Static Group tab. Note that now, the Number of Members value has been updated to reflect the number of members you added to the group.

7. You can proceed in several ways:

- If you are finished adding members to the group (and you have no other tasks to perform on this page), click Save to save your changes.

---

<b>NOTE</b>	Directory Editor does not save the new members to the directory until you click the Save button on the Static Group page.
-------------	---

---

Directory Editor returns you to the Browse page so you can verify that the new group was added with the specified members.

- Click Add Members to continue adding members to the new group.
- Click Remove Members and continue to “Removing Members from the Group” on page 64 to remove members from the group.
- Click Show Members and continue to “Showing Members in the Group” on page 66 to view the existing members of the group.
- Click the Import/Export tab and continue to “Completing the Import Tab” on page 67 to import or export members to your group.
- Click the Extensions tab and continue to “Completing the Extensions Tab” on page 68 to create extensions for your group.

### *Removing Members from the Group*

To remove members from a Group:

1. Select the Static Group tab.
2. Click the Remove Members button.
3. When the Search For Members to Remove From Group search page is displayed, use the Basic Search, Advanced Search, or Filtered Search tabs to locate the member(s) to delete.

---

<b>NOTE</b>	Detailed instructions for using these tabs are provided in Chapter 6, “Searching Directories.”
-------------	--

---

To illustrate the procedure using the Basic Search tab, use one of the following methods:

- **To search for a particular member:** Type the member's name into the text box, use the Search Type menu to specify the object type, and then click Search.

For example, type Jones in the text box and select User (or All). Directory Editor will return all members named Jones.

- **To search for all the members:** Leave the text box blank, use the Search Type menu to specify the object type, and then click Search.

For example, to see a list of all members in the group, leave the text box empty and select User (or All). Directory Editor will return all User members.

4. When the Results table displays, enable the checkbox(es) next to the member(s) you want to remove from the group's membership list, and then click the Remove Selected From Group button.

**Figure 4-15** Removing A Member from the Group

Search For Members to Remove from Group

Basic Search | Advanced Search | Filtered Search

Enter the name of a User, Group, or Organizational Unit to search for:

Jones Search type User

Search Cancel

Remove Selected From Group

Results		
<input type="checkbox"/>	Name	
<input checked="" type="checkbox"/>	Bob Jones	Jones

Directory Editor removes the selected member(s) from the group and returns you to the Static Group tab. Note that the Number of Members value has been updated to reflect the number of members currently in the group.

**NOTE**

- Directory Editor does not actually remove the members from the directory until you click the Save button on the Static Group page.
- When you use the Remove Members feature to remove members in a group, it is possible to retrieve results for entries that do not really exist in the directory. This situation occurs if the `member` or `uniqueMember` attribute of `groupOfNames` or `groupOfUniqueNames` contains values that are not valid DNs.
- A checkbox is displayed for invalid DNs. In addition, to denote a row with an invalid DN, Directory Editor displays the member name in red with an error icon.

5. You can proceed in several ways:

- If you are finished removing members (and you have no other tasks to perform on this page), click Save to save your changes. Directory Editor returns you to the Browse page so you can verify your changes.
- Click Add Members and return to “Adding Members to the Group” on page 62 to add members to the group.
- Click Show Members and continue to “Showing Members in the Group” on page 66 to view the existing members of the group.
- Click the Import/Export tab and continue to “Completing the Import Tab” on page 67 to import or export members to your group.
- Click the Extensions tab and continue to “Completing the Extensions Tab” on page 68 to create extensions for your group.

### *Showing Members in the Group*

After creating and saving a new group, you can use the Show Members button to search for members within that group.

To view a list of current group members, use the following steps:

1. Click the Show Members button.

The Search For Members in Group page displays with the group members listed in a Results table located at the bottom of the page.

2. When you are finished, click OK to return to the Static Group page.

**NOTE**

When you use the Show Members feature to search for group membership, it is possible to retrieve results for entries that do not really exist in the directory. This situation occurs if the `member` or `uniqueMember` attribute of `groupOfNames` or `groupOfUniqueNames` contains values that are not valid DNs.

To denote a row with an invalid DN, Directory Editor displays the member name in red with an error icon.

## Completing the Import Tab

When you are creating a new group, Directory Editor allows you to import group members using the Import/Export tab on the Group page.

**Figure 4-16** Importing Members

The screenshot shows the 'Create' dialog box with the 'Import/Export' tab selected. The 'Import' section is active, showing instructions to import group members from a file. A text field labeled 'File:' is followed by a 'Browse...' button. Below this, there are two checkboxes: 'Import members not found in directory?' and 'Replace existing members?'. At the bottom left of the dialog is an 'Import Members' button. The top right of the dialog has 'Save' and 'Cancel' buttons.

When you import group members, Directory Editor expects a newline-delimited text file with a member DN on each line. (Directory Editor ignores empty lines and lines starting with the pound sign (#).)

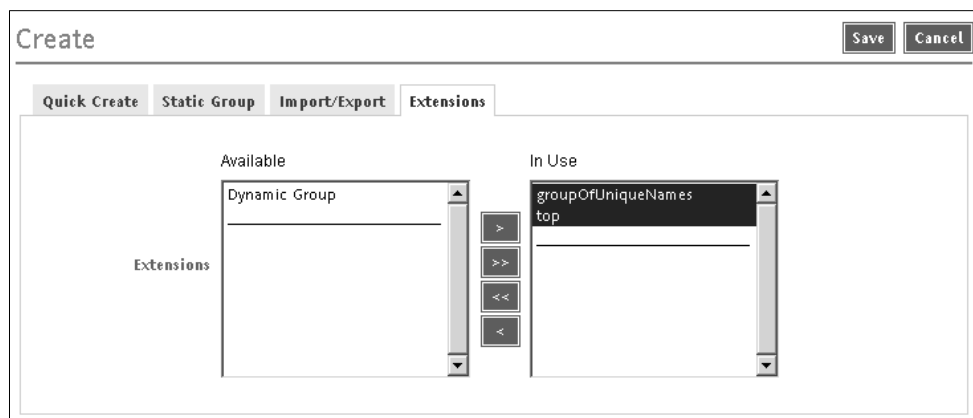
To import members from a file:

1. Type the file name into the text box provided or click Browse to open a File Upload dialog box so you can locate and select the file name.
2. Enable one of the following checkboxes:
  - **To import members** (even if they are not found in the Directory Server), enable the Import members not found in directory? check box.
  - **To replace existing group members** with members in the file (instead of adding members in the file to the existing group members), enable the Replace existing members? check box. This feature is available for both group creates and edits. Directory Editor does not save import changes to the directory until you save the group.
3. When you are ready, click Import Members to import the file.
4. Click Save to save the new group members to your directory.

## Completing the Extensions Tab




You can use the Extensions tab to specify the optional Dynamic Group extension (a group that determines membership at any given moment using a set of LDAP search filters) to the In Use list.

**Figure 4-17** Group Extensions Tab





To specify the Dynamic Group extension, use the following steps:

1. Select the Extensions tab.
2. Use the Extensions selection tool as follows:
  - To add the extension to the In Use list, select the extension name(s) in the Available list and click the  button.
  - To move extensions in the In Use list back to the Available list, click the  button.
  - To move one or more extensions back to the Available list, select the extension name(s) in the In Use list and click the  button.
3. If you add the Dynamic Group extension, Directory Editor adds the Default Dynamic Group Form tab to the Create Page.

**Figure 4-18** Default Dynamic Group Form Tab



Default Dynamic Group Form   Static Group   Import/Export   Extensions

\* Indicates required field

\* Name:

Filters

You can use this tab to add, edit, or remove LDAP search filters.

A dynamic group uses a set of LDAP search filters to determine membership in the group at any given moment. A dynamic group object consists of the `groupOfURLs` object class, a name (cn), and a multi-valued attribute (`memberURL`) containing all of the LDAP URLs that specify the search criteria.

These LDAP URLs contain information about which directory server to search, the scope of the search, the base context of the search, which attributes to return for the search, and the search filter.

For example, you might specify a dynamic group called *Contractors*, where the membership is determined by the `employeeType` equaling *Contractors*. For this example, the LDAP URL might be:

```
ldap://example.sun.com:389/ou=People,dc=sun,dc=com??sub?(
employeeType=Contractors)
```

Dynamic groups are often used to control access to applications and other data in the directory.

Instructions for adding, editing, and removing filters are provided in the following sections:

- “Adding Filters” on page 70
  - “Editing Filters” on page 72
  - “Removing Filters” on page 72
4. When you are finished with the Extensions tab, you can
- Select another tab to provide additional information.
  - Click Save to save the new information to the directory.
  - Click Cancel to clear the text boxes.

### *Adding Filters*

To add a filter for the dynamic group,

1. Select the Default Dynamic Group Form tab.
2. Click Add Filter and use the Advanced Search or Filtered Search tabs to configure a search filter.

---

<b>NOTE</b>	Detailed instructions for using the Advanced Search and Filtered Search tabs are provided in Chapter 6, “Searching Directories.”
-------------	--

---

For example, to add a filter for the Contractors group, you might specify the following parameters on the Advanced Search tab:

- **Select Type:** User
  - **Base Context:** ou=People,dc=example,dc=sun,dc=com
  - **Search Conditions:**
    - **Attribute:** employeeType
    - **Condition:** contains
    - **Value:** contractor
3. When you are finished specifying the filter criteria, you can click the Test button to verify that your filter works as planned.

The results of your filter will display in a Results table. For example:

**Figure 4-19** Example Results Table

The screenshot shows the 'Advanced Search' tab in a web application. The 'Search type' is set to 'User'. The 'Base Context' is 'ou=People,dc=central,dc=sun,d' with a 'Browse' button. The 'Search Conditions' section contains a table with one row: Attribute 'employeeType', Condition 'contains', and Value 'Contractor'. Below this are buttons for 'Remove Selected Attribute(s)', 'Add Attribute', 'Test', 'Save', and 'Cancel'.

Below the search interface is a 'Results' table:

<input type="checkbox"/>	Name	First Name	Last Name
<input type="checkbox"/>	K. C. Jones		Jones
<input type="checkbox"/>	Humphrey Gocart		Gocart

4. If the test is satisfactory, click Save.

### *Editing Filters*

To edit a filter, use the following steps:

1. On the Dynamic Group Form, enable the checkbox next to the filter you want to edit.
2. Click the Edit Selected button to edit the group filter.
3. When the Edit page is displayed, change the search parameters as necessary.
4. You can click Test to verify that your filter works as planned.
5. If you are satisfied with your changes, click Save to save the filter.

### *Removing Filters*

To remove a filter, use the following steps:

1. On the Dynamic Group Form, enable the checkbox next to the filter(s) you want to remove.
2. Click the Remove Selected button to remove the filter.
3. Click Save.

Directory Editor will remove the filter and save the change to the directory.

## Creating Organizational Units

Use the following steps to create a new organization:

1. From the Create tab, select Organizational Unit from the Select type drop-down menu.
2. Indicate where to add the new organization until by typing the parent's context information into the Parent Entry text field (or click Browse to locate the information). For example, you might enter:

`ou=Groups,dc=example,dc=sun,dc=com`

3. When you are finished, click Continue.

---

**NOTE** The Cancel button clears the Organization text box.

---

The Quick Create page is displayed (see Figure 4-20).

**Figure 4-20** Quick Create Tab

The screenshot shows a 'Create' window with a title bar and 'Save' and 'Cancel' buttons. Inside, there are four tabs: 'Quick Create', 'Common Attributes', 'Locality', and 'Other'. The 'Quick Create' tab is active. Below the tabs, there is a legend: '\* Indicates required field'. Under the heading 'Common Attributes', there is a single required field: '\* Organization Name: Engineering'. The text 'Engineering' is entered in the text box.

4. You can proceed in one of two ways:
  - The Organization Name is the only required attribute for this object type. Type a name in the text box provided, and then click Save.  
Directory Editor saves the new organizational unit and the Browse page is displayed. The new name will be displayed in the tree.
  - Use the remaining tabs on this page to specify additional, but optional, information about the object.

The following sections provide instructions for completing each tab on the Create page:

  - “Completing the Common Attributes Page” on page 74
  - “Completing the Locality Page” on page 74
  - “Completing the Other Page” on page 76
5. When you are finished entering the new information on the Create page (being sure to provide all *required* information), click Save to add the new organizational unit to the directory.

## Completing the Common Attributes Page

Use the Common Attributes page to provide the following information about the new organizational unit:

- **Organization Name** (*required*): Enter a name for the new organization.
- **Description**: Enter a description of the organizational unit.
- **Business Category**: Enter a business category for the organization.  
For example, you might enter Product Development for an Engineering organization

**Figure 4-21** Common Attributes Page

The screenshot shows a 'Create' dialog box with four tabs: 'Quick Create', 'Common Attributes', 'Locality', and 'Other'. The 'Common Attributes' tab is selected. Inside this tab, there are three text input fields, each preceded by an asterisk (\*) indicating they are required. The first field is 'Organization Name' with the value 'Engineering'. The second field is 'Description' with the value 'Engineering staff'. The third field is 'Business Category' with the value 'Product Development'. A legend in the top right corner states '\* Indicates required field'. At the top right of the dialog box are 'Save' and 'Cancel' buttons.

## Completing the Locality Page

Use the Locality page to provide the following information about the new organizational unit:

- **Locality**: Enter the organization's location. For example, the city and state.
- **Post Office Box**: Enter the organization's post office box (if applicable).
- **Postal Address**: Enter the organization's street address.
- **Preferred Delivery Method**: Indicate how mail, packages, equipment, and so forth should be shipped to this organizational unit.
- **Office Name**: Enter the name of the facility where the organization is located.
- **Registered Address**: Enter the organization's registered address if the information is different from the postal address.

For example, if the new organization maintains offices in Austin, TX but is officially registered in Delaware, enter the Delaware address in this text box.

- **Street Address:** Enter the organization's street address.
- **State/Province:** Enter the state name or province name.
- **Search Guide:** Enter suggested search criteria if you are using the entry as the base object in the directory tree for a search operation.
- **See Also:** Enter another Directory Server entry that may contain information related to this entry.

**Figure 4-22** Locality Page

The screenshot shows a web form titled "Create" with a "Save" button in the top right corner. Below the title bar are four tabs: "Quick Create", "Common Attributes", "Locality", and "Other". The "Locality" tab is currently selected. The form contains the following fields:

Locality:	Austin, TX
Post Office Box:	
Postal Address:	555 Desperado Lane
Postal Code:	55555-0000
Preferred Delivery Method:	UPS, Fed Ex
Office Name:	MyCompany, Inc
Registered Address:	
Street Address:	555 Desperado Lane, Building 3
State/Province:	Texas
Search Guide:	Austin
See Also:	Centex

## Completing the Other Page

Use the Other Page to provide miscellaneous information about the new organizational unit, as follows:

- **Password:** Enter a password to control who can access and change information related to this organization.
- **Confirm Password:** Re-enter the password to confirm it.
- **Destination Indicator:** Enter the city and country associated with the entry needed to provide Public Telegram Service. (Generally used in conjunction with the `registeredAddress` attribute)
- **Facsimile Telephone Number:** Enter the organization's facsimile number.
- **International ISDN Number:** Enter the organization's ISDN (Integrated Services Digital Network) number.
- **Telephone Number:** Enter the organization's telephone number.
- **Teltex Terminal ID:** Enter the terminal identification number for a Teltex terminal (text telephone).
- **Telex Number:** Enter the organization's telex number.
- **x121 Address:** Enter the organization's Data Terminal Equipment (DTE) address to the company network.



## Creating Domains

Use the following steps to create a new domain in the directory:

1. From the Create tab, select Domain from the Select type drop-down menu.
2. Indicate where to add the new domain by typing the parent's context information into the Parent Entry text field (or click Browse to locate the information). For example, you might enter:

```
dc=example,dc=sun,dc=com
```

3. When you are finished, click Continue (or press your Enter/Return key).

---

**NOTE** The Cancel button clears the Organization text box.

---

The Quick Create page is displayed. (see Figure 4-23).

**Figure 4-23** Quick Create Page

The screenshot shows a 'Create' dialog box with a title bar containing 'Create' and 'Save' and 'Cancel' buttons. Below the title bar are four tabs: 'Quick Create', 'Common Attributes', 'Locality', and 'Other'. The 'Quick Create' tab is selected. Inside the dialog, there is a section titled 'Common Attributes' with a sub-label '\* Domain Name:' followed by a text input field. A note '\* Indicates required field' is located in the top right corner of the dialog area.

4. You can proceed in one of two ways:
  - o The Domain Name is the only required attribute for this object type. Type a name in the text box provided (for example testDC), and then click Save.  
Directory Editor saves the new domain name to the directory and the Browse page is displayed.
  - o Use the remaining tabs on this page to specify additional, but optional, information about the domain.

The following sections provide instructions for completing each tab on the Create page:

- “Completing the Common Attributes Page” on page 74
  - “Completing the Locality Page” on page 74
  - “Completing the Other Page” on page 76
5. When you are finished entering the new information on the Create page (being sure to provide all *required* information), click Save to add the new domain to the directory.

## Completing the Common Attributes Page

Use the Common Attributes page to provide the following information about your domain:

- **Domain Name** (*required*): Enter a name for the new domain.  
For example, engineering.example.sun.com
- **Organization**: Enter a the name of the organizational unit to be associated with the new domain.
- **Description**: Enter a description of the new domain.

## Completing the Locality Page

Use the Locality page to provide the following information about the new organizational unit:

- **Locality Name**: Enter the domain's location.
- **Post Office Box**: Enter a post office box (if applicable).
- **Postal Address**: Enter a street address.
- **Postal Code**: Enter the zip or postal code of the domain.
- **Preferred Delivery Method**: Indicate how mail, packages, equipment, and so forth should be shipped to the organizational associated with this domain.
- **Office Name**: Enter the name of the facility where the domain is located.
- **Registered Address**: Enter the domain's registered address if the information is different from the postal address.

For example, if the organization maintains offices in Austin, TX but is officially registered in Delaware, enter the Delaware address in this text box.

- **Street Address**: Enter the domain's street address.

- **State/Province:** Enter the state name or province name.
- **Search Guide:** Enter suggested search criteria if you are using the entry as the base object in the directory tree for a search operation.
- **See Also:** Enter another Directory Server entry that may contain information related to this entry.

## Completing the Other Page

Use the Other Page to provide miscellaneous information about the new domain, as follows:

- **Associated Name:** Enter a name in the organizational directory tree that is associated with a DNS domain.
- **Password:** Enter a password to control who can access and change information related to this domain.
- **Confirm Password:** Re-enter the password to confirm it.
- **Business Category:** Enter a business category for the domain. For example, you might enter Product Development for an Engineering-related organization
- **Destination Indicator:** Enter the country and city associated with the entry needed to provide Public Telegram Service. (Generally used in conjunction with the `registeredAddress` attribute)
- **Facsimile Telephone Number:** Enter the domain's facsimile number.
- **International ISDN Number:** Enter the domain's ISDN (Integrated Services Digital Network) number.
- **Telephone Number:** Enter the domain's telephone number.
- **Teltex Terminal ID:** Enter the terminal identification number for a Teltex terminal (text telephone).
- **Telex Number:** Enter the domain's telex number.
- **x121 Address:** Enter the domain's Data Terminal Equipment (DTE) address to the company network.

## Creating Localities

Use the following steps to create a new locality:

1. From the Create tab, select Locality from the Select type drop-down menu.
2. Indicate where to add the new locality by typing the parent's context information into the Parent Entry text field (or click Browse to locate the information). For example, you might enter:

```
dc=example,dc=sun,dc=com
```

3. When you are finished, click Continue (or press your Enter/Return key).

---

**NOTE** The Cancel button clears the Organization text box.

---

The Quick Create page is displayed (see Figure 4-24).

**Figure 4-24** Quick Create Tab

The screenshot shows a window titled "Create" with two tabs: "Quick Create" (selected) and "Common Attributes". In the "Quick Create" tab, there is a section titled "Common Attributes". Below this title, there is a text input field labeled "\* Locality Name:" containing the text "North America". To the right of the input field, a small legend states "\* Indicates required field". In the top right corner of the window, there are two buttons: "Save" and "Cancel".

4. You can proceed in one of two ways:
  - o The Locality Name is the only required attribute for this object type. Type a name in the text box provided, and then click Save. Directory Editor saves the new locality name and the Browse page is displayed. The new Locality name will be displayed in the tree.
  - o Use the Common Attributes tab to specify additional, but optional, information about the locality. (Go to Step 1.)
5. Select the Common Attributes tab.

6. Type the following information in the text boxes provided:

- o **Locality Name** (*required*): Enter a name for the new locality.

---

**NOTE** Directory Editor automatically completes this field if you provided a Locality Name on the Quick Create tab.

---

- o **Street Address:** Enter the street and city address for this locality.
- o **State/Province:** Enter the state or province for this locality.
- o **Description:** Enter a description of the locality.
- o **Search Guide:** Enter suggested search criteria if you are using the entry as the base object in the directory tree for a search operation.
- o **See Also:** Enter another Directory Server entry that may contain information related to this entry.

For example, you might enter the following information for a North American sales office:

**Figure 4-25** Locality Common Attributes Page

The screenshot shows a 'Create' dialog box with two tabs: 'Quick Create' and 'Common Attributes'. The 'Common Attributes' tab is active. It contains a form with the following fields and values:

- \* Locality Name:** North America
- Street Address:** 333 Casa Blanca Cove, Austin
- State/Province:** TX
- Description:** North American Sales Headquarters
- Search Guide:** sales headquarters
- See Also:** sales

A legend in the top right corner states: '\* Indicates required field'.

7. When you are finished with this tab, you can

- o Click Save to save the new information to the directory and return to the initial Create page.
- o Click Cancel to clear the text boxes and return to the initial Create page.

# Editing Objects

---

<b>NOTE</b>	Most of the procedures for editing an object's attributes are the same as those you used to create the object. Review the instructions provided in "Creating Objects" on page 45 if necessary.
-------------	--

---

This section explains how to access the Edit pages and provides instructions for enabling/disabling, renaming, and deleting objects using Directory Editor. The information is organized as follows:

- "Accessing an Edit Page" on page 82
- "Enabling or Disabling Objects" on page 87
- "Renaming Objects" on page 89
- "Deleting Objects" on page 91

## Accessing an Edit Page

You can access the Edit pages, using one of the following methods:

- From the Home page, click the Edit My Directory Information link to edit your information in the directory.

---

<b>NOTE</b>	The Edit My Directory Information link does not display unless the user who is logged in has write access to their entry.
-------------	---

---

An Edit page is displayed so you can edit the following information:

- Common Attributes (review page 48)
- Group Membership (review page 50)
- Location Information (review page 51)
- Misc Attributes (review page 52)
- Extensions (review page 53)
- Select the Browse tab, enable a checkbox to the left of a node in the Browse tree.

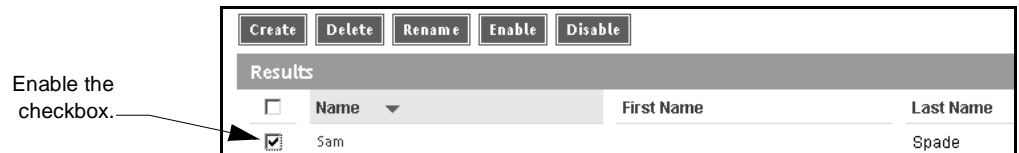
**Figure 4-26** Enable the Checkbox in the Browse Tree

The buttons located above the Browse tree become active. Use these buttons to perform any of the following tasks:

- Edit the object's attributes
- Enable or Disable the object (see page 87 for instructions)
- Rename the object (see page 89 for instructions)
- Delete the object (see page 91 for instructions)
- Select the Browse tab and click the linked name of the node you want to edit.

**Figure 4-27** Click the Linked Name

- Select the Search tab, search for the object you want to edit, enable the checkbox to the left of the object name in the Results table.

**Figure 4-28** Enable the Object in the Results Table

Either method opens the Edit an Object page (shown in the following figure):

**Figure 4-29** Example Edit an Object Page

SaveCancel

Common AttributesGroup MembershipLocation InfoMisc AttributesExtensions

\* Indicates required field

Object: cn=Michael Miller,ou=People,dc=central,dc=sun,dc=com

User Name: mmiller

\* First Name: Michael

\* Last Name: Miller

Common Name/Nickname: Michael MillerAdd

Password: \*\*\*\*\*

Confirm Password: \*\*\*\*\*

Email: mmiller@example.comAdd

Telephone: 111-111-0000Add

Department #: 111-111-0009

Photo URL:

Employee #: 103

**NOTE**

If you are editing an object and remove an object class from the Extensions tab, Directory Editor will remove all attributes belonging to the removed object class when you save the object.



## Importing and Exporting Group Members

When you edit a group, Directory Editor allows you to import and export group members using the Import/Export tab on the Group form.

**Figure 4-30** Import/Export Tab

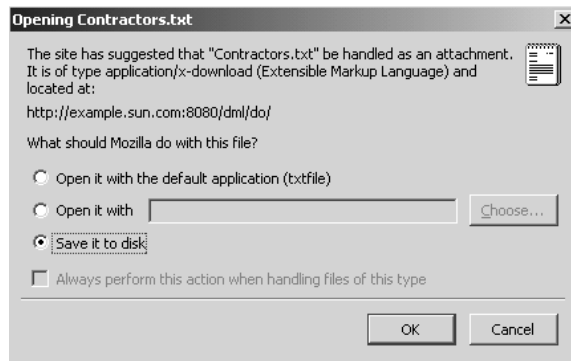
The screenshot shows the 'Import/Export' tab selected among four tabs: 'Default Dynamic Group Form', 'Static Group', 'Import/Export', and 'Extensions'. The 'Export' section contains the text 'Export the group members to a file.' and a 'Download Members' button. The 'Import' section contains the text 'Import group members from a file.' and a note: 'The file should have the DN of each member to add to the group on it's own line.' Below this is a 'File:' label, a text input field, and a 'Browse...' button. At the bottom of the import section are two checkboxes: 'Import members not found in directory?' and 'Replace existing members?'. An 'Import Members' button is at the very bottom.

### *Exporting Group Members*

When you export group members, you download all of the member DNs to a newline-delimited text file.

To export group members:

1. Select the Import/Export tab and then click the Download Members button.
2. When the Opening *<group\_name>.txt* dialog box is displayed, (Figure 4-31), enable one of the radio buttons to indicate how your browser should handle the file.

**Figure 4-31** Specifying How Your Browser Should Open the File

- Open it with the default application (for example, txtfile)
  - Open it with (You must enter an application name in the text field provided.)
  - Save it to disk (You must specify a file name and specify a folder in which to save the file.)
3. After making your selection, click OK to close the dialog box.
  4. Click Save to save the exported group file to your directory.

You should now be able to open the .txt file in any standard text editor (such as Microsoft® Notepad). At this point, you can edit the file, and then import it using the instructions provided in the next section, or store the file for backup purposes.

### *Importing Group Members*

When you import group members, Directory Editor expects a newline-delimited text file with a member DN on each line. (Directory Editor ignores empty lines and lines starting with the pound sign (#).)

To import members from a file:

1. Type the file name into the text box provided or click Browse to open a File Upload dialog box so you can locate and select the file name.
2. Enable on of the following checkboxes:
  - **To import members** (even if they are not found in the Directory Server), enable the Import members not found in directory? check box.
  - **To replace existing group members** with members in the file (instead of adding members in the file to the existing group members), enable the Replace existing members? check box. This feature is available for both group creates and edits. Directory Editor does not save import changes to the directory until you save the group.
3. When you are ready, click Import Members to import the file.
4. Click Save to save the new group members to your directory.

## Enabling or Disabling Objects

You use the Enable and Disable buttons to activate or inactivate a user account for the Sun Directory Server. Once inactivated, a user cannot bind to the directory and the authentication operation will fail.

---

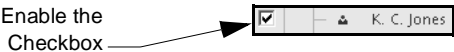
<b>NOTE</b>	When you create an object, it is enabled in the directory by default. However, if you disabled the object at some point, you may need to re-enable it again.
-------------	--

---

To enable or disable an object in the directory, use the following steps:

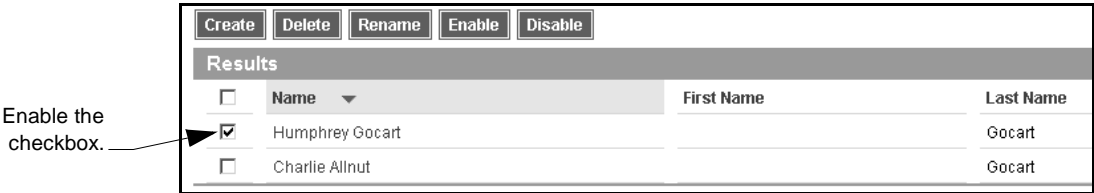
1. Open the Enable page or the Disable page using one of the following methods:
  - o Select the Browse tab, enable a checkbox to the left of the node in the Browse tree, and then click the Enable or Disable button.

**Figure 4-32** Enable the Checkbox in the Browse Tree



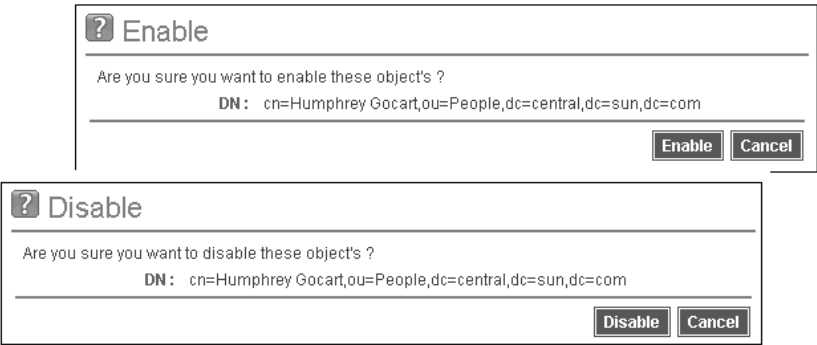
- o Select the Search tab and search for the object to edit. When the Results table is displayed, enable the checkbox to the left of the object name, and then click the Enable or Disable button.

**Figure 4-33** Click the Object Name in the Results Table



The Enable or Disable page is displayed as shown in the following figure:

**Figure 4-34** Example Enable/Disable Pages



2. You are prompted to confirm the action.
  - Click Enable or Disable (whichever is appropriate) to continue.
  - Click Cancel to return to the Search page without making any changes to the object.

## Renaming Objects

To rename an object, use the following steps:

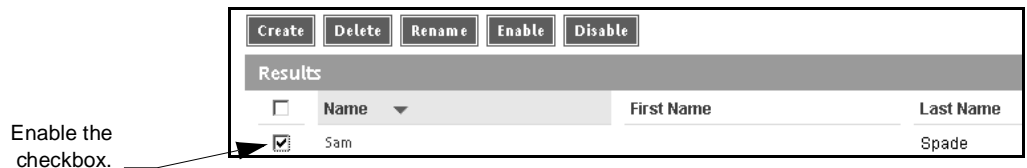
1. Open the Rename page using one of the following methods:
  - Select the Browse tab, enable a checkbox to the left of the object in the Browse tree, and then click the Rename button.

**Figure 4-35** Enable the Checkbox in the Browse Tree



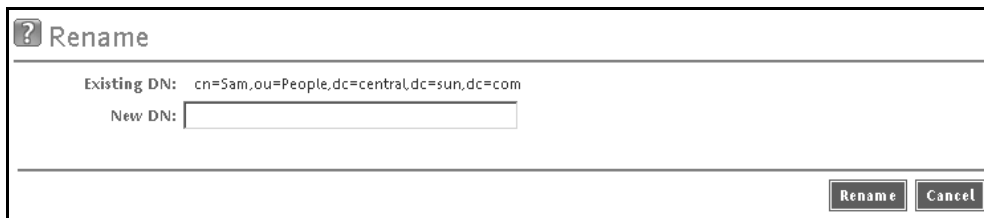
- Select the Search tab, search for the object to edit, enable the checkbox to the left of the object name in the Results table, and then click the Rename button.

**Figure 4-36** Enable the Object in the Results Table



The Rename page is displayed as shown in the following figure:

**Figure 4-37** Rename Page



**Rename**

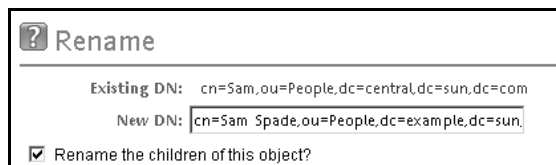
Existing DN: cn=Sam,ou=People,dc=central,dc=sun,dc=com

New DN:

**Rename** **Cancel**

2. Enter the new name in the text box provided.
3. When you are ready, click the Rename button.

**Figure 4-38** Rename Page



**Rename**

Existing DN: cn=Sam,ou=People,dc=central,dc=sun,dc=com

New DN:

☒ Rename the children of this object?

Directory Editor renames the object and returns you to the Browse page so you can verify the change.

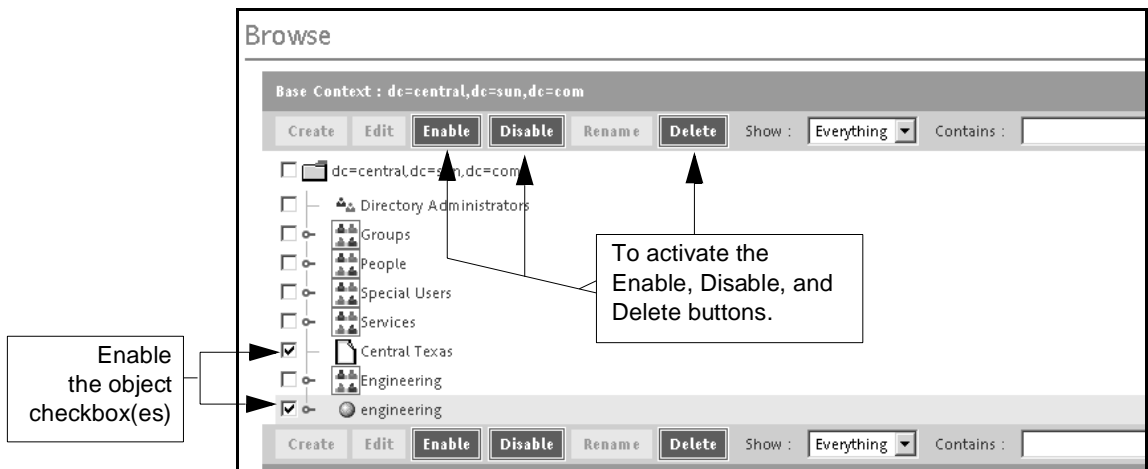
## Deleting Objects

If you want to delete an object you have created, use the following procedure:

1. Select the Browse tab.
2. Enable the checkboxes located to the left of the object(s) you want to delete.

The Enable, Disable, and Delete buttons located above the Browse tree will become active.

**Figure 4-39** Deleting Objects



3. Click the Delete button.

The Delete page is displayed. You are asked to confirm whether to delete the selected object(s) and the children of the selected object.

4. When you are ready, click the Delete button to delete the selected object(s).

Directory Editor immediately removes the object (and children if specified) and then returns you to the Browse page.





# Working With Forms

This chapter explains how to customize the default Directory Editor forms and how to create new forms for your site. The information is organized as follows:

- “Overview” on page 93
- “Accessing the Forms Page” on page 97
- “Customizing Forms” on page 98
- “Creating New Forms” on page 111
- “Testing Your Form” on page 113

## Overview

Directory Editor provides the following forms, which you can customize by incorporating business logic and manipulating data before presenting them to your end users:

**Table 5-1** Directory Editor Forms

Form Name:	Use this form to:
applicationProcess	Edit Directory Editor configuration objects.
Default Browse Form	List the managed directory's contents
Default Create Form	Create users, groups, or organizations
Default Delete Form	Delete users, groups, or organizations
Default Disable Form	Disable specific objects
Default Enable Form	Enable specific objects

**Table 5-1** Directory Editor Forms *(Continued)*

Default Home Form	Configure a Home page
Default Organization Picker Form	Select organizations
Default Rename Form	Rename objects
domain	Define domains for the directory
extensibleobject	Define attribute values for a class of service (COS) template.
groupofnames	Create a list of email distribution aliases.
groupofuniquenames	Add members to or remove members from a group, rename groups, search groups, export DNs to a text file, or display group memberships.
groupOfURLs	Create and edit dynamic groups.
inetorgperson	Control user attributes and to which groups an end-user will belong.
locality	Define localities for the directory
organizationalperson	Create and define user attributes for organizational members
organizationalunit	Create and define attributes for organizational units

- 
- NOTE**
- You cannot edit the *standard* forms (those forms beginning with “Default”), but you can create a customized version of these forms by clicking the Customize button to open the form editor.
  - For information about the specific HTML components that define the default Directory Editor forms, see “Appendix A, “HTML Components Used to Define Directory Editor Forms.”
-

## How Do Forms Work?

Directory Editor forms contain rules that govern how the browser displays user view attributes on a application's web page. These forms control:

- **Page layout and display characteristics.** Controls whether fields are visible, read-only, or hidden from view.  
  
Visible field types include simple text boxes, radio buttons, and selection boxes with multiple values. Fields values also can be based on other fields.
- **How data is used on a page.** Captures data dynamically from a resource (such as a Directory Server) or calculates the data from other fields.
- **How data comes into the system.** Interfaces with web pages or non-interactive systems. In this role, the form has no visual fields, but still provides rules to set default values and other field values.

For example, the Full Name field may not be visible to an administrator using the page, but the field value can be set based on values the end-user enters into the First Name, Middle Name, and Last Name fields.

Populating fields from other fields reduces the amount of data entry that users and administrators must perform, consequently reducing potential data entry errors. Similarly, by providing option menus instead of text input fields, an administrator can select a department from a list instead of having to type the department name.

Various factors affect how the browser displays a form. However, form behavior within a browser is primarily determined by how display components are used in the form. You can associate form fields with a display component that determines how the field is displayed in the browser.

## Why Edit the Forms?

Why customize the default Directory Editor forms, which already provide all the fields that you need to perform actions within the product? Customizing the default forms allow you to better enforce your company's policies and processes:

- Preserve privacy by limiting the amount of user account information displayed on the screen.
- Omit nonessential information to reduce distractions for users performing a specific task. Display only the fields needed to accomplish the current task.

Customizing the default fields in forms allows you to extend and customize the application for your environment. Specifically, you can customize the default forms to:

- Address the specific needs of the users in your organization.

This type of customization is particularly important when different types of administrators must access user data, but it is not appropriate for all of the administrators to have access to all of the user account attributes. For example, a human resources administrator will need to view different subset of user account attributes than a help desk administrator.

---

**NOTE** Directory Editor does not display attributes to users who do not have read access for those attributes. In addition, Directory Editor will display read-only attributes as read-only fields.

---

- Control the display and content of the user account attributes, particularly attributes displayed on the Create and Edit pages. These pages contain most of the attributes that need to be controlled.

- Define default values for user view attributes and their associated attributes.

For example, you could define a default account name for a user based on a combination of first and last initials and their employee ID number instead of the administrator having to key in the information.

- Post-process user view attributes data entry.

For example, you can automatically create a mail account based on the value of a location field.

- Control screen real estate by positioning multiple fields on one line. By rearranging fields, you can make the form more closely resemble a printed form or pre-existing web form.

- Define rules for the way hidden attributes are calculated.

For example, a user's email address can be calculated to be the user's first name, a period, their last name, then the mail domain: *joe.user@example.com*.

- Correspond to the Directory Server environment.

You can create new forms to allow creating and editing objects that are unique to the Directory Server's schema.

# Accessing the Forms Page

If you want to customize or design a form, select the Configure tab, and then click the Forms tab. (See the following figure.)

**Figure 5-1** Forms Tab

Forms

Save New Form

Available Forms		
Name ▼	Select	Action
applicationprocess	<input checked="" type="radio"/> Default	Customize
costemplate	<input checked="" type="radio"/> Default	Customize
Default Browse Form	<input checked="" type="radio"/> Default	Customize
Default Create Form	<input checked="" type="radio"/> Default	Customize
Default Delete Form	<input checked="" type="radio"/> Default	Customize
Default Disable Form	<input checked="" type="radio"/> Default	Customize
Default Enable Form	<input checked="" type="radio"/> Default	Customize
Default Home Form	<input checked="" type="radio"/> Default	Customize
Default Organization Picker Form	<input checked="" type="radio"/> Default	Customize
Default Rename Form	<input checked="" type="radio"/> Default	Customize
domain	<input checked="" type="radio"/> Default	Customize
extensibleobject	<input checked="" type="radio"/> Default	Customize
groupofnames	<input checked="" type="radio"/> Default	Customize
groupofuniquenames	<input checked="" type="radio"/> Default	Customize
groupofurls	<input checked="" type="radio"/> Default	Customize
inetorgperson	<input checked="" type="radio"/> Default	Customize
locality	<input checked="" type="radio"/> Default	Customize
organizationalperson	<input checked="" type="radio"/> Default	Customize
orgnizationalunit	<input checked="" type="radio"/> Default	Customize

This page contains:

- **Name column:** Lists all of the default forms provided with Directory Editor. (Use the scroll bar to view all of the forms in the Name column.)

- **Select column:** Contains radio buttons that are all labeled *Default*.

After you customize any of the default forms, Directory Editor adds a *Custom* radio button to this column. Changing this radio button controls whether Directory Editor uses the Default or Custom form.

- **Action column:** Contains Customize or Edit buttons. Click these buttons to customize the selected form.

If you customize any of the default forms, Directory Editor changes the Customize button to an *Edit* button.

- **Save button:** Click to preserve changes to the Default or Custom radio buttons.
- **New Form button:** Click to create a new form.

Instructions for using this page to customize the default Directory Editor forms or to create new forms for your site are provided in the following sections:

- “Customizing Forms” on page 98
- “Creating New Forms” on page 111

## Customizing Forms

This section provides instructions and examples for customizing forms, and is organized as follows:

- “Opening a Form” on page 99
- “Working with the Forms Page” on page 100
- “Performing Common Tasks” on page 103

## Opening a Form

To open a form for customizing, click the Customize or Edit button in the Action column on the Available Forms page.

A Forms page is displayed, which consists of two views:

- A *tree-view* list of all elements associated with that form on the left
- A *property view* with a graphical representation of the form on the right.

For example, if you decided to customize the Default Create Form, the following page would display:

**Figure 5-2** Default Create Form

When you select elements in the Forms tree to edit, the property view interface changes so you can edit the properties associated with that element. For more detailed information, continue to the next section.

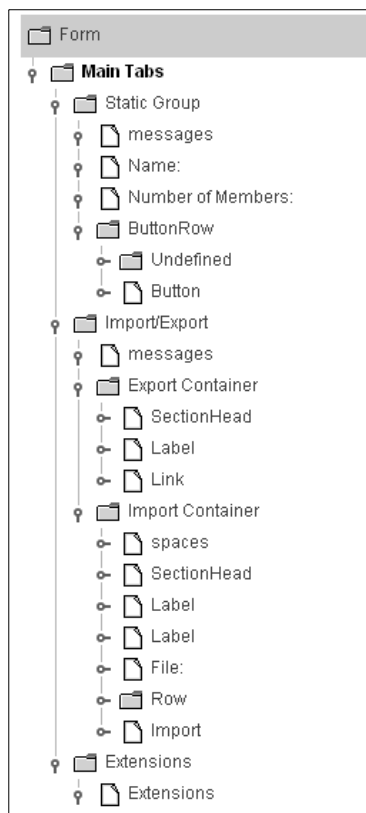
## Working with the Forms Page

This section provides instructions for working with the Forms page.



### Working in Tree View

The tree view (located on the left side of the Forms page) shows a hierarchical view of the form elements. The tree lists each element in order, nesting children under their parent.

**Figure 5-3** Tree View Display (Expanded)





The node icons used in the tree are based on whether the form has children, and include the following:



-  The folder icon displays if the form item has children.
-  The page icon displays if the form item does not have children.



The tree also contains the following icons:

- Click the key icon  to view any nested children in that form item node.
- Click this key icon  to collapse the form item.

To customize the properties associated with each form or field element, click the node label, and the right side of the Forms page changes as follows:

- Click Form  to specify a name, description, title, and/or a subtitle for the form (see “Identifying the Form” on page 103).
- Click any other named element  to specify the field properties and location (see “Customizing Field Properties” on page 105 and “Changing Field Locations” on page 108).
- Click the New Field button to create a new field for the form (see “Creating New Fields” on page 110).

## Working in Property View

The property view (located on the right side of the Forms page) provides a graphical representation of the form you are customizing.

When you initially open a form for editing, the property view displays the form as it is currently configured. However, when you select an element in tree view, the property view changes so you can edit properties associated with that element.

For example, if you select the Default Create Form, the initial property view is shown in Figure 5-4.

**Figure 5-4** Initial Property View: Create Form



The dialog box titled "Initial Property View: Create Form" contains the following elements:

- Buttons: "Continue" and "Cancel" at the top right and bottom right.
- Text: "\* Indicates required field" in the top right corner.
- Field: "New Object Type:" followed by a dropdown menu showing "User".
- Field: "\* Parent Entry:" followed by a text input field and a "Browse" button.

Then, if you select the New Object Type node in tree view, the property view changes as shown in Figure 5-5.

**Figure 5-5** Property View: Editing New Object Type Element

**Field DML\_prop\_label\_type**

Field Properties   Move Field

Name:

Title:  Options: ☐ Required

**Display Properties**

Type:

Null Value:

Size:

Options: ☐ Sorted   ☐ Multiple   ☐ No New Row

Allowed Values:   
groupOfUniqueNames, top  
organizationalUnit, top  
domain, top  
locality, top

Allowed Values Expression:

When you are editing elements, the property view contains two tabs:

- **Field Properties tab:** Provides a schematic representation of the form and enables you to customize properties associated with that element.
- **Move Field tab:** Enables you to control the location of different form elements.

After changing form elements in property view, if you select the Form node in tree view, the property view changes to reflect your changes (if your changes affected the visible features of the form).

---

**NOTE**      Remember, this view is only a preview of the changes made to your form — not a working form.

---

## Performing Common Tasks

This section describes how to perform the common tasks associated with customizing Directory Editor forms. The information is organized as follows:

- “Identifying the Form” on page 103
- “Customizing Field Properties” on page 105
- “Changing Field Locations” on page 108
- “Creating New Fields” on page 110
- “Deleting Fields” on page 110
- “Saving Changes” on page 111

### Identifying the Form

To identify your customized form, you can specify a name, description, title, and/or subtitle as follows:

1. If the Form node is not already selected, click the Form folder icon (top node in the Forms tree) and the page changes as follows:

**Figure 5-6** Identifying a Default Create Forms

The screenshot shows a dialog box titled "Forms". On the left is a tree view with a "Form" folder icon selected. Below it are three items: "Select type:", "Organization", and "buttons". On the right side of the dialog are four text input fields labeled "Name", "Description", "Title", and "Sub-title". At the bottom of the dialog are four buttons: "Save", "New Field", "Cancel", and "OK".

2. Complete one or more of the following fields (all optional):
  - **Name:** Enter a name for this form.
  - **Description:** Enter a meaningful description of the form.
  - **Title:** Enter the text you want to display at the top of the form. For example, *MyCompany Create Form*.
  - **Subtitle:** Enter the text you want to display beneath the form's title. For example, you might indicate the form's purpose: *Use this form to create new users, groups, and organizational units.*

3. When you are finished, click OK.

The Form node in tree view changes to use the Title you specified, and the new form displays on the right side of the page. For example:

**Figure 5-7** New Form Page

The screenshot shows a window titled "Forms" with a "Continue" and "Cancel" button in the top right corner. On the left side, there is a tree view under the heading "MyCompany Create Form". The tree view contains three items: "DML\_prop\_label\_type", "DML\_prop\_label\_parent", and "buttons". Below the tree view are three buttons: "Save", "New Field", and "Cancel". The main area of the window is titled "MyCompany Create Form" and contains the text "Use this form to create new users, groups, organizational units, domains, and localities." Below this text is a form with two fields: "New Object Type:" with a dropdown menu showing "User", and "\* Parent Entry:" with a text box and a "Browse" button. A note "\* Indicates required field" is located in the top right of the main area. At the bottom right of the window are "Continue" and "Cancel" buttons.

4. Continue as follows:
  - If you have other changes to make to the current form, select another node in the tree.
  - If you want to add one or more new fields, click the New Field button and proceed to "Creating New Fields" on page 110.
  - If you have no other changes to make, click Save.

## Customizing Field Properties

To customize properties for a particular field on your form, use the following steps:

1. In the Form tree, click the field element you want to customize.  
For example, click Select type and the following page is displayed:

**Figure 5-8** Customizing Field Properties

Forms

MyCompany Create Form

New Object Type:  
Text  
buttons

Save New Field Cancel

**Field New Object Type:**

Field Properties Move Field

Name: primaryObjectClass

Title: New Object Type: Options: ☐ Required

**Display Properties**

Type: Select

Null Value:

Size:

Options: ☐ Sorted ☐ Multiple ☐ No New Row

Allowed Values:  
inetOrgPerson, organizationalPerson, pers  
groupOfUniqueNames, top  
organizationalUnit, top  
domain, top  
locality, top

Allowed Values Expression:

OK Delete

The content of the Field Properties tab will change, depending on which field element you select. The following table lists all of the different Field Properties parameters you might encounter as you work.

**Table 5-2** Field Properties

Property	Description
Name	<p>Type a name for this field.</p> <ul style="list-style-type: none"> <li>You must enter a name that specifies a view path for all fields that display as editing components (such as text boxes, checkboxes, and selects).</li> <li>You do not have to name fields that do not display as editing components (such as <code>SectionHead</code> or <code>Javascript</code>); however, you can name non-editing fields if they are referenced by another form through a Field reference.</li> <li>You must use the following format to name fields that correspond to LDAP attributes:  <code>attributes[&lt;attribute_name&gt;].value</code>            For example, use the following format to name a cn attribute:  <code>attributes[cn].value</code> </li> </ul>
Title	Specify text to display adjacent to the field.
Options:	<p><b>Required checkbox:</b> Enable this checkbox if the field is required to process the form. If you enable this option:</p> <ul style="list-style-type: none"> <li>The user must enter a non-null value in this field before submitting the form.</li> <li>A red asterisk is displayed to the left of the field and a message displays at the top of the form to explain that a red asterisk denotes required fields.</li> </ul>
<b>Display Properties:</b>	
Type	Use the drop-down menu to select which type of display element you want to use for the new field. For example, do you want to create a text box or a button?
Available Title	Enter a title for the available values of a multi-select object.
Selected Title	Enter a title for the selected values of a multi-select object.
Allowed Values	Use this text box to enter allowed values for the component. The end-user can select only those values provided in this list. (Some values may already be displayed by default.)
Allowed Values Expression	Use an expression to create a list of allowed values for the component. For convenience when setting properties in XML forms, you can specify the allowed values as a comma-delimited list.
Size	Specify the character width of the control (for text boxes).
Maximum Length	Specify the maximum number of characters that can be entered in a text box.
Display Name	Specify text to display next to a check box.
Rows/Columns	Specify the number of rows and columns to display in a text area.
Null Label	Specify a label to display in select fields to indicate that nothing is selected.
Labels	Specify radio button labels.

**Table 5-2** Field Properties *(Continued)*

Show Child Count	Enable this checkbox to see how many children are under each node in the Forms tree. The number will be displayed in brackets after the node name. For example, People [ 21 ].
Show Only Branches	Enable this checkbox to display just the branches in the Forms tree instead of the branches and leaves. For example, if you open a branch that only has leaf nodes in the Browse dialog — you will not see anything; however, on the Browse tab you will see all the leaves.
Allow Multiple Selections	Enable this checkbox to allow the user to select multiple elements in the Forms tree.
Show Current Selections on Footer	Enable this checkbox to display currently selected elements in the Forms tree footer.
Show Current Selections on Header	Enable this checkbox to display currently selected elements in the Forms tree header.
Yoke to Active Mode	<p>Enable this checkbox to allow the window to automatically scroll to the last node with which you interacted.</p> <p>Normally, if you are working with a large Forms tree, and you expand a branch node that you had to scroll down to see, when you are done with the expansion you would have to reset the tree to the top. However, if you enable this property, the tree will automatically scroll the window to the expanded node.</p>
Options:	<p>Enable one or more of the following checkboxes — as necessary:</p> <ul style="list-style-type: none"> <li>• <b>Left Label:</b> Forces the label to display to the left of a field.</li> <li>• <b>No New Row:</b> Forces the field to appear to the right of the proceeding field. For example, Name fields are examples where this is useful, where it is desirable to allow the user to enter the last name, first name, and middle initial from right to left, rather than down the page.</li> <li>• <b>Multiple:</b> Controls whether multiple selections are available for a select field.</li> <li>• <b>Multi-Valued:</b> Allows multiple values for a text field.</li> <li>• <b>Ordered:</b> Controls whether multi-select values can be moved up and down in the selected list.</li> <li>• <b>Read-Only:</b> Creates a non-editable field.</li> <li>• <b>Secret:</b> Causes typed text to display as asterisks (*). (typically used for encrypted data, such as passwords)</li> <li>• <b>Sorted:</b> Controls whether select and multi-select values are sorted alphabetically.</li> </ul>

## 2. On the Field Properties tab, you can

- Customize one or more of the property parameters described in the preceding table, and then click OK.
- Click the Delete button (located at the bottom of the tab) to remove a field from this form. Repeat to delete additional fields.

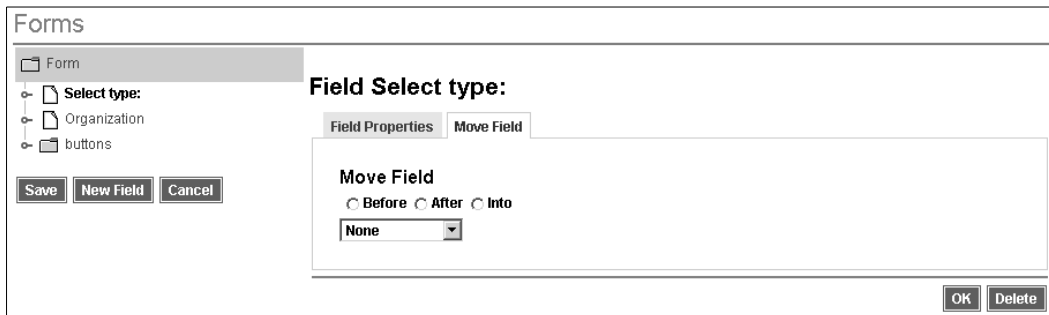
3. When you are finished customizing the properties for this field, you can:
  - Select the Move Field tab to change the field's location on this form. (Proceed to "Changing Field Locations" on page 108.)
  - Add or remove a field (Proceed to "Creating New Fields" on page 110 or "Deleting Fields" on page 110 respectively.)
  - Select a different field to customize that field's properties.
  - Save or Cancel your changes and return to the Available Forms page. (Proceed to "Saving Changes" on page 111.)

## Changing Field Locations

Use the following steps to change the location of a particular field on your form:

1. In the Form tree, click the field element you want to move.
2. When the Properties tabs are displayed, select the Move Field tab.

**Figure 5-9** Move Field Tab

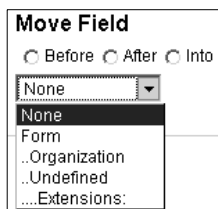


The Move Field tab contains three radio buttons and a drop-down menu. Elements displayed in the drop-down menu are indented to indicate whether they are tabs, fields, or nested children.



For example, in the following figure, *Form* is the form, *Organization* and *Undefined* are field elements, and *Extensions:* is a child of *Undefined*.

**Figure 5-10** Move Field Menu




---

**NOTE** You can only move field elements and children *into* the Form.

---

3. Decide where to move the selected field, enable one of the following buttons, and then make a selection from the drop-down menu:
  - **Before button:** Moves the selected field element above a field you select from the drop-down menu.
  - **After button:** Moves the selected field element below a field you select from the drop-down menu.
  - **Into button:** Moves the selected field element into a tab you select from the drop-down menu.
4. When you are finished with the Move Field tab, you can:
  - Select a different field to customize.
  - Add or remove a field (Proceed to “Creating New Fields” on page 110 or “Deleting Fields” on page 110 respectively.)
  - Save or Cancel your changes and return to the Available Forms page. (Proceed to “Saving Changes” on page 111.)

---

**NOTE** You must click the Save button located under the Form tree to save your change(s) to the directory.

---

## Creating New Fields

To Create a new field for your form:

1. Click the New Field button, located under the Form tree.

Directory Editor adds an undefined node to the Form tree, and the property view changes as follows:

**Figure 5-11** Creating a New Field

The screenshot shows the 'Forms' window with a tree view on the left containing 'Form', 'Select type:', 'Organization', and two 'Undefined' nodes. Below the tree are 'Save', 'New Field', and 'Cancel' buttons. The main area is titled 'Field Undefined' and contains two tabs: 'Field Properties' (selected) and 'Move Field'. The 'Field Properties' tab has input fields for 'Name' and 'Title', a 'Display Properties' section with a 'Type' dropdown set to 'None', and an 'Options' section with a 'Required' checkbox. At the bottom are 'OK' and 'Delete' buttons.

2. Complete the Field Properties tab as described in “Customizing Field Properties” on page 105.
3. Use the Move Field tab to position the new field on your form as described in “Changing Field Locations” on page 108.
4. When you are finished, click Save save your change(s) to the directory.

## Deleting Fields

To Delete a field element from your form, click the Delete button that is located at the bottom of the Field Properties tab. Directory Editor will remove that field element from the form (and tree view).

---

**NOTE** You must click the Save button located under the Form tree to save your change(s) to the directory.

---

## Canceling Changes

If you make a mistake while customizing a form or decide you do not want to use the form you customized, click the Cancel button located at the bottom of the Form tree.

Directory Editor will cancel any unsaved changes made to that form and immediately return you to the Available Forms page.

## Saving Changes

You can save your customization changes at any time by clicking the Save button located at the bottom of the Form tree. Directory Editor will immediately save all changes made to that form.

# Creating New Forms







To create a new form, use the following instructions:

1. Click the New Form button (located at the top and bottom of the Forms page).  
A new page displays (see the following figure).

**Figure 5-12** Creating a New Form

The screenshot shows a window titled "Forms". At the top, there is a label "Structural Class:" followed by a dropdown menu currently showing "account". Below this, on the left, is the label "Auxiliary Classes:". To the right of this label is a list box containing the following classes: bootableDevice, certificationAuthority, certificationAuthority-V2, dcObject, extensibleObject, ieee802Device, inetAdmin, inetDomain, and inetSubscriber. To the right of the list box are several small buttons: a right arrow, a double right arrow, a double left arrow, a left arrow, a plus sign, and a minus sign. To the right of these buttons is an empty rectangular area. At the bottom left of the window is a button labeled "Select Classes".

2. Select a structural class from the drop-down menu.

3. If applicable, use the Auxiliary Classes selection tool buttons to select one or more auxiliary classes for your new form, as follows:
  - Select one or more auxiliary classes from the available list and click the  button to move them to the selected list. (Press your Shift key and click on items in the list to select multiple auxiliary classes.)
  - Click the  button to move all available auxiliary classes to the selected list.
  - Click the  button to move all auxiliary classes from the selected list back to the available list.
  - Select auxiliary classes from the selected list and click the  button to move them back to the available list.
  - Use the  (move up) and  (move down) buttons to change the order of auxiliary classes in the selected list
4. After making your selections, click the Select Classes button to add the classes to your new form.

Another page displays, containing a tab for the structural class and one tab for each auxiliary class you selected.

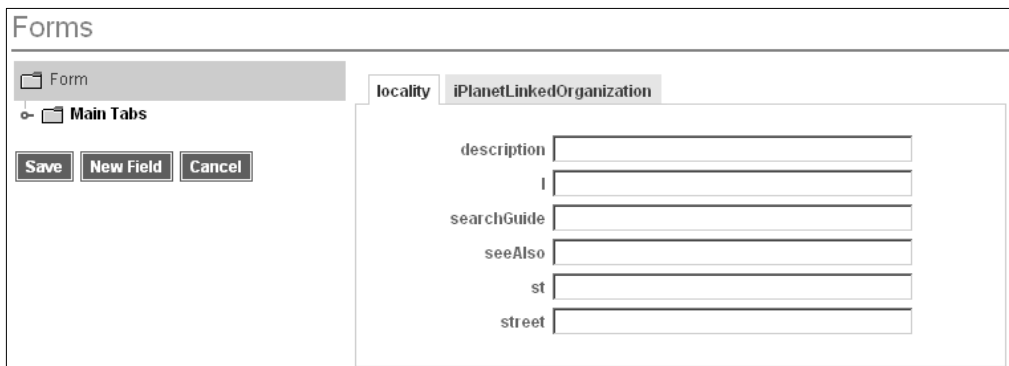
---

**NOTE** This page is a preview of the form you are creating.

---

For example, if you specified the `locality` structural class and the added `iPlanetLinkedOrganization` Auxiliary class, the new page would look like the following figure:

**Figure 5-13** Adding Classes to a Form

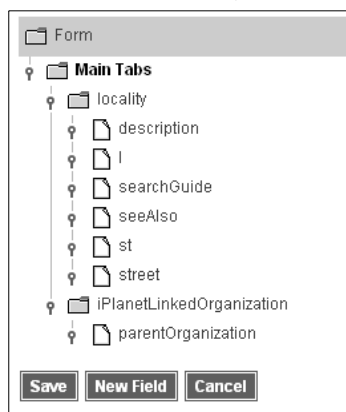


The screenshot shows a window titled "Forms". Inside, there's a sidebar on the left with a tree view showing "Form" and "Main Tabs". Below the tree are three buttons: "Save", "New Field", and "Cancel". The main area on the right displays a preview of the form. It has two tabs: "locality" (selected) and "iPlanetLinkedOrganization". The "locality" tab contains several text input fields with labels: "description", "I", "searchGuide", "seeAlso", "st", and "street".

The tab names, and the content of each tab, will differ with each structural class and auxiliary class selection.

5. Complete the sub-element text boxes on each tab, being certain to provide all required information (identified by a red asterisk).
6. Select the Form, Tab, and field element nodes to specify the field properties (see page 5-79) and to specify the field locations (see page 5-74) for each.

**Figure 5-14** Specify Properties for Each Node in the Tree



7. When you are finished, click Save to save your new form to the directory.

## Testing Your Form

The best way to test your form before loading it into your production environment, is to preview the form while you are editing it.

In addition, you may want to create a staging or test environment that you can use when you are developing new forms. You can export the staged forms (select Configuration -> Import/Export), and then import them into your production environment.

---

**NOTE** For importing and exporting instructions, see “Backing Up and Restoring Configurations” on page 149.

---



# Searching Directories

You can use the Directory Editor Search tab to search for objects in the directory installation.

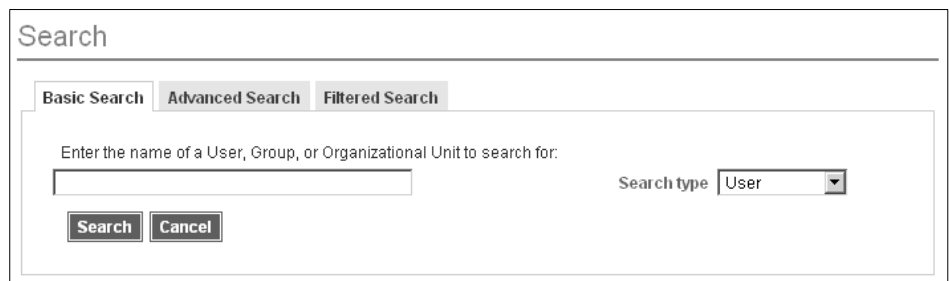
This chapter explains how to configure and execute three different types of searches from the Directory Editor Search page. The information is organized as follows:

- “Executing a Basic Search” on page 115
- “Executing an Advanced Search” on page 118
- “Executing a Filtered Search” on page 124

## Executing a Basic Search

Use the Basic Search tab (the default page) to search for users, groups, or organizational units.

**Figure 6-1** Search Results



The screenshot shows a window titled "Search". Inside, there are three tabs: "Basic Search", "Advanced Search", and "Filtered Search". The "Basic Search" tab is selected. Below the tabs, there is a text input field with the placeholder text "Enter the name of a User, Group, or Organizational Unit to search for:". To the right of the input field is a "Search type" dropdown menu with "User" selected. At the bottom of the input area are two buttons: "Search" and "Cancel".

To perform a basic search, use the following steps:

1. Enter the name of a user, a group, or organizational unit into the text field provided.  
  
For example, type **manager** for a list of all objects that have “manager” in their name.
2. Select a Search Type from the drop-down menu.
3. Click the Search button or press Return (Enter) on your keyboard.

The search results will display in table-format at the bottom of the Search page.

---

**NOTE** By default, Directory Editor will return a maximum of 100 results. If Directory Editor finds more than 100 entries, a note will display above the Results table.

If there are too many items returned, you can narrow the search criteria. To change the default limit for the number of objects displayed, select Configure > Search Options. When the Search Configuration page is displayed, change the Number of Results value.

---

For example, the following figure shows the result of searching for *manager*:

**Figure 6-2** Search Results

The screenshot displays the 'Search' interface. At the top, there are three tabs: 'Basic Search' (selected), 'Advanced Search', and 'Filtered Search'. Below the tabs, a text field contains 'manager' and a 'Search type' dropdown menu is set to 'All'. 'Search' and 'Cancel' buttons are present. Below the search area, there are buttons for 'Create', 'Delete', 'Rename', 'Enable', and 'Disable'. The 'Results' section shows a table with columns for 'Name', 'First Name', and 'Last Name'. The table lists four entries: 'Accounting Managers', 'HR Managers', 'QA Managers', and 'PD Managers', each with an unchecked checkbox in the first column.

	Name	First Name	Last Name
<input type="checkbox"/>	Accounting Managers		
<input type="checkbox"/>	HR Managers		
<input type="checkbox"/>	QA Managers		
<input type="checkbox"/>	PD Managers		



A bank of buttons also displays just above the Results table. You use these buttons in conjunction with the table.

Note that each entry in the Results table is preceded by a checkbox. You can enable one or more checkboxes to select those rows for an action and then click one of the following buttons:

- **Create:** Opens a Create page so you can add a new user, group, or organizational unit. (See Chapter 4, “Creating and Editing Objects.”)
- **Delete:** Opens an Delete page, where you must confirm whether to delete the object. Click Delete to confirm the action or click Cancel to terminate the action.
- **Rename:** Opens a Rename page so you can specify a new name for that object and for children of the object. (See Chapter 4, “Creating and Editing Objects.”)
- **Enable:** Opens an Enable page, where you must confirm whether to enable the object. Click Enable to confirm the action or click Cancel to terminate the action.
- **Disable:** Opens an Disable page, where you must confirm whether to disable the object. Click Disable to confirm the action or click Cancel to terminate the action.

---

<b>NOTE</b>	You can also click any linked entry in the table's Name column, to open the Edit an Object page where you can edit the object's attributes. See Chapter 4, “Creating and Editing Objects.” for more information about editing objects.
-------------	--

---

# Executing an Advanced Search

Select the Advanced Search tab to search for objects by specifying search conditions, which can be combined with **and** operations.

**Figure 6-3** Advanced Search Tab

Search

Basic Search Advanced Search Filtered Search

Search type: User

Base Context:  Browse

Search Conditions

	Attribute	Condition	Value
<input type="checkbox"/>	audio	contains	

Remove Selected Attribute(s) Add Attribute

Search Cancel

To configure and execute an advanced search, use the following steps:

1. Select one of the following object types from the Search type menu:
  - **User**
  - **Group**
  - **Organizational Unit**
  - **Other**

When you select the Other option, the Object Classes selection tool displays (similar to the one shown in the following figure).

**Figure 6-4** Selecting Object Classes

Search

Basic Search **Advanced Search** Filtered Search

Search type: Other

Available Object Classes

Object Class

- account
- alias
- applicationEntity
- applicationProcess
- bootableDevice
- cacheObject
- calEntry
- certificationAuthority
- certificationAuthority-V2

Selected Object Classes

Base Context:  **Browse**

Search Conditions

	Attribute	Condition	Value
<input type="checkbox"/>	audio	contains	

**Remove Selected Attribute(s)** **Add Attribute**

**Search** **Cancel**

- You can select object classes from the Available Object Classes list and click the Add button **>** to move the object classes to the Selected Object Classes list.
  - Click Add All **>>** to move all available object classes to the Selected Object Classes list.
  - Click Remove All **<<** to move all object classes from the Selected Object Classes list back to the Available Object Classes list.
  - Select object classes from the Selected Object Classes list and click Remove **<** to move the object classes from the Selected list back to the Available Object Classes list.
2. Enter a Base Context in the text box provided (or click Browse to search the organizational units in the directory).

The Search will return results from items in this container and “contained” containers.

3. Use the Search Conditions table to configure your search criteria, as follows:
  - a. Click the drop-down button in the Attribute column to select an attribute from the list.

---

**NOTE**

- The attributes in this list are determined by the object type specified in Step 1.
    - To add rows to the Search Conditions table, click the Add Attribute button.
    - To Remove rows from the Search Conditions table, enable the checkbox in the leftmost column and then click the Remove Selected Attribute(s) button.
  - You cannot remove all rows in this table. If you try to remove the only remaining row, an error will result.
- 

- b. Click the drop-down button in the Conditions column and select one of the following conditions from the list:
    - **contains:** Search for attributes that contain the specified attribute value.
    - **equals:** Search for attributes that match the specified attribute value exactly.
    - **does not equal:** Search for attributes that *do not* match the specified attribute value.

4. Enter a value to search for into the Value column of the table.

---

**NOTE**

The Attribute, Condition, and Value entries are “OR’ed” together to create the search criteria.

---

5. Click the Search button.

The search results will display in table-format just below the text field.

---

<b>NOTE</b>	<p>By default, Directory Editor will return a maximum of 100 results. If Directory Editor finds more than 100 entries, a note will display above the Results table.</p> <p>If there are too many items returned, you can narrow the search criteria. To change the default limit for the number of objects displayed, select Configure &gt; Search Options. When the Search Configuration page is displayed, change the Number of Results value.</p>
-------------	--

---

The search results will display in table-format at the bottom of the Search page.

As with a Basic Search, a bank of buttons (Create, Delete, Rename, Enable, and Disable) also displays just above the Results table. Use these buttons in conjunction with the table as described in Step 2 on page 119.

---

<b>NOTE</b>	<p>Click any of the linked entries in the Name column, and the Edit an Object page displays so you can edit the object's attributes. (See Chapter 4, "Creating and Editing Objects" for information about editing objects.)</p>
-------------	---

---

Following are several examples of an Advanced Search.

**Example 1:** Someone reports a "fender-bender" in the parking lot and you want to notify the car owner. You could specify the following parameters to identify the owner:

- **Search type:** User
- **Base Context:** dc=example,dc=sun,dc=com
- **Search Conditions:**

**Table 6-1** Example Search Parameters

Attribute	Condition	Value
carlicense	equals	<license_number>

---

**NOTE** The Attribute, Condition, and Value entries are “OR’ed” together to create the search criteria.

The following figure shows the result of searching for the license number:

**Figure 6-5** Example 1 Search Results

Basic SearchAdvanced SearchFiltered Search

Search typeUser

Base Context:dc=example,dc=sun,dc=comBrowse

Search Conditions

	Attribute	Condition	Value
<input type="checkbox"/>	carLicense	equals	kF1111

Remove Selected Attribute(s)Add Attribute

SearchCancel

CreateDeleteRenameEnableDisable

Results

<input type="checkbox"/>	Name	First Name	Last Name
<input type="checkbox"/>	K. C. Jones	K. C.	Jones

**Example 2:** You want to send an email to all Marketing managers. You could specify the following parameters to identify these managers:

- **Search type:** User
- **Base Context:** dc=example,dc=sun,dc=com

122 Sun Java™ System Directory Editor 1 2004Q4 • Installation and Configuration Guide

- **Search Conditions:**

**Table 6-2** Example Search Parameters

Attribute	Condition	Value
businessCategory	equals	Marketing
employeeType	equals	Manager

- NOTE**
- The Attribute, Condition, and Value entries are “OR’ed” together to create the search criteria.
  - Remember to click the Add Attribute button to add rows (additional attributes) to the Search Conditions table.

The following figure shows the result of searching for Marketing managers:

**Figure 6-6** Example 2 Search Results

Basic SearchAdvanced SearchFiltered Search

Search type User

Base Context: dc=example,dc=sun,dc=com Browse

Search Conditions

	Attribute	Condition	Value
<input type="checkbox"/>	<span>businessCategory</span>	<span>equals</span>	<span>Marketing</span>
<input type="checkbox"/>	<span>employeeType</span>	<span>contains</span>	<span>Manager</span>

Remove Selected Attribute(s) Add Attribute

Search Cancel

CreateDeleteRenameEnableDisable

Results

<input type="checkbox"/>	Name	First Name	Last Name
<input type="checkbox"/>	ou=People,dc=central,dc=sun,dc=com		
<input type="checkbox"/>	Kenneth Suter	Kenneth	Suter
<input type="checkbox"/>	Bob Jones	Bob	Jones

# Executing a Filtered Search

Select the Filtered Search tab to specify free-form query strings to search for objects in the directory.

**Figure 6-7** Filtered Search tab



The default search string on this tab is `(&(objectclass=Person)(cn=*))` or a string that represents the current basic or advanced search specified on the Basic or Advanced Search tabs.

You can use the default search string or specify your own query in the Search Filter text field. For example, if you want to search for all groups in the directory, type the following text in the Search Filter text field and click Search:

```
(|(ObjectClass=groupofnames)(ObjectClass=groupofUniqueNames))
```

The following figure shows the result of the filtered search.



**Figure 6-8** Example Filtered Search Results

Basic SearchAdvanced SearchFiltered Search

Search Filter

((ObjectClass=groupofnames)(ObjectClass=groupofUniqueNames))

Search

Cancel

CreateDeleteRenameEnableDisable

Results

<input type="checkbox"/>	Name	First Name	Last Name
<input type="checkbox"/>	Directory Administrators		
<input type="checkbox"/>	Accounting Managers		
<input type="checkbox"/>	HR Managers		
<input type="checkbox"/>	QA Managers		
<input type="checkbox"/>	PD Managers		
<input type="checkbox"/>	admin		

As with the other search methods, the search results will display in table-format at the bottom of the Search page.

**NOTE**

By default, Directory Editor will return a maximum of 100 results. If Directory Editor finds more than 100 entries, a note will display above the Results table.

If there are too many items returned, you can narrow the search criteria. To change the default limit for the number of objects displayed, select **Configure > Search Options**. When the Search Configuration page is displayed, change the **Number of Results** value.

In addition, a bank of buttons (Create, Delete, Rename, Enable, and Disable) displays just above the Results table. Use these buttons in conjunction with the table as described in step 2 on page 6-82.

**NOTE**

Click any of the linked entries in the Name column, and the Edit an Object page displays so you can edit the object’s attributes. (See Chapter 4, “Creating and Editing Objects” for information about editing objects.)

# Editing the Search Page Configuration

The procedures in this section describe how to change the appearance and content of your Search page. The information is organized as follows:

- “Accessing the Search Configuration Page” on page 126
- “Specifying Default Settings” on page 127
- “Configuring the Results Table” on page 128
- “Specifying Search Types” on page 130

## Accessing the Search Configuration Page

To open the Search Configuration page, select the Configure > Search Options. The Search Configuration page is displayed as follows:

**Figure 6-9** Search Configuration Page

This page contains three tabs:

- **Settings:** Use this tab to configure a default search filter for the Search page.
- **Results Table:** Use this tab to configure a results table for the Search page.
- **Search Types:** Use this tab to specify the type of searches available for the Search page.

## Specifying Default Settings

Use the following procedure to configure a default search filter for the Search page.

1. Select the Settings tab (see Figure 6-9).
2. Modify the default filter shown in the Default Search Filter text box or enter a new filter.

---

**CAUTION** Improper use of a search filter can impact both Directory Editor and Directory Server performance. For detailed information about constructing filters, refer to the *Sun Java™ System Directory Server 5 2004Q2 Administration Guide*.

---

3. Modify or enter a new display name in the Display Name text box.

You may want Directory Editor to return results in the same way your company displays data. For example; `cn`, `uid`, and so forth.

---

**NOTE** The dollar sign (\$) that precedes this value denotes the start of an attribute name.

---

4. You can click Save to apply your changes (or click Cancel and return to the Home page).

---

**NOTE** You do not have to Save your changes on each page — you can wait until you are finished working with all of the tabs and then save your changes to the directory.

---

From here, you can select another tab to continue configuring your Search page.

## Configuring the Results Table

Use the following procedure to configure a results table for the Search page.

1. Select the Results Table tab.

**Figure 6-10** Results Table Page

Search Configuration Save Cancel

Settings Results Table Search Types

Available Attributes

Results Columns

- abstract
- accountUnlockTime
- aci
- aciRights
- aciRightsInfo
- adminRole
- adminUrl
- administratorContactInfo
- aliasedObjectName

Selected Attributes




- givenName
- sn




Display Names

Attribute Name	Display Name
dn	Name
givenName	First Name
sn	Last Name

Column Width

Number of results

2. Use the Results Columns selection tool to add attributes to or remove attributes from the Selected Attributes list, as follows:
  - Select one or more attributes from the Available Attributes list and click the  button to move them to the Selected Attributes list. (Press your Shift key and click on items in the list to select multiple attributes.)
  - Click the  button to move all available attributes to the Selected Attributes list.
  - Click the  button to move all attributes from the Selected Attributes list back to the Available Attributes list.

- Select one or more attributes from the Selected Attributes list and click the  button to move them back to the Available Attributes list.
  - Use the  (move up) and  (move down) buttons to change the order of attributes in the Selected Attributes list.
3. By default, Directory Editor populates the Display Names table with attribute names and display names.
- You can replace the default names in the Display Names column by typing directly into the text boxes provided.
4. Enter a value in the Use the Length of each field in the results table text box to specify the maximum character length for fields results table.
5. Enter a value in the Number of Results text box to specify the maximum number of search results (how many lines) to display in the Results table.

---

<b>NOTE</b>	If the number of search results exceed the maximum specified for the Results table, a warning message is displayed.
-------------	---

---

## Specifying Search Types

You can use the Search Types page to modify the kind of search Directory Editor performs by default. Use the following procedure to specify the type of searches available for the Search page.

1. Select the Search Types tab.

**Figure 6-11** Search Types Page

Settings Results Table Search Types

Default Search Type:

Name	Value
<input type="checkbox"/> User	<div>Available Objectclasses</div> <div> account alias applicationEntity applicationProcess bootableDevice cacheObject calEntry certificationAuthority certificationAuthority-V2 </div> <div>Selected Objectclasses</div> <div> person organizationalPerson inetOrgPerson </div>
<input type="checkbox"/> Organization	<div>Available Objectclasses</div> <div> account alias applicationEntity applicationProcess bootableDevice cacheObject calEntry certificationAuthority certificationAuthority-V2 </div> <div>Selected Objectclasses</div> <div> organizationalUnit organization </div>
<input type="checkbox"/> Group	<div>Available Objectclasses</div> <div> account alias applicationEntity applicationProcess bootableDevice cacheObject calEntry certificationAuthority certificationAuthority-V2 </div> <div>Selected Objectclasses</div> <div> groupOfNames groupOfUniqueNames </div>

Remove Selected Types(s) Add Type

2. If necessary, change the Default Search Type name by entering a new value in the text box provided.
3. Use the table to change the search type names and associated attributes that are available in the Search Type menu list.

- a. To change the name of a search type, delete the old name and enter a new value.

For example, you might want to change Organization to Offices.

- b. Use the Objectclasses selection tools in the Value column to create a list of valid values for that search type.

For example, you might want to specify the following values for the User type:

- Person
- organizationalPerson
- inetorgPerson (*default*)
- engineeringPerson

---

**NOTE** If you create a custom user class, you should add the user as an object class to the user object.

---

4. When you are finished, click the Save button to save your changes and return to the Home page (or click Cancel to undo your changes).

---

**NOTE** You do not have to Save your changes on each page — you can wait until you are finished working with all of the tabs and then save your changes to the directory.

---

## Adding Search Types

To add search types to the table, use the following procedure:

1. Click the Add Type button located beneath the table.
2. When Directory Editor adds a new row to the table, enter a name for the search type in the text box provided in the Name column.
3. Use the Objectclasses selection tool in the Value column to create a list of valid values for the new search type.

For example, you might want to specify a Location search type with the following objectclasses:

- country
  - friendlyCountry
  - room
4. When you are finished, click Save to save your changes and return to the Home page (or click Cancel to undo your changes).

## Removing Search Types

To remove objects from the Search Types menu list,

1. Enable the checkbox next to the search type you want to delete.
2. Click the Remove Selected Types button located beneath the table.  
Directory Editor immediately removes that row from the table.
3. Click Save to save your changes and return to the Home page, or click Cancel to undo your changes and return to the Home page.



# Configuring Directory Editor

Use the information provided in this chapter to configure your Directory Editor application to user control access to applications and application components, and to define relative distinguished names (RDN) configurations. This chapter is organized as follows:

- “Controlling User Access” on page 133
- “Working with Naming Attributes” on page 142

## Controlling User Access

*Authentication* and *Authorization* are terms used to describe methodologies for controlling access to applications or application components.

- **Authentication** is the process by which an application challenges a client to supply credentials — typically providing a user name and password through a log-in page. Based on these credentials, the application determines whether a user (or a different client, such as another application) can use the application.
- **Authorization** assumes that a client has already been authenticated and is in the process of determining whether a client can use a *component* of the application. Authorization takes a more fine-grained approach to access control.

You can use the Directory Editor Authorization page to authenticate clients *and* control access to Directory Editor components using fine-grained authorization

---

**NOTE** By default, you must have the Manager role to access the Authorization page. For more information about roles, see “Understanding Roles” on page 134.

---

To understand authorization, you must understand the terms *role*, *principal*, and *accessible components*, which are described in the following sections:

- “Understanding Roles” on page 134
- “Understanding Principals” on page 135
- “Understanding Accessible Components” on page 135

## Understanding Roles

A *role* describe a user's function within the enterprise hosting the Directory Editor. For example, your organization may have CEO, help desk administrator, and janitor roles.

Roles determine with which parts of Directory Editor you can interact. By default, Directory Editor is pre-configured with two roles:

- **Manager:** This role has access to all of Directory Editor's features and functionality.
- **Default:** This role has more limited capabilities within the application. For example, you can browse directory data but you cannot edit that data.

Directory Editor enables you to add roles that support interactions appropriate for your enterprise. For example, you might want to define a role that allows users to edit directory data but does not allow them to access the Configure tab.

Every role is associated with a set of *principals* that assume the role (see the next section, “Understanding Principals”). For Directory Editor, there is a single group in the directory server (called the *Manager Group*) that serves as the principal corresponding to the Manager role.

You use the Managed Directory page to specify the Manager Group at configuration time (also available after configuration by selecting Configure > Managed Directory). So, if you have a particular user that should have full access to all Directory Editor functions, make that user's DN a member of the Manager Group.

## Understanding Principals

A *principal* represents an entity (such as an individual, corporation, or login ID). The term *subject* is used to describe entities (typically human users). Subjects can be represented by multiple, differing principals — just as people can be represented by their credit card number to banks and by their UNIX account name to system administrators. The credit card numbers and UNIX account names are principals in this case.

Because Directory Editor is focused on directory data management, its principals are all represented using the following directory objects:

- DN (distinguished names)
- Directory Server groups
- Directory Server roles

A user entering a DN in the log-in page can be represented by several different principals, depending on the data in the directory. For example, if the user's account ID happens to be a member of a specific group, that user will be represented by the account ID's DN and by the DN of the group to which the DN belongs.

In Directory Editor, objects representing the user (or subject) are stored in the HTTP session. After the user enters an account ID and a password on the Directory Editor's log-in page, Directory Editor populates the subject with all of the various principals (person entries, groups, and roles) associated with that account ID.

## Understanding Accessible Components

*Accessible components* are the set of internal Directory Editor components that are accessible by the users. Users may not know they are interacting with certain accessible components; consequently, administrators must understand which components are required to complete certain tasks.

Currently, all accessible components are either *actions* or *views*.

- **Actions** are represented by a string such as `/Edit`.
- **Views** are represented by a string is similar to `.nav_tab.Edit`.

---

**NOTE** Typically, each view is represented using both an *unselected* representation and a *selected* representation, such as `.nav_tab_selected.Edit`.

---

The rest of this section provides instructions for defining, editing, and deleting roles. The section is organized as follows:

- “Accessing the Authorization Page” on page 136
- “Defining Directory Editor Roles” on page 137
- “Editing Roles” on page 140
- “Deleting Roles” on page 141

## Accessing the Authorization Page

To open the Authorization page, select the Configure tab and then select the Authorization tab. The Authorization page is displayed as follows:

**Figure 7-1** Authorization Page

HomeListCreateSearchConfigure

FormsStartupManaged DirectorySearch ConfigurationRDN ConfigurationAuthorizationImport/Export

Authorization

Create RoleEdit Selected RoleDelete Selected Role(s)

	Role	Principals	Accessible Components
<input type="checkbox"/>	Manager	cn=Directory Administrators,dc=central,dc=sun,dc=com	*
<input type="checkbox"/>	Default	*	.nav_tab_selected.Home .nav_tab.List /List .nav_tab_selected.Edit .nav_tab.Search /Search .nav_tab.Edit .nav_tab_selected.Search .nav_tab_selected.List /Edit .nav_tab.Home /Home

Create RoleEdit Selected RoleDelete Selected Role(s)

This page consists of the following features:

- **Role table:** By default, this table contains the Manager and Default roles.
- **Create Roles button:** Click this button to create additional roles appropriate for your enterprise. (See “Defining Directory Editor Roles” on page 137.)
- **Edit Selected Roles button:** Click this button to change properties, principals, and accessible resources associated with a particular role. (See “Editing Roles” on page 140.)
- **Delete Selected Roles button:** Click this button to remove roles. (See “Deleting Roles” on page 141.)

## Defining Directory Editor Roles

Before you can define new Directory Editor roles for your enterprise, you must decide which tasks a common set of users must perform. For example, all of your help desk administrators must have write access to directory data.

After you have identified these tasks, use the following steps to create a new role:

1. Select Configure > Authorization.
2. When the Authorization page is displayed (see Figure 7-1 on page 136), click the Create Role button.
3. On the Create Role page, enter a meaningful name into the Role Name text box. For example, Help Desk Administrators.

**Figure 7-2** Role Properties Tab

**Create Role:**

Role Properties

Principals

Accessible Resources

Role Name:

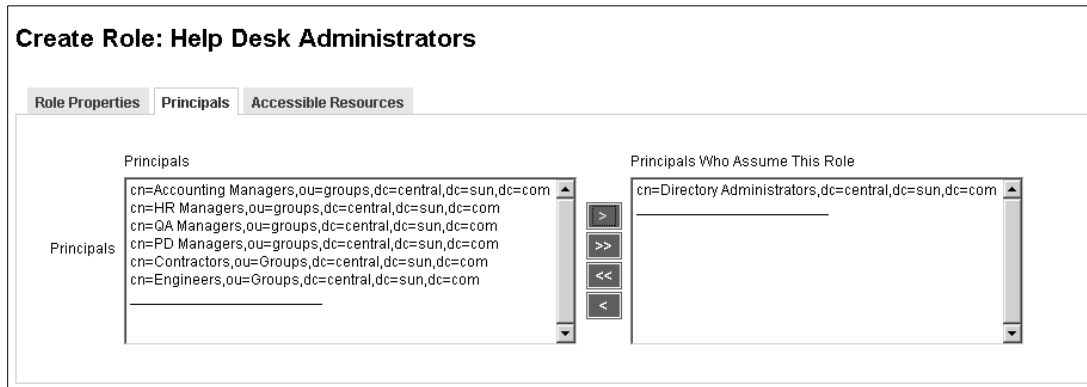
4. Select the Principals tab and use the Principals selection tool to specify the set of principals you want to assume the new role.





---

**NOTE** At this time, you can only assign groups as principals.

---

**Figure 7-3** Principals Tab



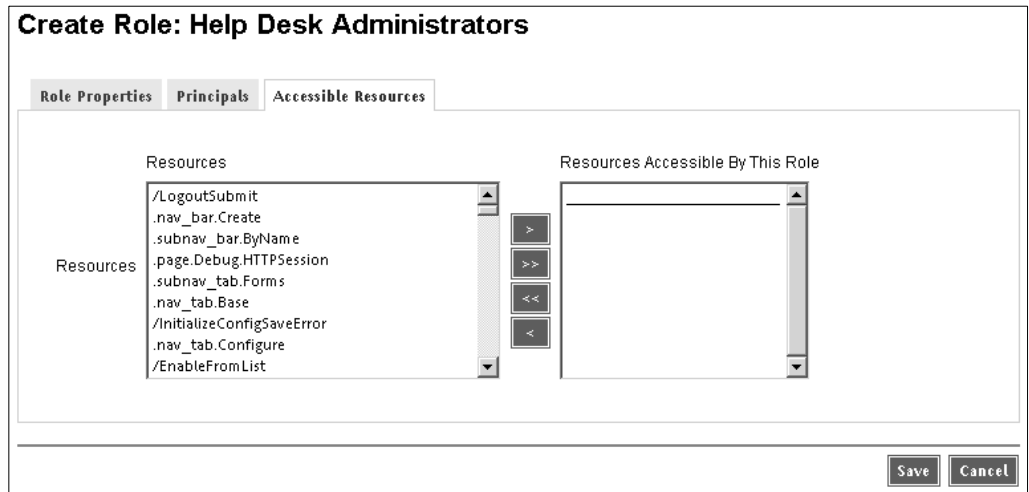
- You can select one or more principals from the Principals list and click the  button to move them to the Principals Who Assume This Role list. (Press your Shift key and click on items in the list to select multiple principals.)
- Click the  button to move all principals to the Principals Who Assume This Role list.
- Click the  button to move all principals from the Principals Who Assume This Role list back to the Principals list.
- Select principals from the Principals Who Assume This Role list and click the  button to move them back to the Principals list.





For example, you might assign the Directory Administrators to this role.

5. Select the Accessible Resources tab (Figure 7-4) to specify the set of components that can be accessed by the new role.

**NOTE** For a complete list of available resources; including their type, category, and a description see Appendix B, “Resources for Role Configuration.”

**Figure 7-4** Accessible Resources Tab



- o Select one or more resources from the Resources list and click the  button to move them to the Resources Accessible By This Role list. (Press your Shift key and click on items in the list to select multiple resources.)
- o Click the  button to move all resources to the Resources Accessible By This Role list.
- o Click the  button to move all resources from the Resources Accessible By This Role list back to the Resources list.
- o Select resources from the Resources Accessible By This Role list and click the  button to move them back to the Resources list.

For example, for a Help Desk Administrator, you might want to assign all the Debug-related resources to this role.

6. Click Save to save the new role and to add it to the Roles table (or click Cancel to return to the Authorization page without saving your changes).

Figure 7-5 shows the updated Roles table.

**Figure 7-5**      New Role Added to the Roles Table

	Role	Principals	Accessible Components
<input type="checkbox"/>	Help Desk Administrators	cn=Directory Administrators,dc=central,dc=sun,dc=com	.page.Debug.HTTPSession .subnav_tab.Forms /Help /InitializeConfigSaveError .subnav_tab.Debug.Config .nav_tab.Configure .subnav_bar.Debug.Memory /Debug/Memory .subnav_tab_selected.NamingAttributeConfig /NamingAttributeConfig /Debug/Config .subnav_bar.Debug.Config .subnav_bar.Debug.HTTPSession .page.Debug /ExportConfiguration .page.Error .page.Debug.Memory .subnav_tab_selected.Debug.Memory .nav_tab_selected.Debug .subnav_tab.Debug.HTTPSession .nav_tab.Debug .nav_bar.Debug .page.Debug.Config /Debug .subnav_bar.Debug.Unselected .subnav_tab.Debug.Memory .subnav_tab_selected.Debug.Config .subnav_tab_selected.Debug.HTTPSession /Debug/HTTPSession

## Editing Roles

To edit selected authorization roles, use the following steps:

1. Select Configure > Authorization.
2. When the Authorization page is displayed, click the checkbox located next to the role you want to edit.

**Figure 7-6**      Click the Checkbox



3. Click the Edit Selected Roles button to open the Edit Roles page.



**Figure 7-7** Edit Role Page

4. The process for editing a role is the same as the process you used to create it. Review the instructions provided in “Defining Directory Editor Roles” on page 137 if necessary.
5. When you are finished, click Save (or click Cancel to return to the Authorization page without saving your changes).

## Deleting Roles

---

**NOTE** You cannot delete the Manager or the Default roles.

---

To delete selected authorization roles, use the following steps:

1. Select Configure > Authorization.
2. When the Authorization page is displayed, enable the checkbox(es) located next to the role(s) you want to delete.

**Figure 7-8** Click the Checkbox

<input checked="" type="checkbox"/>	Help Desk Administrators
-------------------------------------	--------------------------

3. Click the Delete Selected Role(s) button and Directory Editor will immediately remove the selected role(s) from the Roles table.

# Working with Naming Attributes

This section provides instructions for defining, editing, and deleting naming attributes. The section is organized as follows:

- “Accessing the Naming Attributes Page” on page 142
- “Creating New Object Class Naming Attribute Mappings” on page 144
- “Editing Naming Attributes” on page 146
- “Deleting Selected Naming Attributes” on page 147

## Accessing the Naming Attributes Page

To create new objects, Directory Editor must know how to construct DNs (distinguished names) for the new objects.

For example, if your customer wants to use `uid` (user ID) as the naming attribute for `inetOrgPerson` instead of `cn` you might specify the following DN for a newly created entry:

```
cn=Mike Miller,dc=example,dc=com
```

instead of:

```
uid=mmiller,dc=example,dc=com
```

Directory Editor ships with a small set of default naming attributes to use for object classes, so it is important that you modify these mappings to match the naming conventions used by your enterprise for naming directory objects. You must configure any object class that you add to the create page with naming attributes.

To access the Naming Attributes page,

1. Select the Configuration > Naming Attributes.

The Naming Attributes page is displayed as follows:

**Figure 7-9** Naming Attributes Page

Naming Attributes

New

Edit Selected

Delete Selected

	Object Class	Naming Attributes
<input type="checkbox"/>	mailgroupmember	mgmemMemberOfGroup
<input type="checkbox"/>	groupofnames	cn
<input type="checkbox"/>	locality	l
<input type="checkbox"/>	domain	dc
<input type="checkbox"/>	person	cn
<input type="checkbox"/>	mailgroup	cn
<input type="checkbox"/>	inetorgperson	uid
<input type="checkbox"/>	groupofuniquenames	cn
<input type="checkbox"/>	organizationalunit	ou
<input type="checkbox"/>	organization	o

New

Edit Selected

Delete Selected

This page consists of the following features:

- **Naming Attribute Mapping table:** By default, this table contains a small set of object classes and their naming attributes.
- **New button:** Click this button to add object classes (and define naming attributes for those object classes) to your enterprise. (See “Creating New Object Class Naming Attribute Mappings” on page 144.)
- **Edit Selected button:** Click this button to edit the naming attributes currently selected for the selected object class. (See “Editing Naming Attributes” on page 146.)
- **Delete Selected button:** Click this button to remove an object class and associated naming attributes from the table. (See “Deleting Selected Naming Attributes” on page 147.)

## Creating New Object Class Naming Attribute Mappings




After you have identified these tasks, use the following steps to create a new naming attribute:

2. Select **Configure > Naming Attributes**.
3. When the **Naming Attributes** page is displayed (see Figure 7-9 on page 143), click the **New** button.
4. A new **Naming Attributes** page is displayed (Figure 7-10). Select the object class from the **Object Class** menu.

**Figure 7-10** New Naming Attributes Page

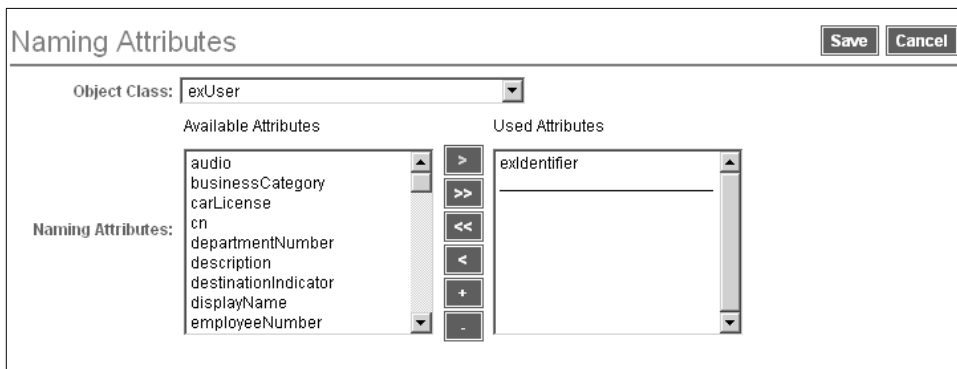
The screenshot shows a window titled "Naming Attributes" with "Save" and "Cancel" buttons in the top right corner. Inside the window, there is a label "Object Class:" followed by a dropdown menu currently showing "account". Below this, there are two lists: "Available Attributes" on the left and "Used Attributes" on the right. The "Available Attributes" list contains the following items: description, host, l, o, objectClass, ou, seeAlso, and uid. Between these two lists is a vertical column of buttons: a single right arrow (>), a double right arrow (>>), a double left arrow (<<), a single left arrow (<), a plus sign (+), and a minus sign (-). The "Used Attributes" list is currently empty.

5. Use the **Naming Attributes** selection tool to specify naming attributes for the new object class, as follows:
  - Select one or more naming attributes from the **Available Attributes** list and click the **>** button to move them to the **Used Attributes** list. (Press your Shift key and click on items in the list to select multiple naming attributes.)
  - Click the **>>** button to move all naming attributes to the **Used Attributes** list.
  - Click the **<<** button to move all naming attributes from the **Used Attributes** list back to the **Available Attributes** list.

- Select naming attributes from the Used Attributes list and click the  button to move them back to the Available Attributes list.
- Use the  (move up) and  (move down) buttons to change the order of attributes in the Used Attributes list

For example, you might specify a new objectclass called `exUser` for extending the default user object and have an attribute called `exIdentifier` as the naming attribute.

**Figure 7-11** New Object Class and Naming Attribute Added to the Table



6. Click Save to save the new object class and attribute(s) (or click Cancel to return to the Naming Attributes page without saving your changes).

Figure 7-12 shows the new entry added to the Object Class table.

**Figure 7-12** Updated Table

	Object Class	Naming Attributes
<input type="checkbox"/>	groupofnames	cn
<input type="checkbox"/>	exuser	exIdentifier
<input type="checkbox"/>	locality	l
<input type="checkbox"/>	domain	dc
<input type="checkbox"/>	person	cn
<input type="checkbox"/>	mailgroup	cn
<input type="checkbox"/>	inetorgperson	uid
<input type="checkbox"/>	groupofuniquenames	cn
<input type="checkbox"/>	organizationalunit	ou
<input type="checkbox"/>	organization	o

## Editing Naming Attributes

To edit selected naming attributes, use the following steps:

1. Select **Configure > Naming Attributes**.
2. When the **Naming Attributes** page is displayed, click the checkbox located next to the role you want to edit.

**Figure 7-13** Click the Checkbox

<input checked="" type="checkbox"/>	person	cn
-------------------------------------	--------	----

3. Click the **Edit Selected** button to open a new **Naming Attributes** page (similar to Figure 7-14).

**Figure 7-14** Editing the Naming Attributes

**Naming Attributes** [Save] [Cancel]

Object Class: person

Available Attributes

- description
- objectClass
- seeAlso
- sn
- telephoneNumber
- userPassword

Used Attributes

- cn

Naming Attributes:

Note that the **Object Class** menu is not available on this page. Instead, **Directory Editor** displays the selected object class name.

4. Use the **Naming Attributes** selection tool to add or remove naming attributes. Review the instructions provided in “**Creating New Object Class Naming Attribute Mappings**” on page 144 if necessary.
5. When you are finished, click **Save** (or click **Cancel** to return to the **Authorization** page without saving your changes).

## Deleting Selected Naming Attributes

To delete selected naming attributes, use the following steps:

1. Select Configure > Naming Attributes.
2. When the Naming Attributes page is displayed, enable the checkbox(es) located next to the object class(es) you want to delete.

**Figure 7-15** Click the Checkbox

	Object Class	Naming Attributes
<input type="checkbox"/>	groupofnames	cn
<input type="checkbox"/>	locality	l
<input type="checkbox"/>	domain	dc
<input type="checkbox"/>	person	cn
<input checked="" type="checkbox"/>	mailgroup	mgrpDeliverTo,owner
<input type="checkbox"/>	inetorgperson	cn
<input type="checkbox"/>	groupofuniquenames	cn
<input type="checkbox"/>	organizationalunit	ou
<input type="checkbox"/>	organization	o

3. Click the Delete Selected button and Directory Editor will immediately remove the selected object class(es) from the table.





# Backing Up and Restoring Configurations

This chapter describes how to back up your configurations to a file and how to restore configurations from a file using Directory Editor. The information is divided into the following sections:

- “Accessing the Backup/Restore Page” on page 150
- “Backing Up Your Configuration” on page 150
- “Restoring Your Configuration” on page 152

---

**CAUTION** You must use caution when backing up or restoring configurations. Directory Editor will re-initialize the application, and you will be required to log in again after a successful back up or restore operation.

---

# Accessing the Backup/Restore Page

To open the Backup/Restore page, select the Configure tab and then select the Backup/Restore tab. The Backup/Restore page is displayed as follows:

**Figure 8-1** Backup/Restore Page

Home | Browse | Create | Search | **Configure**

Forms | Startup | Managed Directory | Search Options | Naming Attributes | Authorization | **Backup/Restore**

## Backup/Restore

Backup and restore the configuration to and from XML.

**Backup Configuration to File**

⚠ Use this feature with caution as this application will be reinitialized.  
You will have to log in again after successfully restoring the configuration.

Configuration File:  **Browse...**

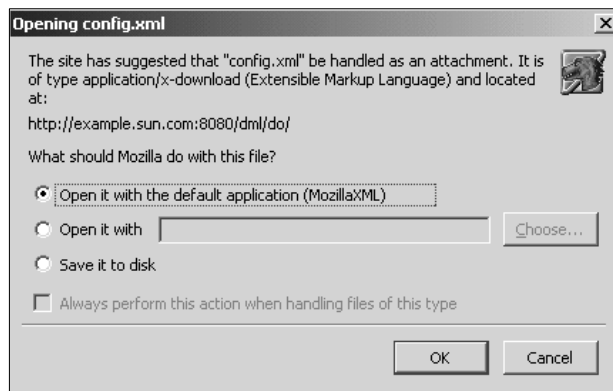
**Restore Configuration from File**

## Backing Up Your Configuration

The Directory Editor back-up feature enables you to save your current configuration to a file or edit your configuration file with an XML editing tool.

To back up your configuration, use the following procedure:

1. Click the Export a Configuration to File button.
2. When the Opening config.xml dialog box displays (Figure 8-2), enable one of the radio buttons to indicate how your browser should handle the file.

**Figure 8-2** Specifying How Your Browser Should Open the File

- Open it with the default application (for example, MozillaXML)
  - Open it with (You must enter an application name in the text field provided.)
  - Save it to disk (You must specify a file name and specify a folder in which to save the file.)
3. After making your selection, click OK to close the Opening config.xml dialog box.

You should now be able to open the configuration XML file in your browser. At this point, you can edit the file, and then import it using the instructions provided in the next section, or store the file for backup purposes.

# Restoring Your Configuration

To restore a configuration from a file, use the following procedure:

1. Type the configuration path and file name in the Configuration File text box, or click Browse to locate the file you want to restore.

This file should be an XML file you created using the Backup function (described in the previous section). You can import the entire configuration or just part of the configuration as long as the XML is well-formed and valid.

2. Click the Import a Configuration to File button.
  - If the import is successful the Log-In page is displayed and you must log in again.
  - If there are errors in your file, Directory Editor will prompt you to correct those errors before trying to import the file again.

# Removing the Software

Use the information provided in this chapter to remove Directory Editor from your Windows or UNIX system and how to undeploy Directory Editor from your application server. The information is organized as follows:

- “Removing the Software” on page 153
- “Undeploying Directory Editor from Your Application Server” on page 156

## Removing the Software

Use the instructions in one of the following sections to remove the Directory Editor software from your machine

- “Using the Directory Editor Uninstaller” on page 154
- “Uninstalling Manually From a Windows Machine” on page 154
- “Uninstalling Manually From a Unix Machine” on page 155

## Using the Directory Editor Uninstaller

Use the following steps to uninstall the software using the Directory Editor uninstaller:

1. Open a command window, and go to the directory where you installed Directory Editor.
2. Type the following command to launch the Uninstall wizard:  

```
java uninstall_Directory_Editor
```
3. The Welcome screen displays. Read the information provided, and then click Next.
4. When the Ready to Uninstall screen is displayed, click Uninstall Now.  
When the uninstallation is complete, the Uninstallation Summary screen displays to report status.
5. You can click the Details button for more information, or click Close to exit the Uninstall Wizard.
6. To remove any links or references to the Directory Editor software, you must undeploy Directory Editor from your application server. Use the instructions provided in “Undeploying Directory Editor from Your Application Server” on page 156.

## Uninstalling Manually From a Windows Machine

Use the following steps to remove Directory Editor from a Windows machine:

1. Stop your application server.
2. Locate and open the Directory Editor webapps directory in your application server's directory.
3. Delete the dml.war file and the dml directory.
4. If necessary, return your application server's \bin directory and double-click startup.bat to restart the application.

5. Remove all Directory Editor configuration information from the Sun Java System Directory Server or Sun ONE Directory Server, which (by default) is located under `Services:DML`.
  - a. Log into Directory Server.
  - b. Select the Directory tab.
  - c. Click the Services node in the left pane.
  - d. Locate and click the DML node under Services.
  - e. Select Edit > Delete from the Directory Server menu bar (or right-click on the DML node and select Delete from the pop-up menu) to remove the DML node and all of its children.
6. To remove any links or references to the Directory Editor software, you must undeploy Directory Editor from your application server. Use the instructions provided in “Undeploying Directory Editor from Your Application Server” on page 156.

## Uninstalling Manually From a Unix Machine

Use the following steps to remove Directory Editor from a Unix machine:

1. Stop your application server.
2. Locate and open the `/bin` directory.
3. Double-click `shutdown.sh` to stop the Tomcat application.
4. Locate and open the `/webapps` directory.
5. Delete the `dml.war` file and the `dml` directory.
6. If necessary, return to the `/bin` directory and double-click `startup.sh` to restart the Tomcat application.
7. Remove configuration information from the Directory Server, which (by default) is located under `Services:DML`.
8. To remove any links or references to the Directory Editor software, you must undeploy Directory Editor from your application server. Use the instructions provided in “Undeploying Directory Editor from Your Application Server” on page 156.

# Undeploying Directory Editor from Your Application Server

This section provides instructions for undeploying from the following application servers:

- “From Sun Application Servers” on page 156
- “From Tomcat Application Servers” on page 156
- “From WebLogic Application Servers” on page 157
- “From WebSphere Application Servers” on page 157

## From Sun Application Servers

Use the following instructions to undeploy Directory Editor from a Sun Application Server:

1. Open the Sun ONE Administration Console.
2. In the left pane, open the Web Apps folder.
3. Locate the `dml` node (or the name used when you deployed Directory Editor) and enable the checkbox next to that item.
4. Click the Undeploy button, and when you are prompted to confirm the action, click OK.

## From Tomcat Application Servers

Use one of following methods to undeploy Directory Editor from a Tomcat Application Server:

- “Using the Tomcat Application Manager” on page 157
- “Undeploying Manually” on page 157



## Using the Tomcat Application Manager

Use the following instructions to undeploy Directory Editor using the Tomcat Application Manager:

1. Open the Tomcat Application Manager.
2. In the Applications table, click the `dml` link and then click the `Undeploy` link in the Command column.

## Undeploying Manually

To undeploy Directory Editor manually from your Tomcat application server, use the instructions for uninstalling manually provided in “Removing the Software” on page 153.

## From WebLogic Application Servers

Use the following instructions to undeploy Directory Editor from a WebLogic Application Server:

1. Open the WebSphere Console.
2. In the left pane, open the Deployments and Web Applications nodes.
3. Select `dml` in the table and then click the trash can icon.
4. When you are prompted to confirm the action, click Yes.

## From WebSphere Application Servers

Use the following instructions to undeploy Directory Editor from a WebSphere Application Server:

1. Open the WebSphere Administration Console.
2. Click the Enterprise Application link to display a list of applications.
3. Locate the `dml` node and enable the checkbox next to that item.
4. Click the Stop button to stop the application.
5. Enable the `dml` checkbox again, and then click the Uninstall button.



# Error Logging and Troubleshooting

This chapter explains how to troubleshoot problems you may encounter as you work with Directory Editor. The information is organized into the following chapters:

- “Before You Call Technical Support” on page 159
- “Obtaining Information About the Problem” on page 160
- “Common Problems and Solutions” on page 165

## Before You Call Technical Support

Before you call technical support, please get as much information as possible about the problem you are experiencing.

1. Using instructions provided in “Obtaining Information About the Problem” on page 160, increase logging levels where appropriate and then try to re-create the problem.
2. Save all logs and error information, and be ready to provide this information to the technical support staff.
3. Be prepared to answer questions about your Directory Editor installation. For example,
  - Which application server and version are you running?
  - Which JDK version are you running?
  - Which operating system is running the application server?
  - Which browser version and operating system are you using to access Directory Editor?

- Which user was logged in when the problem occurred?
- How are the Startup Properties and Managed Directory configured?
- Which Directory Editor build are you running?

---

<b>TIP</b>	To determine the build number, use one of the following methods: <ul style="list-style-type: none"><li>• Hover your mouse cursor over the Version string at the top of the Directory Editor page. A tooltip message will display with the actual build number or (depending on your browser) the build number will be displayed as a message in the status bar (usually at the bottom of the browser window).</li><li>• Right-click in the Directory Editor page and select View Page Source. When the source page is displayed, select Edit &gt; Find in this page to search for "Build number."</li></ul>
------------	---

---

## Obtaining Information About the Problem

The first step to solving problems in Directory Editor is to figure out what went wrong. You can use information provided in this section to help diagnose and solve problems.

### Information in the Page

If an error message is displayed on a web page, the page may contain more detailed information about the problem. To locate this information, right-click on the page and when a pop-up menu displays, select View Page Source.

Inspect the page, and if it contains a Java stack trace, select File > Save Page As from the browser's menubar to save the page.

## Information in the Logs

There are various log files that may contain relevant information about the problem. These log files are described in the following sections:

- “Directory Editor Log” on page 161
- “Identity Manager Log” on page 162
- “Directory Server Log” on page 163
- “Application Server Logs” on page 164

### Directory Editor Log

The Directory Editor log contains log statements generated from Directory Editor.

Possible log levels include:

- **FATAL:** Fatal error messages (*least detail*)
- **ERROR:** Non-fatal error messages
- **WARN:** Potentially harmful situations
- **INFO:** Informational messages highlighting application state
- **DEBUG:** Fine-grained informational events
- **TRACE:** Maximum detail

You can set these logging levels per Java class or package. For example, to get detailed messages about the operations being sent to the LDAP server, you can add the following line to the file:

```
log4j.logger.com.sun.dml.resource.LDAPConnector=TRACE
```

The `log4j.properties` file shipped with Directory Editor has more documentation about how to set logging levels, and some commonly used levels.

You can configure the log’s location and the logging levels in the `log4j.properties` file, which is located at

```
<application root>/WEB-INF/classes/log4j.properties
```

To change the log file location and logging levels, edit the `log4j.properties` file in a text editor. You must restart Directory Editor after saving your changes for the changes to take effect.

You can change the loggers in a log file to use a different appender. For example, the default `log4j.properties` file has an appender defined for a rolling log file that rolls with a maximum file size. To use this appender, you must change the `log4j.rootLogger` file to use the name of the new appender (for example, `log4j.rootLogger=WARN,rollingfile`).

For more information about configuring appenders and logging levels in the `log4j.properties` file, see the Log4j documentation provided at the following location:

<http://logging.apache.org/log4j/docs/>

## Identity Manager Log

Some of Directory Editor's core components are provided by Sun Java™ System Identity Manager, which also supports logging. You can configure this product's logging using the Java system properties or the following file:

`<application root>/config/Waveset.properties`

Logging levels are configurable on a per package, per class, or per method basis. By default, the Identity Manager log goes to the `WSTrace.log` file located in the `<application root>/config/` directory. You can change this file by defining the `trace.destination` and `trace.Logging` levels as follows:

- **0:** Disable tracing
- **1:** Entry and exit of public methods, plus major exceptions
- **2:** Entry and exit of all methods
- **3:** Significant informational displays (such as variable values that control flow) that occur only once per method invocation
- **4:** Informational displays that occur *n* times per method invocation

Here as an example of tracing configuration in the `Waveset.properties` file:

### Code Example 10-1 Sample Identity Manager Trace Configuration

```
trace.enabled=true
trace.level=0
trace.level.com.waveset.repository.ServerRepository=3
trace.level.com.waveset.repository.ServerRepository#init=0
trace.destination=STDOUT
# trace.destination=file
# trace.file=/var/log/WSTrace.log
```

You can also turn on tracing for forms and XPRESS by setting the `form.trace` and `xpress.trace` properties to `true` in the `Waveset.properties` file. You must restart the application server for any Identity Manager logging changes to take effect. For more information, see the *Sun Java™ System Identity Manager Technical Deployment Guide*.

## Directory Server Log

Sun Java™ System Directory Server and Sun ONE Directory Server both can log directory access and detailed auditing messages.

The audit log contains detailed information about changes to the directory — such as creates, edits, deletes, and renames. The log entries provide information about:

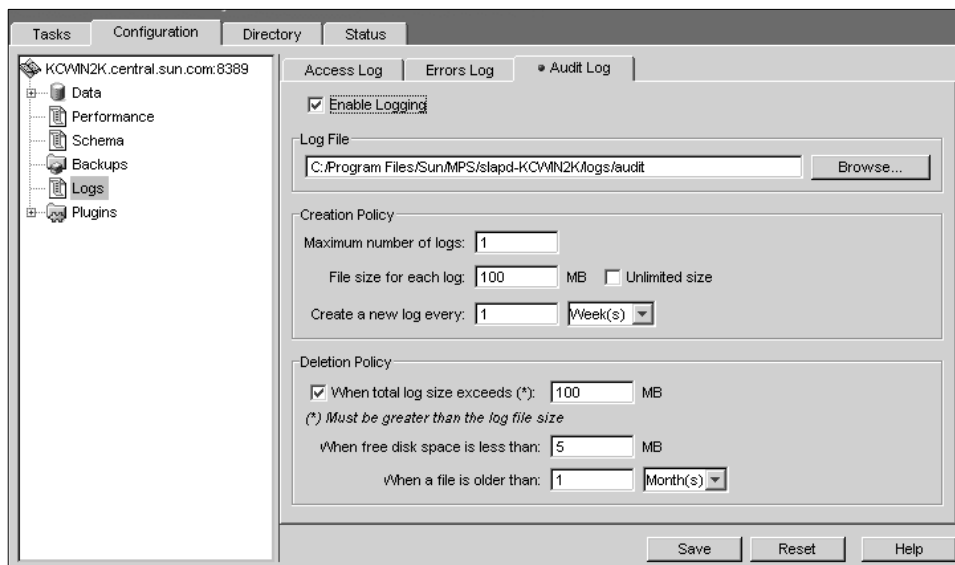
- Time of the audit event
- DN of the affected object
- Type of change
- Specific attributes values that changed
- Who performed the change

These log files are available at `<directory server root>/slapd-<host>/logs/` in the access and audit files respectively. (Directory Server keeps back-ups of these files in time-stamped files.)

The access file is enabled by default, and logs messages for directory binds, unbinds, searches, creates, edits, deletes, and renames.

The audit log file is not enabled by default. To enable audit logging, use the following steps:

1. Log into the Sun ONE Server Console.
2. Double-click a server name in the tree to open it.
3. Navigate to the Audit log configuration under Configuration > Logs > Audit Log.
4. Enable audit logging by enabling the Enable Logging check box.

**Figure 10-1** Enabling Audit Logging

5. Click Save.

## Application Server Logs

Generally, relevant information is available from one of the previously mentioned sources; but occasionally, error information will be logged in the application server's log files. The location and format of these messages varies, depending on which application server is being used to run Directory Editor.

Consult your application server's product documentation for information about accessing and configuring logs.



# Common Problems and Solutions

This section describes some common problems you might encounter while using Directory Editor and explains how to remedy these problems.

- “java.lang.OutOfMemoryError” on page 165
- “Failed to Save Startup Configuration Properties” on page 165
- “java.security.AccessControlException” on page 167
- “WebSphere is Extremely Slow During Startup” on page 168
- “Changing the Object Class and Attributes Used for Configuration Objects” on page 169

## java.lang.OutOfMemoryError

An `OutOfMemoryError` message in the Directory Editor log file indicates the application server has run out of memory.

To increase the amount of memory available to the application server, you can add the `-Xmx<size>` setting to your JVM — using the `<size>` value to specify the maximum amount of memory the JVM is allowed to use. For example, `-Xmx512M` tells the server to use a maximum of 512 megabytes of memory.

Consult your application server’s product documentation for information about how to set command line arguments for the application server’s JVM.

## Failed to Save Startup Configuration Properties

When Directory Editor is first started, you are prompted to enter some start-up properties that define where the configuration information for Directory Editor should be stored, how to bind to store the configuration information, and a few other options. Saving this information may fail if the application server cannot write to the `<application root>/WEB-INF/startup.properties` file.

You will encounter this problem if you have configured the application server to run this application in unexpanded mode or the application server’s security features do not permit access to the host file system.

- **To start Directory Editor using the configuration you have specified:**  
Click Continue Without Saving to run the application in *non-production mode* (see “Running in Non-Production Mode” on page 43), which means your Directory Editor configuration will not be saved and you will be prompted to re-enter the configuration information every time you restart Directory Editor.
- **To fix this problem permanently:** If you are running from an unexpanded WAR (web application archive) file, extract and manually edit the `WEB-INF/startup.properties` file and then repackage the WAR file with the changed `startup.properties`.

---

<b>NOTE</b>	You can also use this technique to edit the logging configuration mentioned in the previous section.
-------------	--

---

To manually edit the `startup.properties` file in an unexpanded WAR file:

- a. From a command prompt, navigate to the directory containing the `dml.war` file.
- b. Unzip the `dml.war` file with the following command (`jar` is installed with the Java Development Kit (JDK)):  

```
jar xvf dml.war
```
- c. Edit the `startup.properties` file to specify the required properties.
- d. Use the following command to update the `dml.war` with the updated `startup.properties`:  

```
jar uvf dml.war WEB-INF/startup.properties
```
- e. Re-deploy the updated `dml.war` file to the application server.

If you cannot save the updated `startup.properties` in an expanded WAR file, the application server does not have write access to the file. It might be necessary to give the user running the application server write access to `startup.properties`.

## java.security.AccessControlException

Directory Editor requires access to some privileged operations that the application server may not allow by default. This problem will become apparent when a user first attempts to log into Directory Editor and sees the *Application Error* page.

The Directory Editor logs will show a `java.security.AccessControlException`.

To remedy this problem you must edit the security policy file for your application server and grant access to certain actions, as follows:

---

**NOTE** In Sun ONE Application Server you will find this file in the following location:

`<App Server Home>/domains/<Domain Name>/config/server.policy`

---

1. Edit the `server.policy` file.
2. Add the following lines to the end of the block granting access to “all remaining code”:

```
permission javax.security.auth.AuthPermission"getLoginConfiguration";
permission javax.security.auth.AuthPermission"setLoginConfiguration";

permission javax.security.auth.AuthPermission
"createLoginContext.SunDirectoryLogin";

permission javax.security.auth.AuthPermission"modifyPrincipals";
```

3. Save the file.
4. Restart the application server.

## WebSphere is Extremely Slow During Startup

With some platforms, there is a performance impact if you use the JCE provided with that platform. If you experience a long start-up time, use Sun's JCE 1.2.2, which you can download from:

<http://java.sun.com/products/jce/>

Installation Instructions are provided at the following location:

[http://java.sun.com/products/jce/jce122\\_install.html](http://java.sun.com/products/jce/jce122_install.html)

If you install Sun JCE using the installation instructions provided, and you encounter exceptions during WAS startup, you may have to re-order the security providers specified in the following file so that the IBMJCE is first:

```
${java.home}/lib/security/java.security file
```

For example,

```
security.provider.1=com.ibm.crypto.provider.IBMJCE
```

```
security.provider.2=sun.security.provider.Sun
```

```
security.provider.3=com.sun.crypto.provider.SunJCE
```

Modify the `${java.home}/lib/security/java.policy` file as follows:

```
grant codeBase "file:${java.home}/lib/ext/
sunjce_provider.jar" {
    permission java.io.FilePermission
        "${java.home}/jre/lib/ext/jce1_2_2.jar", "read";
    permission java.lang.RuntimePermission "getProtectionDomain";
    permission java.security.SecurityPermission "putProviderProperty.SunJCE";
};
```

## Changing the Object Class and Attributes Used for Configuration Objects

By default, Directory Editor stores its configuration information in `applicationProcess` objects located in the configuration directory under:

`ou=1.0,ou=DML,ou=Services,<base context>`

If you want to change the location of these objects, the object class that is used to store the object, and/or the attributes used to store configuration information perform the following steps:

1. Back-up the existing configuration.
  - a. In Directory Editor, select **Configure > Backup/Restore**.
  - b. On the Backup/Restore page, click the **Backup Configuration to File** button.
  - c. When the Opening `config.xml` dialog box is displayed, enable the **Save to Disk** button to save the configuration XML file to your disk.
2. Delete the configuration organizational unit in the directory server.
  - a. Log into the Sun ONE Directory Server Console for the directory server that contains the configuration information.
  - b. Navigate to the configuration organizational unit in the directory at `ou=DML,ou=Services,<base context>`.
  - c. Right-click the DML entry and select **Delete**.
  - d. Click **Yes** to confirm deletion of the DML organizational unit.
3. Change the necessary properties in the `<application root>/WEB-INF/startup.properties` file.
  - a. To change the location where the configuration is stored, modify the `datastore.location` property so it points to the appropriate configuration organizational unit.
  - b. To change the object class used to store configuration object, modify the `datastore.objectClass` property to the appropriate object class.

---

**NOTE** You must define all required (*MUST*) attributes for the object class using the `datastore.dmlIdAttribute` or the `datastore.xmlObjectAttribute`.

---

- c. To change attributes in the object class used to store configuration information, modify the `datastore.dmlIdAttribute` and `datastore.xmlObjectAttribute`.
    - `dmlIdAttribute` contains a unique identifier used by Directory Editor
    - `xmlObjectAttribute` contains a (possibly large) XML text description of the object
4. Restart the application server.
5. Restore the configuration you backed up in Step 1.
  - a. In Directory Editor, select **Configure > Backup/Restore**.
  - b. On the Backup/Restore page, click the **Browse** button to locate and select the file you saved in Step 1. Click **Open**.
  - c. Click the **Restore Configuration from File** button to restore the file.

# HTML Components Used to Define Directory Editor Forms

This appendix describes the HTML components you can use when customizing Directory Editor forms.

## Overview

To customize a Directory Editor form, you can use Directory Editor's Form XML language (also called forms), to describe HTML display components.

Directory Editor interprets this language at runtime to build the necessary components, allowing you to generate new pages dynamically with little or no additional Java development — which greatly simplifies customization.

This section contains the following topics:

- “What Are HTML Components?” on page 172
- “Component Classes” on page 172
- “Page Processor Requirements for HTML Components” on page 173
- “Component Subclasses” on page 174

# What Are HTML Components?

HTML display components are instances of Java classes that generate a string of HTML text. Each display component has:

- **Class name** (defined in the field by the `DisplayClass` element): Identifies the component class, which determines the component's fundamental behavior and defines the set of properties recognized by the component.
- **Properties** (defined in the field with `Property` elements): Further define field behavior and appearance. You can specify one or more properties.

Specify display components as follows:

```
<Field name=' Name '>
  <Display class=' Class '>
    <Property name=' Name ' value=' Value '/>
  </Display>
</Field>
```

## Component Classes

HTML components are independent objects that can be combined in various ways. Related components are organized into classes. There are two major groups of component classes:

- **Basic Component classes:** Components used to display and edit a single value.
- **Container classes:** Components that can contain one or more components.

## Basic Component Classes

Common component classes include the components that are used to display and edit a single value. These components are defined in the section titled Basic Components.



# Container Classes

A *container* class defines a collection of components that are visually organized in a certain way. Typically, creating a container class results in the generation of an HTML `table` tag. Simple containers can concatenate the components horizontally or vertically. Other containers allow more flexible positioning of components and may add ornamentation around the components.

Because containers are themselves components, any container can be placed inside another container. You can use this mechanism to build complex page layouts.

For example, many pages consist of a title, followed by a list of editing fields, followed by a row of form submission buttons. You can create a page by creating a `Panel` component using vertical orientation that contains a `Label` component, an `EditForm` component, and a `ButtonRow` component.

The `EditForm` component itself contains some number of subcomponents. The `ButtonRow` component is simply a `Panel` that uses horizontal orientation and contains a list of `Button` components.

# Page Processor Requirements for HTML Components

This section describes the Directory Editor page processor requirements to display forms that implement HTML components. It covers:

## Hidden Parameters

Most components have a name that corresponds to the name of a parameter posted from an HTML form. Directory Editor reserves a few parameter names for general use. Do not use these names as component names.

**Table A-1** Hidden Parameter Names

Reserved Name	Description
id	Contains the ID of the object being edited
command	Contains the value of the button used to submit the form
activeControl	Contains the name of the last component that was active on the form

**Table A-1** Hidden Parameter Names *(Continued)*

Reserved Name	Description
message	Can contain an informational message to be displayed at the top of the page
error	Can contain an error message to be displayed at the top of the page

## Component Subclasses

All components extend the `Component` class, which defines the properties common to most components. In addition, some components extend the `Container` class, which gives them the ability to contain other components.

Each `Component` subclass defines a number of properties that are used to specify the characteristics of the component beyond those implied by the `Component` base class. For example, the `Label` component supports a `font` property, which can be used to specify the font used when rendering the label.

## Naming Conventions

Properties always begin with a lowercase letter and use camel case to separate adjacent words. Access method names are formed by capitalizing the property name, and prefixing either `get` or `set`. For example, the property named `font` is accessible from Java using the following methods:

- `getFont`
- `setFont`

The data type for each property varies and is documented with the property. the terminology used to describe property value types is described in the following table.

# Data Types

This table lists the data types allowed in component properties.

**Table A-2** Component Properties Data Types

Reserved Name	Description
null	Indicates that a property has no value
string	<p>Represents the most common data type. String values are usually represented by an instance of the Java <code>String</code> class. Some components are values of any class. These are implicitly coerced to strings with the <code>toString</code> method.</p> <p>Unless otherwise specified, you can assume that all properties are of type <code>string</code>.</p> <p><b>Example:</b></p> <pre>&lt;String&gt;Hello World&lt;/String&gt;</pre>
list of string	<p>Indicates that the value is expected to be a list of one or more strings. In Java, this value is always implemented as an instance of the <code>List</code> class. The elements of the list are then expected to be instances of the <code>String</code> class.</p> <p><b>Example:</b></p> <pre>&lt;List&gt;   &lt;String&gt;choice one&lt;/String&gt;   &lt;String&gt;choice two&lt;/String&gt; &lt;/List&gt;</pre>
comma list	<p>Represents a single string that contains character ranges separated by a comma character. Components that have multi-valued properties typically allow either <code>list of string</code> values or <code>comma list</code> values. When a comma list is assigned to a property, the component converts it to a list of strings by extracting the substrings between commas and adding them to a list in the order in which they appear in the string.</p>

**Table A-2** Component Properties Data Types *(Continued)*

Reserved Name	Description
boolean	<p>Represents a logically <code>true</code> or <code>false</code> value. Many properties require Boolean values.</p> <p>A Boolean value can be represented several ways. The <code>false</code> value represents one of the following values:</p> <ul style="list-style-type: none"> <li>• A null value</li> <li>• A Boolean object whose value is <code>false</code></li> <li>• An Integer object whose value is 0</li> <li>• A string whose value is anything other than <code>true</code>.</li> </ul> <p>When defining components in XML, Boolean values are most often represented with the strings <code>true</code> and <code>false</code>. (Any string value other than <code>true</code> is considered <code>false</code>.)</p>
int	<p>Represents a positive or negative integer. The <code>int</code> properties are commonly specified using strings. In that case, the string is coerced to an integer using the usual rules. If a string contains invalid integer characters, the coerced value is zero.</p>

## Base Component Class

The `Component` class is the base class for all HTML components. It contains the properties that are common to most components. Not all `Component` properties are relevant in every subclass. For example, `Component` defines a property `allowedValues` that can contain a list of value constraints. This property is relevant only in subclasses that allow value editing such as `Select` or `MultiSelect`.

Further, `Container` classes almost never directly represent an editable value. Consequently, any properties related to the component value are irrelevant. Some properties are relevant only if the component is contained within a specific `Container` class.

This section describes the Base Component Classes used by Directory Editor:

- “name” on page 177
- “title” on page 177
- “value” on page 178
- “allowedValues” on page 178
- “primaryKey” on page 179

- “required” on page 179
- “noNewRow” on page 179
- “location” on page 180
- “help” on page 180
- “command” on page 180
- “onClick” on page 181
- “nowrap, align, width, and colspan” on page 182

## name

Specifies the internal name of a field. All editing components must have a name, which is typically unique among all components displayed on the page. The `name` value is a string that is usually a path to a view attribute.

Container components do not require names and any assigned names are ignored. When building components from Java, component names are defined by the application. When building components from XML forms, component names are derived from the names of `Field` elements in the form. Field names are in turn path expressions within the view object that is used with the form.

### Example:

```
<Field name ='global.firstname'>
```

## title

(*Optional*) Specifies the external name of a field. Titles are typically used with the `EditForm` container, which builds an HTML table that contains titles in one column and components in another.

Components do not render their own titles. Rendering of titles is controlled by the container. Many containers ignore titles.

### Examples:

```
<Property name='title' value='FirstName' />

<Property name='title'>
  <expression>
    <concat>
      <s>Edit User: </s>
      <ref>waveset.accountId</ref>
```

```

        </concat>

    </expression>

</Property>

```

In this example, the field title is in part derived dynamically from the user's Directory Editor account ID.

## value

Editing components have a value that may be null. The value is typically set automatically by Directory Editor from an attribute in a view. Some components allow you to set the value by explicitly ignoring current view content. This value can be null.

The `Component` class allows the value to be any Java object. The subclass must coerce the value to a particular type when it is assigned, or when the HTML is generated. Component values are almost always `String` objects or `List` objects that contain strings. See “Data Types” on page 175 for more information about component value types.

Most container classes do not have values. If you assign a value, it is ignored. Some containers do allow values (for example, `TabPanel` and `WizardPanel`).

When building components from XML forms, the value is usually derived by using the component name as a path into the underlying view object, which contains all the values being edited.

### Example:

```
<Property name= 'value' value='false' />
```

## allowedValues

Specifies an optional list of allowed values for the component. If specified, the component allows you to select from only values that are on the list. If the component supports value restrictions, the list of allowed values is stored here. The value is always a list and usually contains strings. For convenience when setting properties from XML forms, you can also specify the allowed values as a comma list.

### Example:

```

<Property name='allowedValues' value= 'Mon, Tue, Wed, Thurs, Fri' />
<Property name='allowedValues' />

    <expression>

```

```

        <call name='DaysoftheWeek' />
    </expression>
</Property>

```

## primaryKey

This property is recognized only by the `SortingTable` container. The `SortingTable` container organizes components into a table with each column expected to contain components of the same class. `SortingTable` allows the rows to be sorted according to the values in any column. Typically, the sort order is determined from the value of each component in the column. There may be cases, however, where the value of the component is not suitable for sorting or may be inefficient to compare. In these cases, you can specify an alternate numeric sorting key.

## required

If `true`, indicates that the field is expected to have a value before the form is submitted. If the component is contained within an `EditForm`, a red \* (asterisk) will be placed after the component to indicate that the user must enter a value before saving. If the required schema map attribute is selected, (that is, set to a value of `true`), the field is always required.

The value of the property must be either `true` or `false`.

### Example:

```
<Property name='required ' value='true ' />
```

## noNewRow

If `true`, the field displays on the Directory Editor page next to the previous field. If not specified or set to `false`, the field appears on a new line, directly under the previous field. The default value is `false`.

This Boolean property is recognized only if the field is contained in a form that uses the `EditForm` display class. Typically, `EditForm` renders each component on a new row with the titles aligned in the left column and the component in the right column. To conserve space, you can concatenate several components on the same row. If the component also has a title, the title is rendered as non-highlighted text between the previous component and this component.

### Values include:

```
value='true ' | 'false '
```

**Example:**

```
<Property name='noNewRow ' value='true' />
```

**location**

Use if the container defines more than one display area and the component must be added to a specific area. Some containers allow the placement of components to be controlled by assigning a value to the `location` property.

For example, the `BorderedPanel` container supports five different display areas: north, south, east, west, and center.

The recognized values for the `location` property are defined by the container. If you do not assign a location, or assign a location name that is not recognized, the container places the component in the default location.

**help**

Specifies text that may be displayed to assist the user in understanding purpose of the field. In most Directory Editor pages, this will cause the `<icon>` icon to be displayed next to the component title. Moving the mouse over this icon will cause the help text to be displayed in the left margin.

The value of the property can either be literal text to be displayed, or it can be a message catalog key. Literal text can include HTML markup.

**command**

Specifies a command to submit when a component is modified. (When a user makes a change to a value, form output is recalculated.)

This property is typically used with the `Button` component. Some components must cause immediate submission of the surrounding HTML form when they are modified so that the application can regenerate the page based on that modification. Setting the `command` property to a non-null value causes this behavior.

When the `command` property is set, and the component is modified, the form is posted and an extra hidden parameter named `command` is posted whose value is the value of the `command` property.

The command specifies how the system will process the edits that have been made to a view. The `command` property must have one of the following values.



**Table A-3** Command Property Values

Value	Description
Save	Causes the edits to be saved.
Cancel	Causes the edits to be discarded.
Recalculate	Causes the page to be regenerated.
SaveNoValidate	Causes the edits to be saved, but no form validation to be performed.

Because specifying a command value of `Recalculate` is so common in forms, an shorter alternative syntax is available. The `Display` element has an attribute named `action` that when set to `true`, has the same effect as setting the `command` property to *Recalculate*.

```
<Display class='Select' action='true'>
```

## onClick

When specified, the value is expected to contain JavaScript that will be assigned as the value of the `onClick` attribute of the `input` element generated for this component. Not all components support the `onClick` property.

Use of this property is rare and requires detailed knowledge of the generated HTML. If you use this property, the page must typically contain a JavaScript component that defines JavaScript functions you call from within the `onClick` value.

### Example:

```
<Property name='onClick' value="Uncheck(this.form,
'resourceAccounts.selectAll');" />
```

**NOTE** Once forms are stored in the repository, Directory Editor always uses single quotes to surround attribute values. If single quotes appear within the attribute value, they will be replaced with `&#039;`. To prevent this escaping you can represent the string in an XPRESS expression:

```
<Property name='onClick'>
  <s>Uncheck(this.form, 'resourceAccounts.selectAll');</s>
</Property>
```

## onChange

Similar to `command`. The value can be an arbitrary JavaScript statement to run when the field is modified.

Not all components support the `onChange` property.

Use of this property is rare and requires detailed knowledge of the generated HTML. If you use this property, the page must typically contain a `Javascript` component that defines JavaScript functions you call from within the `onChange` value.

## nowrap, align, width, and colspan

Most containers position subcomponents by surrounding them with an HTML `table` tag. The HTML generated for each component then is typically contained in a `td` tag. Some containers can recognize the `nowrap`, `align`, `width`, and `colspan` properties and use them when generating the surrounding table cell tag. You can use these components to adjust the position and size of the component within the container.

The `nowrap` property affects how some components are displayed if they contain a long string of text. If the value of `nowrap` is `false` or unspecified, the browser can break up the component text into multiple lines when it is displayed. If the value of `nowrap` is `true`, the browser will try to keep the component text on a single line.

The `align` property is rarely used. You can use `align` to adjust the element horizontally on the form. Allowed values are `left`, `right`, and `center`.

Use of the `colspan` property is deprecated.

### Examples

```
<Property name='width' value='3' />
<Field name='Start Day' prompt='Day' nowrap='true' />
```

## Basic Components

This section describes the Basic Components used by Directory Editor:

- “BackLink” on page 183
- “Button” on page 184
- “DatePicker” on page 185
- “Javascript” on page 185

- “Label” on page 186
- “Link” on page 186
- “MultiSelect” on page 187
- “Radio” on page 189
- “SectionHead” on page 190
- “Select” on page 190
- “SimpleTable” on page 191
- “Text” on page 191
- “Text Area” on page 192
- “SimpleTable” on page 191

## BackLink

Displays a link that returns to the previous page. The behavior of this component is the same as that of the browser Back button. However, you may want to place this link in a convenient position on the page.

The only property for this display component is `text`, which is used to specify text for the link. If you do not specify `text`, the link defaults to `Back`.

### Example:

```
<Field name='back'>
  <Display class='BackLink'>
    <Property name='value' value='previous page' />
  </Display>
</Field>
```

## Button

Displays a button.

Properties for this display component are:

- **name:** Specifies the name of the parameter that will be posted when the user clicks this button. This property is optional; if not specified, the default value is `command`.
- **value:** Specifies the value of the parameter posted when the user clicks this button.
- **label:** Specifies the visible text that displays on the button.
- **command:** Specifies an optional value to submit along with the `name` parameter (for example, Save, Cancel, Recalculate).
- **postURL:** Specifies an alternate, target URL to which the form will be posted. This value overrides the URL specified in the JSP.
- **hiddenID:** Specifies an optional value for an id parameter to be included in the form post data.

### Example:

```
<Display class='Button'>
    <Property name='label' value='Change Password' />
    <Property name='value' value='Recalculate' />
</Display>
```

## Checkbox

Displays a checkbox.

- When enabled, the box represents a value of `true`.
- When disabled, the box represents a `false` value.

Properties for this display component are:

- **label:** (Optional) Specifies a label that is displayed to the right of the checkbox. It is displayed adjacent to the component, but is not displayed in the title column
- **leftLabel:** Specifies that the label should appear to the left of the checkbox.
- **value:** Determines the state of the checkbox. If the value is logically true, the checkmark is displayed.

**Example:**

```
<Field name='accounts[NT].passwordExpired'>
  <Display class='checkbox'>
    <Property name='title' value='Password is Expired' />
  </Display>
</Field>
```

**DatePicker**

Allows the user to specify a date using an applet that displays a calendar. The field is displayed in the form as a calendar icon. When the icon is clicked, the calendar applet is launched in a separate window.

Properties for this display component are:

The only property for this display component is `value`, which specifies the date to be highlighted on the calendar as the current date. You can parse the `Date` property from either a `Date` object or a `String` object. String values are expected to conform to the "MM/dd/yyyy" format.

**Example:**

```
<Field name='ExpireDate'>
  <Display class='DatePicker'>
    <Property name='title' value='Set Password Expire date' />
  </Display>
</Field>
```

**Javascript**

Use to insert pre-formatted JavaScript into the page. This is useful if you are using the `onClick` or `onChange` properties in components and want to call custom JavaScript functions.

Though not required, you should specify the `name` property when building components from XML forms. Using features such as field loops and field inclusion, you can add more than one JavaScript component containing the same script to the page. During HTML generation, JavaScript components that have the same name are included only once.

The component has an extended property named `script` that can contain the JavaScript text.

You can also include JavaScript by setting the property named `source`. This should be a string containing a URL fragment relative to the base context. It is the JavaScript contained in the indicated file to be loaded by the browser.

## Label

Displays a string of text.

Properties for this display component are:

- **value:** Defines the text to be displayed. The value can be either a string or a list of strings. When the value is a list, each string in the list is displayed on a separate line.
- **leftPad:** Specifies the number of spaces to insert to the left of the label.
- **pad:** Specifies the number of spaces to insert to the left and right of the label.
- **rightPad:** Specifies the number of spaces to insert to the right of the label.

---

**NOTE**      If no padding is specified, the default padding is leftPad=2, rightPad=2.

---

### Example:

```
Title<Field>
  <Display class='Label'>
    <Property name='title' value='Account ID' />
    <Property name='value'>
      <ref>waveset.accountId,/ref>
    </Property>
  </Display>
</Field>
```

- **font:** Specifies the font style. The value must be one of the styles defined in the styles/style.css file of the Directory Editor installation directory.
- **color:** Specifies the label color. Use standard HTML color formatting (# xxxxxx) to specify the color value.

## Link

Places a link on the page.

Properties for this display component are:

- **URL:** Specifies the target Uniform Resource Locator (URL).
- **imageURL:** (*optional*) Specifies the URL to an icon or image that will be rendered to the right of the link.

- **imageURL2:** (*optional*) Specifies the URL to an icon or image used will be rendered to the right of the first image.
- **hoverText:** Specifies text to display when the mouse rests over the first or second image.
- **id:** (*optional*) Specifies a value to be included as the id query argument in the link.
- **arguments:** (*optional*) Specifies a set of name/value pairs to be included as query arguments.
- **extraURL:** (*optional*) Specifies an additional URL fragment to be included after the base URL and arguments.

### Example:

```
<Field>
  <Display class='Link'>
    <Property name='name' value='Request Group Access' />
    <Property name='URL' value='user/processLaunch.jsp?newView=true'>
    <Property name='id' value='Group Request Process' />
  </Display>
</Field>
```

## MultiSelect

Displays a multiple-selection text box, which displays as a two-part object in which a defined set of values in one box can be moved to a selected box. Values in the left box are defined by the `allowedValues` property, values are often obtained dynamically by calling a Java method such as `FormUtil.getResources`. The values displayed in the right side of a multiple-selection box are populated from the current value of the associated view attribute, which is identified through the field name.

The form titles for this two-part object are set through the `availabletitle` and `selectedtitle` properties.

Properties for this display component are:

- **availableTitle:** Specifies the title of the available box.
- **selectedTitle:** Specifies the title of the selected box.
- **ordered:** Defines whether selected items can be moved up or down within the list of items in the text box. A true value indicates that additional buttons will be rendered to permit selected items to be moved up or down.

- **allowedValues:** Specifies the values associated with the left side of the multiple-selection box. This value must be a list of strings.

---

**NOTE**      The *<Constraints>* element can be used to populate this box, but its use is deprecated.

---

- **sorted:** Specifies that the values in both boxes will be sorted alphabetically.
- **noApplet:** Specifies whether the multiselect will be implemented with an applet or with a pair of standard HTML select boxes. The default is to use an applet, which is better able to handle long lists of values.
- **typeSelectThreshold:** (Available only when the `noApplet` property is set to `true`.) Controls whether a type-ahead select box appears under the `allowedValue` list. When the number of entries in the left select box reaches the threshold defined by this property, an additional text entry field appears under the select box. As you type characters into this text field, the select box will scroll to display the matching entry if one exists. For example, if you enter **w**, the select box scrolls to the first entry that begins with **w**.

#### Example:

```
<Field name='accounts[LDAP].LDAPDept' type='string'>
  <Display class='MultiSelect' action='true'>
    <Property name='title' value='LDAP Department' />
  </Display>
  <Constraints>
    <o>
      <List>
        <String>Sales</String>
        <String>Marketing</String>
        <String>International Sales</String>
      </List>
    </o>
  </Constraints>
</Field>
```



## Radio

Displays a horizontal list of one or more radio buttons. A user can select only one radio button at a time. If the component value is null or does not match any of the allowed values, no button is selected.

Properties for this display component are:

- **title:** Specifies the title for all radio buttons.
- **labels:** Specifies an alternate list of button labels. The labels list must be as long as the values in the `allowedValues` list. Alternate labels can be used in cases where the values are cryptic. For example, values can be letter codes such as "H", "M", and "S", but you would use this property to identify button labels "hours", "minutes", and "seconds".
- **allowedValues:** Specifies the value associated with each button. This value must be a list of strings.
- **value:** Specifies values for the buttons. This value accepts one string. If not set, then the values are the same as the labels.

### Example:

To display a container that contains two fields that prompt the user for a delayed start time for an activity, specify the following code. It will create a text field in which the user enters a numeric value and a second field of radio buttons that the user sets to specify the increment. The field has three radio buttons labeled (in order) Days, Hours, and Minutes, with values of minutes, hours, days, weeks, and months.

The `StartDate` field converts the value specified by the user into hours.

The only property for this display component is `value`, which specifies the date to be highlighted on the calendar as the current date. Date can be parsed from either a Date object or a String object. String values are expected to conform to the "MM/dd/yyyy" format.

### Example:

```
<Field name='ExpireDate'>
  <Display class='DatePicker'>
    <Property name='title' value='Set Password Expire date' />
  </Display>
</Field>
```

## SectionHead

Displays a new section heading defined by the value of the `text` property. It is an extension of the `Label` class that sets the font property to a style that results in large bold text. It also sets the `pad` property to zero to eliminate the default two-space padding. Use it to break up long forms into sections separated by a prominent label.

The only property for this display component is `text`, which specifies the text to be displayed.

### Example:

```
<Field>
  <Display class='SectionHead'>
    <Property name='text' value ='Calculated Fields' />
  </Display>
</Field>
```

## Select

Displays a single-selection list box. Values for the list box must be supplied by the `allowedValues` property.

Properties for this display component are:

- **allowedValues:** Specifies the list of selectable values for display in the list box.
- **autoSelect:** When set to `true`, this property causes the first value in the `allowedValues` list to be automatically selected if the initial value for the field is `null`.
- **multiple:** When set to `true`, allows more than one value to be selected.
- **nullLabel:** Specifies the text that displays at the top of the list box when no value is selected.
- **size:** (*optional*) Specifies the maximum number of rows to display. If the number of rows exceeds this size, a scroll bar is added.
- **sorted:** When set to `true`, causes the values in the list to be sorted.
- **valueMap:** Maps raw values to displayed values.

The component supports the `command` and `onChange` properties.

**Example:**

```

<Field name='city' type='string'>
  <Display class='Select'>
    <Property name='title' value='City' />
    <Property name='allowedValues'>
      <List>
        <String>Austin</String>
        <String>Portland</String>
        <String>New York</String>
      </List>
    </Property>
  </Display>
</Field>

```

**SimpleTable**

Arranges components in a grid with an optional row of column titles at the top. The only property for this display component is `columns`, which assigns column titles and defines the width of the table as defined in a list of strings.

**Text**

Displays a regular text entry box.

Common properties for this display component are:

- **size:** Specifies the number of characters that are visible in the text entry box. The box size is recalculated depending upon the length of the text in the box.
- **maxLength:** Specifies the maximum length of the string that can be edited in the text box.
- **secret:** Displays \*\*\*\*\* (asterisks) in the place of entered text. This option is most often used in password fields.
- **readOnly:** Displays read-only text. This text cannot be edited by the user. You might use this property if, for example, you want to display resource attribute information that an administrator needs to view when creating or editing user accounts.
- **submitOnEnter:** When this property is set and the Text field has focus, then when the user presses the Enter key, the form is submitted using the command that is specified in the property value. In the following example, the form is submitted exactly as though the user has clicked the Save button.

**Example:**

```

<Field name='variables.identityID'>
  <Display class='Text'>
    <Property name='required'>
      <Boolean>true</Boolean>
    </Property>
    <Property name='title' value='Identity ID' />
    <Property name='size' value='32' />
    <Property name='maxLength' value='128' />
    <Property name='submitOnEnter' value='Save' />
  </Display>
</Field>

```

**Text Area**

Displays a multi-line text entry box.

Properties for this display component are:

- **rows:** Specifies the number of text area rows.
- **columns:** Specifies the number of text area columns.
- **readOnly:** Displays read-only text in the text entry box.

**Example:**

To display a text box with five visible rows that wraps after each 70 characters specify:

```

<Field name='Description'>
  <Display class='TextArea'>
    <Property name='rows' value='5' />
    <Property name='columns' value='70' />
  </Display>
</Field>

```

If the user enters text beyond the defined visible rows, the text area displays a scroll bar.

# Resources for Role Configuration

The following table lists the resources that are available for configuring roles.

**Table B-1** Accessible Resources for Role Configuration

Resource Name	Type	Category	Description
/AddGroupMembersSearch	Struts Action	Object Edit/Create	Displays a form for adding members to a group object
/BootstrapEditor	Struts Action	Initialization	Edit Startup Properties
/CloseAndCancelWindow	Struts Action	Core	Handles pop-up dialog cancel actions
/CloseAndSaveWindow	Struts Action	Core	Handles pop-up dialog save actions
/ConfigImpExp	Struts Action	Configuration	Displays a form for backing up and restoring the configuration to and from file
/ConfigureManagedDirectory	Struts Action	Configuration	Displays a form for editing of managed directory properties during runtime
/Create	Struts Action	Object Edit/Create	Displays a form for editing object attributes before creation
/CreateFromList	Struts Action	Object Edit/Create	Displays a form for editing object attributes before creation; returning to the directory tree when finished
/CreateOptions	Struts Action	Object Edit/Create	Displays a form for specifying the type and location for a new object
/CreateOptionsFromList	Struts Action	Object Edit/Create	Displays a form for specifying the type and location of a new object; returning to the directory tree when finished
/Debug	Struts Action	Debug	Enables the Debug tab

**Table B-1** Accessible Resources for Role Configuration *(Continued)*

Resource Name	Type	Category	Description
/Debug/Config	Struts Action	Debug	Forward user to a page showing startup properties
/Debug/HTTPSession	Struts Action	Debug	Forward user to a page showing current HTTP session state
/Debug/Memory	Struts Action	Debug	Forward user to a page showing current memory usage
/Delete	Struts Action	Object Edit/Create	Display a form prompting the user form confirmation before deleting an object
/DeleteFromList	Struts Action	Object Edit/Create	Display a form prompting the user form confirmation before deleting an object; returning to the directory tree when finished
/Dialog	Struts Action	Core	Handle pop-up dialog interactions
/Disable	Struts Action	Object Edit/Create	Display a form prompting the user form confirmation before disabling an object
/DisableFromList	Struts Action	Object Edit/Create	Display a form prompting the user form confirmation before disabling an object; returning to the directory tree when finished
/Edit	Struts Action	Object Edit/Create	Display a form for editing an existing object's attributes
/EditPassword	Struts Action	Object Edit/Create	Display a form for editing the user's own password
/Enable	Struts Action	Object Edit/Create	Display a form prompting the user form confirmation before enabling an object
/EnableFromList	Struts Action	Object Edit/Create	Display a form prompting the user form confirmation before enabling an object; returning to the directory tree when finished
/EndBootstrapMode	Struts Action	Initialization	Sets internal system state indicating that the application is ready for use
/Error	Struts Action	Core	Forward user to a generic error dialog
/ExportConfiguration	Struts Action	Configuration	Writes the current configuration to an output stream
/ExportGroupMembers	Struts Action	Object Edit/Create	Download a newline-delimited file of the member DNs of a given group
/ExportLDIF	Struts Action	Configuration	Download an LDAP file of a given object

**Table B-1** Accessible Resources for Role Configuration *(Continued)*

Resource Name	Type	Category	Description
/Forbidden	Struts Action	Core	Forward user to an error dialog telling them they are forbidden from accessing the requested resource
/Help	Struts Action	Core	Forward user to the online help browser
/Home	Struts Action	Primary Navigation	Forward user to the home page
/Initialize	Struts Action	Initialization	Initializes the application upon startup verifying startup configuration
/InitializeConfig	Struts Action	Initialization	Forward user to a page for specifying startup properties
/InitializeConfigDirSubmit	Struts Action	Initialization	Handle startup properties
/InitializeConfigSaveError	Struts Action	Initialization	Forward user to a dialog saying that there was a problem persisting startup properties
/InitializeManagedDirectory	Struts Action	Initialization	Display a form for editing managed directory properties during initialization
/InlineDialogResponse	Struts Action	Core	Handles response from inline dialogs
/List	Struts Action	Primary Navigation	Forward user to the directory tree
/Login	Struts Action	Core	Forward user to log in page
/LoginSubmit	Struts Action	Core	Handles log in parameters
/LogoutSubmit	Struts Action	Core	Handles log out parameters
/ManageForms	Struts Action	Configuration	Display a form allowing the user manage system forms
/ManageRoles	Struts Action	Configuration	Display a form allowing the user manage application roles
/NamingAttributeConfig	Struts Action	Configuration	Display a form allowing the user specify object naming policy
/NoManagersWarning	Struts Action	Initialization	Forward user to a dialog warning the user that the manager group has no members
/PasswordExpired	Struts Action	Core	Forward user to a dialog saying that the user's password has expired

**Table B-1** Accessible Resources for Role Configuration *(Continued)*

Resource Name	Type	Category	Description
/PasswordExpiring	Struts Action	Core	Forward user to a dialog saying that the user's password will expire and giving them the option to change it
/RemoveGroupMembersSearch	Struts Action	Object Edit/Create	Display a form allowing the user to remove members from a group
/Rename	Struts Action	Object Edit/Create	Display a form prompting the user form confirmation before renaming an object
/RenameFromList	Struts Action	Object Edit/Create	Display a form prompting the user form confirmation before renaming an object; returning to the directory tree when finished
/Search	Struts Action	Primary Navigation	Forward user to a page allowing them to search the directory
/SearchConfig	Struts Action	Configuration	Display a form allowing the user specify properties the application will use for searching the directory
/SetApplicationMode	Struts Action	Initialization	Sets internal system state indicating the mode at which the application is operating
/ShowGroupMembersSearch	Struts Action	Object Edit/Create	Display a form allowing the user to display the current members from a group
/Unavailable	Struts Action	Core	Forward user to a dialog saying that the application is currently unavailable
.nav_bar.Base	Struts Tile	Core	Base type for navigation bars
.nav_bar.Configure	Struts Tile	Configuration	Navigation bar with the configuration tab selected
.nav_bar.Create	Struts Tile	Object Edit/Create	Navigation bar with the create tab selected
.nav_bar.Debug	Struts Tile	Debug	Navigation bar with the debug tab selected
.nav_bar.Home	Struts Tile	Primary Navigation	Navigation bar with the home tab selected
.nav_bar.List	Struts Tile	Primary Navigation	Navigation bar with the directory tree tab selected
.nav_bar.Search	Struts Tile	Primary Navigation	Navigation bar with the search tab selected
.nav_tab_selected.Base	Struts Tile	Core	Base type for selected navigation bar tabs



**Table B-1** Accessible Resources for Role Configuration *(Continued)*

Resource Name	Type	Category	Description
.nav_tab_selected.Configure	Struts Tile	Configuration	Selected navigation tab for configure
.nav_tab_selected.Create	Struts Tile	Object Edit/Create	Selected navigation tab for create
.nav_tab_selected.Debug	Struts Tile	Debug	Selected navigation tab for debug
.nav_tab_selected.Home	Struts Tile	Primary Navigation	Selected navigation tab for home
.nav_tab_selected.List	Struts Tile	Primary Navigation	Selected navigation tab for the directory tree
.nav_tab_selected.Search	Struts Tile	Primary Navigation	Selected navigation tab for search
.nav_tab.Base	Struts Tile	Core	Base type for unselected navigation bar tabs
.nav_tab.Configure	Struts Tile	Configuration	Unselected navigation tab for configure
.nav_tab.Create	Struts Tile	Object Edit/Create	Unselected navigation tab for create
.nav_tab.Debug	Struts Tile	Debug	Unselected navigation tab for debug
.nav_tab.Home	Struts Tile	Primary Navigation	Unselected navigation tab for home
.nav_tab.List	Struts Tile	Primary Navigation	Unselected navigation tab for the directory tree
.nav_tab.Search	Struts Tile	Primary Navigation	Unselected navigation tab for search
.page.AddGroupMembersSearch	Struts Tile	Object Edit/Create	Page displaying a form for adding members to a group
.page.AuthBase	Struts Tile	Core	Base type for application pages requiring authentication for access
.page.Base	Struts Tile	Core	Base type for application pages
.page.Bootstrap	Struts Tile	Initialization	Page for specifying startup properties at initialization time
.page.BootstrapEditor	Struts Tile	Configuration	Page for specifying startup properties at run time

**Table B-1** Accessible Resources for Role Configuration *(Continued)*

Resource Name	Type	Category	Description
.page.BootstrapSaveError	Struts Tile	Initialization	Dialog saying that there was a problem persisting startup properties
.page.CloseAndCancelWindow	Struts Tile	Core	Page for handling pop-up dialog cancel requests
.page.CloseAndSaveWindow	Struts Tile	Core	Page for handling pop-up dialog save requests
.page.ConfigImpExp	Struts Tile	Configuration	Page displaying a form for backup and restore of configuration data
.page.Create	Struts Tile	Object Edit/Create	Page displaying a form for editing object attributes before creation
.page.Debug.Config	Struts Tile	Debug	Page showing startup properties
.page.Debug.HTTPSession	Struts Tile	Debug	Page showing current HTTP session state
.page.Debug.Memory	Struts Tile	Debug	Page showing current memory usage
.page.Delete	Struts Tile	Object Edit/Create	Page displaying a form prompting the user form confirmation before deleting an object
.page.Dialog	Struts Tile	Core	Page for displaying information in a pop-up
.page.Disable	Struts Tile	Object Edit/Create	Page displaying a form prompting the user form confirmation before disabling an object
.page.Edit	Struts Tile	Object Edit/Create	Page displaying a form for editing an existing object's attributes
.page.EditPassword	Struts Tile	Object Edit/Create	Page displaying a form for editing the user's own password
.page.Enable	Struts Tile	Object Edit/Create	Page displaying a form prompting the user form confirmation before enabling an object
.page.Error	Struts Tile	Core	A generic error dialog
.page.Forbidden	Struts Tile	Core	Error dialog telling the user they are forbidden from accessing the requested resource
.page.Forms	Struts Tile	Configuration	Page displaying a form for management of system forms
.page.Help	Struts Tile	Core	Page displaying the help viewer

**Table B-1** Accessible Resources for Role Configuration *(Continued)*

Resource Name	Type	Category	Description
.page.Home	Struts Tile	Primary Navigation	Page displaying a form for the user's home page
.page.InitializeManagedDirectory	Struts Tile	Initialization	Page displaying a form for specification of managed directory properties during initialization
.page.Initializing	Struts Tile	Core	Page displaying a wait message for use during initialization
.page.InlineDialog	Struts Tile	Core	Base page for supporting inline dialogs
.page.List	Struts Tile	Primary Navigation	Page displaying a form representing the directory tree
.page.Login	Struts Tile	Core	Page prompting for log in credentials
.page.ManagedDirectory	Struts Tile	Initialization/Configuration	Page displaying a form for editing managed directory properties during run time
.page.NamingAttributeConfig	Struts Tile	Configuration	Page displaying a form allowing the user specify object naming policy
.page.NoManagersWarning	Struts Tile	Initialization	A dialog warning the user that the manager group has no members
.page.PasswordExpired	Struts Tile	Core	A dialog saying that the user's password has expired
.page.PasswordExpiring	Struts Tile	Core	A dialog saying that the user's password will expire and giving them the option to change it
.page.Popup	Struts Tile	Core	Base page for supporting pop-up dialogs
.page.RemoveGroupMembersSearch	Struts Tile	Object Edit/Create	Page displaying a form for removing members from a group
.page.Rename	Struts Tile	Object Edit/Create	Page displaying a form prompting the user form confirmation before renaming an object
.page.Roles	Struts Tile	Configuration	Page displaying a form for management of system roles
.page.Search	Struts Tile	Primary Navigation	Page displaying a form for searching the directory
.page.SearchConfig	Struts Tile	Configuration	Page displaying a form for configuration of application directory searches
.page.Secondary	Struts Tile	Core	Base page for content secondary pages

**Table B-1** Accessible Resources for Role Configuration *(Continued)*

Resource Name	Type	Category	Description
.page.ShowGroupMembersSearch	Struts Tile	Object Edit/Create	Page displaying a form for showing members of a group
.page.Unavailable	Struts Tile	Core	Error dialog telling user that the application is currently unavailable
.subnav_bar.Base	Struts Tile	Core	Base type for sub-navigation bars
.subnav_bar.BootstrapEditor	Struts Tile	Configuration	Sub-navigation bar with the startup properties tab selected
.subnav_bar.ConfigImpExp	Struts Tile	Configuration	Sub-navigation bar with the configuration backup/restore tab selected
.subnav_bar.Debug.Config	Struts Tile	Debug	Sub-navigation bar with the debug tab showing startup properties selected
.subnav_bar.Debug.HTTPSession	Struts Tile	Debug	Sub-navigation bar with the debug tab showing HTTP session state selected
.subnav_bar.Debug.Memory	Struts Tile	Debug	Sub-navigation bar with the debug tab showing current memory usage selected
.subnav_bar.ManagedDirectory	Struts Tile	Configuration	Sub-navigation bar with the managed directory tab selected
.subnav_bar.NamingAttributeConfig	Struts Tile	Configuration	Sub-navigation bar with the naming attribute configuration tab selected
.subnav_bar.Roles	Struts Tile	Configuration	Sub-navigation bar with the role management tab selected
.subnav_bar.SearchConfig	Struts Tile	Configuration	Sub-navigation bar with the Search Options tab selected
.subnav_divider	Struts Tile	Core	The tab divider
.subnav_tab_selected.Base	Struts Tile	Core	Base type for selected sub-navigation tabs
.subnav_tab_selected.BootstrapEditor	Struts Tile	Configuration	Selected sub-navigation tab for startup properties
.subnav_tab_selected.ConfigImpExp	Struts Tile	Configuration	Selected sub-navigation tab for backup/restore of configuration
.subnav_tab_selected.Debug.Config	Struts Tile	Debug	Selected sub-navigation tab for showing startup properties for debug
.subnav_tab_selected.Debug.HTTPSession	Struts Tile	Debug	Selected sub-navigation tab for showing HTTP session state for debug

**Table B-1** Accessible Resources for Role Configuration *(Continued)*

Resource Name	Type	Category	Description
.subnav_tab_selected.Debug.Memory	Struts Tile	Debug	Selected sub-navigation tab for showing current memory usage for debug
.subnav_tab_selected.Forms	Struts Tile	Configuration	Selected sub-navigation tab for managing system forms
.subnav_tab_selected.ManagedDirectory	Struts Tile	Configuration	Selected sub-navigation tab for specifying managed directory properties
.subnav_tab_selected.NamingAttributeConfig	Struts Tile	Configuration	Selected sub-navigation tab for managing naming attributes
.subnav_tab_selected.Roles	Struts Tile	Configuration	Selected sub-navigation tab for managing system roles
.subnav_tab_selected.SearchConfig	Struts Tile	Configuration	Selected sub-navigation tab for specifying search properties
.subnav_tab.Base	Struts Tile	Core	Base type for unselected sub-navigation tabs
.subnav_tab.BootstrapEditor	Struts Tile	Configuration	Unselected sub-navigation tab for startup properties
.subnav_tab.ConfigImpExp	Struts Tile	Configuration	Unselected sub-navigation tab for backup/restore of configuration
.subnav_tab.Debug.Config	Struts Tile	Debug	Unselected sub-navigation tab for showing startup properties for debug
.subnav_tab.Debug.HTTPSession	Struts Tile	Debug	Unselected sub-navigation tab for showing HTTP session state for debug
.subnav_tab.Debug.Memory	Struts Tile	Debug	Unselected sub-navigation tab for showing current memory usage for debug
.subnav_tab.Forms	Struts Tile	Configuration	Unselected sub-navigation tab for managing system forms
.subnav_tab.ManagedDirectory	Struts Tile	Configuration	Unselected sub-navigation tab for specifying managed directory properties
.subnav_tab.NamingAttributeConfig	Struts Tile	Configuration	Unselected sub-navigation tab for managing naming attributes
.subnav_tab.Roles	Struts Tile	Configuration	Unselected sub-navigation tab for managing system roles
.subnav_tab.SearchConfig	Struts Tile	Configuration	Unselected sub-navigation tab for specifying search properties



# Index

## A

### access

- anonymous log-ins 30
- configuring policy files 14, 18
- controlling for applications 133
- Directory Editor permissions 2, 9, 167
- dynamic groups 70
- logging directory access 163
- logging in 36
- Manager Group 33
- user 135
- WEB-INF/classes directory 31

### accessible

- components 135
- resources 139, 193

### Accessible Resources tab 139

### action attributes 181

### action components 135

### Advanced Search tab 118

### align property 182

### allowed values property 178

### anonymous log-ins 30

### Apache Tomcat 2

### application servers

- configuring policy files 14, 18
- deploying Directory Editor 23, 25
- deploying Directory Editor on WebLogic 26
- installing Sun 13
- installing Tomcat 17
- installing WebLogic 24
- Sun 10, 13
- supported 2

### undeploying Directory Editor 156

### applications, controlling access 133

### attribute values 94

### attributes

- action 181
- adding/removing 34, 48, 120, 128, 143
- calculating hidden 96
- creating 143, 144
- default 142
- defining values 94, 96
- deleting 147
- displaying 96
- dollar sign (\$) 127
- editing 82, 125, 143, 146, 169, 193, 194, 198
- inetorgperson 94
- LDAP 106
- lists 120
- Locality 74
- logged information 163
- Mail User 55
- managing 201
- mapping 144
- member 66, 67
- memberURL 69
- Naming Attribute Mapping table 143
- naming attributes 142, 143
- Naming Attributes page 142
- naming corresponding fields 106
- onClick 181
- organizationalperson 94
- organizationalunit 94
- read-only 96
- registeredAddress 76, 79

- removing 147
- required 47, 57, 61, 73, 77, 80, 169
- schema map 179
- searching for 120
- selecting 120
- specifying Common Attributes 48, 74, 78
- specifying Default NT User 57
- specifying extensions 53
- specifying for managed directory 34
- specifying Locality 80
- specifying Misc Attributes 52
- specifying UNIX User 59
- uniqueMember 66, 67
- using single quotes 181
- view 177, 178, 187
- audit logs 163
- authenticating
  - clients 133
  - users 34
- authentication, description/definition 133
- Authorization page
  - accessing 133, 136
  - controlling user access 133
  - defining roles 137
  - deleting roles 141
  - editing roles 140
- authorization, description/definition 133
- auxiliary classes
  - selecting 112

## B

- BackLink component 183
- base component classes 176
- Basic Component classes 172
- Basic Search tab 65, 115
- BEA WebLogic 2
- Browse
  - button 42
  - dialog box 42
  - page 30, 36, 39
- browsers, supported 3
- browsing

- Browse dialog box 42
  - directory contents 39
  - with anonymous log-in 30, 36
- build numbers 160
- Button component 184

## C

- central log directories xi
- changing LDAP passwords 37
- Checkbox component 184
- class of service (COS) template 94
- classes
  - adding to enterprise 143
  - adding to forms 112
  - Basic Component 172
  - Component 172, 176, 178
  - configuring logging levels 162
  - container 172, 173, 174
  - custom 131
  - datastore.dmlIdAttribute 31
  - datastore.location 31
  - datastore.objectClass 31
  - datastore.xmlObjectAttribute 31
  - default naming attributes 142
  - editing 169
  - groupOfNames 61
  - groupOfUniqueNames 61
  - groupOfURLs 69
  - HTML display components 172
  - Label 190
  - List 175
  - location 169
  - mapping naming attributes 144
  - removing 84
  - selecting auxiliary 112
  - selecting object 119
  - selecting structural 111
  - specifying naming attributes 144
  - string 175
- clients, authenticating 133
- colspan property 182
- command property 180



**commands**

- Directory Editor conventions xi
- extracting dml.war 166
- installing Directory Editor 10
- java 4, 10, 154
- launching uninstall wizard 154
- minimum/maximum heap size 4
- starting Directory Editor 22, 24
- stopping Directory Editor 23, 24
- updating dml.war 166

**comments and suggestions xiv****Common Attributes, specifying 48, 74, 78****Component classes 172, 176, 178****components**

- BackLink 183
- Button 184
- Checkbox 184
- classes 172
- DatePicker 185
- Javascript 185
- Label 186
- Link 186
- MultiSelect 187
- Radio 189
- SectionHead 190
- Select 190
- SimpleTable 191
- subclasses 174
- Text 191
- Text Area 192

**components, accessible 135****configuration directory 5, 29, 30, 40, 169****Configuration Directory Server parameters 29****configuring**

- default search filters 126, 127
- Directory Editor 39
- policy files 2, 14, 18
- Results table 128
- Search page 126
- search types 130
- servers 3
- WebLogic server 25

**contacting tech support 159****container classes 172, 173, 174****controlling access**

- applications/application components 133

**conventions**

- default paths and file names xi
- mnemonics xi
- naming 174
- naming attributes 142
- symbol xi
- typographic x

**Create page**

- accessing 39, 46, 117
- Common Attributes tab 48, 78, 80
- creating domains 77
- creating groups 60
- creating new users 46
- Default Dynamic Group Form tab 69
- Default NT User tab 57
- Extensions tab 68
- Group Membership tab 50
- Import/Export tab 67
- Locality tab 78
- Location tab 51
- Misc Attributes tab 52
- Other tab 79
- Quick Create tab 80
- Static Group tab 62
- Unix User tab 59

**creating**

- domains 77
- form fields 110
- forms 111
- groups 60
- localities 80
- naming attributes 143
- objects 45
- organizations 72
- roles 137
- users 46

**credentials 30, 36, 38, 133, 199****custom classes 131****customizing**

- default fields 96
- default forms 95
- field locations 108
- form field properties 105
- forms 95, 98

## D

- data types 175
- database directories, supported 3
- datastore.dmlIdAttribute 31, 169
- datastore.dmlIdAttribute property 170
- datastore.location 31
- datastore.location property 169
- datastore.objectClass 31
- datastore.objectClass property 169
- datastore.xmlObjectAttribute 31
- datastore.xmlObjectAttribute 169
- datastore.xmlObjectAttribute property 170
- DatePicker component 185
- default
  - application server instance folders 14
  - BackLink value 183
  - Button value 184
  - changing default search types 130
  - changing limits 41, 116, 121, 125
  - configuration information location 155, 169
  - configuring default search filter 126
  - fields 96
  - filtered search string 124
  - forms 93
  - group object 61
  - inetorgperson 131
  - installation directory xi
  - log files 162
  - log4j.properties file 162
  - naming attributes 142
  - noNewRow value 179
  - padding 186, 190
  - paths and filenames xi
  - ports 17
  - roles 133, 134, 141
  - search filters 127
  - search method 39
  - search results 41
  - search results limits 116, 121, 125
  - setting default values 95
  - user extensions 53
- Default Create Form 99
- Default NT User attributes, specifying 57
- Default NT User tab 57
- Default role 134
- defining 94
  - attribute values 96
  - dml.home variables 14
  - domains 94
  - roles 137
- Delete page, accessing 117
- deleting
  - classes 84
  - form fields 110
  - naming attributes 143, 147
  - objects 91
  - roles 141
- deploying Directory Editor 16, 23, 25, 26
- directories
  - browsing 39
  - central log default xi
  - creating objects 39
  - default paths and filenames xi
  - Directory Editor installation 11
  - editing information 39
  - managed 33
  - searching 39
  - top-level objects 40
  - WEB-INF/classes 22, 31
- Directory Editor
  - configuring 39
  - installing 10, 13
  - installing manually 23
  - installing on WebLogic 24
  - installing on WebSphere 26
  - installing with Tomcat Manager 22
  - navigating 38
  - starting 22, 24
  - stopping 23, 24
  - undeploying 156
  - uninstalling 153
- directory information 39
- Directory Server
  - authenticating to 29
  - installation prerequisites 5
  - logs 163
  - port number 29
  - publications viii
- Disable page
  - accessing 117

- disabling
  - objects 88
- DisplayClass element 172
- dml.home, defining 14
- dml.war file 23
  - default installation directory xi
  - extracting 166
  - installing 24, 26
  - installing manually 23
  - locating 23
  - removing 154, 155
  - updating 166
- DNs
  - downloading 85, 194
  - exporting 94
  - invalid 66, 67
- documentation
  - downloading xii, 13
  - overview xii
  - recommended
    - reading xii
- dollar sign (\$) 127
- domains
  - creating 77
  - defining 94
  - mail user attributes 55
  - NT user attributes 57
  - selecting 24
  - tree view 40
  - Unix user attributes 59
- downloading
  - J2SDK 8
  - JCE 1.2.2 168
  - member DNs 85
  - product documentation xii, 13
  - Sun products xii
- Dynamic Group extensions 68
- dynamic groups
  - access control 70
  - adding filters 70

## E

- Edit an Object page
  - accessing 39, 84, 117
- Edit page
  - accessing 41, 82
- Edit Password page, accessing 39
- Edit Roles page 140
- editing 39
  - attributes 82, 125, 146, 169, 193, 194, 198
  - filters 72
  - LDAP passwords 39
  - naming attributes 146
  - objects 82
  - roles 140
  - Search page 126
  - startup.properties 166
- editing objects 45
- Enable page
  - accessing 117
- enabling
  - objects 88
- exceptions
  - policy files 9, 167, 168
  - startup.properties 165
- exporting group members 85
- extensions
  - Dynamic Group 68
  - specifying 53
  - user 53

## F

- features 35
- field locations, customizing 108
- field properties
  - customizing 105
  - descriptions 106
- Field Properties tab 107, 110
- fields
  - creating 110
  - customizing 96
  - deleting 110

- files
  - installed with Directory Editor 10
- filtered searches 124
  - default search string 124
- filters
  - adding to dynamic groups 70
  - editing 72
  - removing 72
- Firefox 3
- forms
  - adding classes 112
  - changing field locations 108
  - creating 111
  - creating fields 110
  - customizing 95, 98
  - customizing field properties 105
  - default 93
  - Default Create Form 99
  - deleting fields 110
  - descriptions 93
  - field properties 102, 106
  - identification 103
  - move fields 102
  - opening 99
  - overview 95
  - property view 99, 101
  - setting default values 95
  - standard/default 94
  - testing 113
  - tree view 99, 100
- Forms page
  - accessing 97
  - description 98

## **G**

- groupOfNames 61
- groupOfUniqueNames 61
- groupOfURLs 69
- groups
  - adding members 62
  - assigning users 50
  - creating 60
  - default object 61

- importing/exporting members 85
- removing members 64
- searching for 115
- showing members 66
- viewing members 64, 66

## **H**

- heap size 4
- Help 40, 43
- help property 180
- hidden attributes 96
- hidden parameters 173
- Home page 37
- HTML display components 172

## **I**

- IBM WebSphere 2
- identifying forms 103
- importing group members 85
- inetorgperson 94, 131
- installation instructions 7
- installer, launching 10
- installing
  - application servers 8
  - Directory Editor 10, 13
  - Directory Editor for Sun application servers 13
  - Directory Editor for Tomcat 17
  - Directory Editor for WebLogic 24
  - Directory Editor for WebSphere 26
  - Directory Editor manually 23
  - specifying the directory and path 11
  - status 12
  - Sun application servers 13
  - Tomcat application servers 17

**J**

## J2SDK

- downloading 8
- requirements 7

## Java classes 172

## java commands

- adding heap size 4
- installing Directory Editor 10
- uninstall 154

## Java compilers 8

Java Development Kit. See *JDK*.Java Virtual Machine. See *JVM*.

## JAVA\_HOME 18

- adding 9
- setting 18

## Javascript component 185

## JCE, downloading 168

## JDK

- preferred 8
- requirements 5

## JVM

- adding options for dml.home 14
- increasing memory 165
- properties to allow read access 20
- running Java classes 8
- setting heap size 4

**K**

## key icon 101

**L**

## Label class 190

## Label component 186

## launching the installer 10

## LDAP attributes 106

## LDAP passwords

- changing 37

## LDAP passwords, editing 39

## license agreement 10

## Link component 186

## List classes 175

## localities, creating 80

## Locality attributes, specifying 74, 80

## location property 180

## locations, user 51

## Log In page 36

## log4j.properties file 162

## logging

- attribute changes 163
- auditing messages 163
- configuring logging levels 162
- setting levels 161

## logging in

- anonymously 30
- Directory Editor Log In page 36
- error messages 30

## logging out 40

## logs

- application server 164
- default 162
- default path xi
- default paths and file names xi
- Directory Server 163
- getting information 161
- java.security.AccessControlException 167
- WSTrace.log file 162

**M**

## mail accounts 55

## Mail User attributes, specifying 55

## managed directory 5

- specifying attributes 34
- specifying properties 33

## Manager role 133, 134

## managing attributes 201

## mapping naming attributes 144

## member attribute 66, 67

## member DNs, downloading 85

## memberURL attribute 69

- memory
  - increasing 165
  - requirements 4
- Microsoft Internet Explorer 3
- Misc Attributes, specifying 52
- mnemonics, using xi
- Move Field tab 108
- Mozilla 3
- MultiSelect component 187

## **N**

- name property 177
- naming attributes
  - accessing Naming Attributes page 142
  - creating 144
  - default 142
  - deleting 147
  - editing 146
  - mapping 144
  - Naming Attribute Mapping table 143
  - removing 147
  - specifying 34
- Naming Attributes page 143
- naming conventions 174
- Netscape 3
- node keys 41
- noNewRow property 179
- non-production mode 43
- nowrap property 182

## **O**

- object classes
  - adding 143
  - default naming attributes 142
  - editing 169
  - location 169
  - mapping naming attributes 144
  - specifying naming attributes 144

- objects
  - creating 45
  - creating domains 77
  - creating groups 60
  - creating localities 80
  - creating organizations 72
  - creating users 45
  - deleting 91
  - editing 45
  - enabling/disabling 88
  - renaming 89
  - searching for 118
  - top-level directory 40
- onChange property 182
- onClick attribute 181
- onClick property 181
- online Help 40, 43
- online resources xii
- online support xii
- operating systems, supported 2
- organizational units
  - searching for 115
- organizationalperson attribute 94
- organizationalunit attribute 94
- organizations
  - creating 72

## **P**

- page source
  - viewing 160
- pages
  - Edit 41
  - Home 37
  - Quick Create 61
  - Search Configuration 41, 126
  - Search For Members to Remove From Group 64
- parameters
  - hidden 173
- passwords
  - changing 37
  - editing 39
- patches

- information about xii
- performance
  - impacts 27, 127, 168
  - optimizing 3, 4
- permissions 2
- policy files
  - configuring application servers 2
  - configuring Sun Application Server 14
  - configuring Tomcat 18
  - exceptions 9, 167, 168
  - removing permissions 43
- ports, specifying 29
- prerequisites
  - recommended reading viii
- primary key property 179
- principals
  - description/definition 134, 135
  - representing with directory objects 135
  - specifying 138
- product documentation
  - downloading xii, 13
  - related xii
- product downloads xii
- product features 35
- product support xiii
- properties
  - align 182
  - allowed value 178
  - colspan 182
  - command 180
  - datastore.dmlIdAttribute 170
  - datastore.location 169
  - datastore.objectClass 169
  - datastore.xmlObjectAttribute 170
  - help 180
  - location 180
  - managed directory 33
  - name 177
  - naming conventions 174
  - noNewRow 179
  - nowrap 182
  - onChange 182
  - onClick 181
  - primary key 179
  - required 179
  - title 177
  - value 178
  - width 182

- Property elements 172
- property view 99, 101
- publications
  - Directory Server viii, xii
  - downloading xii
  - related xii

## Q

- query strings 124
- Quick Create page 61, 77
- quotes, using 181

## R

- Radio component 189
- read-only attributes 96
- recommended
  - reading viii, xii
  - server configurations 3
- registeredAddress attribute 76, 79
- related documentation xii
- removing
  - Directory Editor 153
  - dml.war file 154, 155
  - filters 72
  - naming attributes 143, 147
- Rename page
  - accessing 117
- renaming objects 89
- required
  - attributes 47, 57, 61, 73, 77, 80, 169
- required property 179
- requirements
  - application servers 2
  - browsers 3
  - database directories 3
  - J2SDK 7

- Java compilers 8
- JDK 5
- memory 4
- operating systems 2

- resources
  - online xii

- Results table
  - configuring 128
  - using 117

- roles
  - accessible resources 193
  - default 133, 134, 141
  - defining 137
  - deleting 141
  - description/definition 134
  - editing 140

## S

- schema map attributes 179

- Search Configuration page
  - accessing 41, 126
  - configuring default search filters 127
  - configuring Results table 128
  - configuring search types 130

- Search For Members to Remove From Group
  - page 64

- search methods
  - advanced 118
  - authenticating users 34
  - basic 116
  - descriptions 39
  - example searches 121
  - filtered 124

- Search page
  - accessing 39
  - Advanced Search tab 118
  - Basic Search tab 115
  - configuring 126
  - configuring default search filters 126, 127
  - editing 126
  - Filtered Search tab 124
  - search types 115

- search results

- changing default limits 41, 116, 121, 125
  - maximum results 116, 125

- search types
  - changing default 130
  - configuring 130
- searching
  - advanced search 118
  - basic search 115
  - default filtered search string 124
  - examples 121
  - filtered search 124
  - for attributes 120
  - search conditions 120

- SectionHead component 190

- Select component 190

- server configurations, recommended 3

- setclasspath.bat file 18

- setclasspath.sh file 18

- SimpleTable component 191

- single quotes, using 181

- software, supported 1

- specifying
  - startup.properties 28
- starting Directory Editor 22, 24
- startup.properties
  - exceptions 165
  - persisting 31
  - saving 166
  - specifying 28, 30, 31, 166

- statistics, viewing 41

- stopping and restarting
  - Tomcat 23
  - WebLogic 24

- stopping Directory Editor 23, 24

- string classes 175

- structural classes
  - selecting 111

- subclasses, component 174

- subject
  - description/definition 135
- suggestions and comments xiv
- Sun application servers
  - installing 10, 13
  - installing Directory Editor 13



- undeploying Directory Editor 156
- Sun Java™ System Application Server 2
- Sun Java™ System Directory Server
  - installation prerequisites 5
- Sun ONE Application Server 2
- Sun online resources xii
- support, contacting 159
- support, product xii, xiii
- supported
  - application servers 2
  - browsers 3
  - database directories 3
  - operating systems 2
  - software 1
- symbol conventions xi

## T

- tabs
  - Advanced Search 118
  - Basic Search 65, 115
  - Common Attributes 48, 78, 80
  - Default Dynamic Group Form 69
  - Extensions 68
  - Field Properties 107, 110
  - Filtered Search 124
  - Group Membership 50
  - Import/Export tab 67
  - Locality 78
  - Location 51
  - Misc Attributes 52
  - Move Field 108
  - Other 79
  - Quick Create 80
  - Static Group 62
  - Unix User 59
- technical support xiii
- technical support, contacting 159
- testing forms 113
- Text Area component 192
- Text component 191
- third-party websites xiii
- title property 177

- Tomcat application servers
  - deploying Directory Editor 23
  - installing 17
  - installing Directory Editor 17
  - policy file 18
  - stopping and restarting 23
  - undeploying Directory Editor 156
- Tomcat Manager, installing Directory Editor 22
- tree view 99, 100
- typographic conventions x

## U

- undeploying Directory Editor 156
- uninstall wizard 154
- uninstalling Directory Editor 153, 154
- uniqueMember attribute 66, 67
- UNIX User attributes, specifying 59
- updating dml.war 166
- user view attributes 95
- users
  - assigning to groups 50
  - authenticating 34
  - creating 46
  - default extensions 53
  - mail accounts 55
  - miscellaneous information 52
  - physical location 51
  - searching for 115
  - Windows NT accounts 57

## V

- value property 178
- variables
  - adding JAVA\_HOME 9
  - dml.home 14
- view attribute 177, 178, 187
- view components 135
- viewing
  - build number 160

- directory contents 39
- folder contents 41
- form elements 100
- group members 64, 66
- page source 160
- statistics 41

## **W**

WEB-INF/classes directory permissions 22, 31

### **WebLogic**

- configuring server 25
- deploying Directory Editor 25, 26
- installing Directory Editor 24
- stopping and restarting 24
- undeploying Directory Editor 157
- Web application home directory 24

### **websites**

- comments and suggestions xiv
- Directory Server publications viii, xii
- Microsoft product documentation xiii
- Sun product documentation vii
- Sun resources xii
- support xiii
- third-party xiii

### **WebSphere**

- installing Directory Editor 26
- undeploying Directory Editor 157

width property 182

WSTrace.log file 162