

Novell Designer 2.1 for Identity Manager 3.5.1

2.1

www.novell.com

ADMINISTRATION GUIDE

September 28, 2007



Novell[®]

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. For more information on exporting Novell software, see the [Novell International Trade Services Web page \(http://www.novell.com/info/exports/\)](http://www.novell.com/info/exports/). Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 2007 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, 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 [Novell Legal Patents \(http://www.novell.com/company/legal/patents/\)](http://www.novell.com/company/legal/patents/) and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see [Novell Documentation \(http://www.novell.com/documentation/\)](http://www.novell.com/documentation/).

Novell Trademarks

For a list of Novell trademarks, see [Trademarks \(http://www.novell.com/company/legal/trademarks/tmlist.html\)](http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

Designer for Identity Manager: Help	15
1 What's New	17
1.1 What's New in Designer 2.1	17
2 Designer Concepts	19
2.1 Target Audience	20
2.2 Identity Manager: Overview	20
2.2.1 Managing Identities, Access, and Groups	20
2.2.2 How Identity Manager Works	21
2.3 Designer's Features	22
2.4 Highly Productive Environment	22
2.5 Tools for Developers	23
2.6 Features That Require Identity Manager 3.x Engines	23
2.6.1 Driver Set and Driver Log Events	24
2.6.2 Policy Script	25
2.6.3 Support for Entitlements	25
2.6.4 Support for Workflow-Based Provisioning	26
2.7 Designer and iManager	26
3 Planning Identity Solutions	29
3.1 Security: Best Practices	29
3.2 Solving Identity Issues	30
3.3 How to Use Designer	34
3.3.1 Designing New Solutions	34
3.3.2 Redesigning Existing Solutions	34
3.3.3 Working on Live Systems	35
4 Installing Designer	37
4.1 Requirements	37
4.1.1 System Requirements	37
4.1.2 Platform Requirements	37
4.2 Installing	37
4.2.1 Installing Designer If Eclipse Is Not Already Installed	38
4.2.2 Installing Designer If Eclipse Is Already Installed on Your System	40
4.3 Uninstalling Designer	40
5 Getting Help	41
5.1 Help from the Welcome Page	41
5.2 Browsing for Information	43
5.3 Getting Context-Sensitive Help	43
5.4 Help from the Web	44
5.5 Viewing Demos	45
5.6 Searching for Information	45

5.7	Customizing the Browser and Help Server	46
6	Creating a Project	49
6.1	When No Project Exists	49
6.2	When You Want to Create an Additional Project	50
6.3	When You Want to Import a Project	51
7	Building a Model	53
7.1	Basic Tasks	53
7.2	Accessing the Modeler	54
7.3	Selecting a Modeling Mode	55
7.3.1	Developer Mode	56
7.3.2	Architect Mode	57
7.3.3	Dataflow Mode	61
7.3.4	Table Mode	62
7.4	Working from the Palette	62
7.4.1	About the Palette	63
7.4.2	Palette Operations	63
7.4.3	Using Generic Applications	64
7.4.4	Fly-Out Palette	64
7.4.5	Resizing the Palette	65
7.4.6	Docking the Palette	66
7.4.7	Arranging Folders and Applications	66
7.4.8	Changing the Layout	67
7.4.9	Keyboard Support for the Palette	68
7.5	Copying and Pasting	68
7.5.1	Copying Applications	68
7.5.2	Copying a Driver Set	69
7.5.3	Copying an Identity Vault	70
7.5.4	Copying a Domain Group	71
7.5.5	Copying between Editors	71
7.6	Moving Items	72
7.7	In-Line Editing	73
7.8	Tooltips and Toolbar	73
7.9	Organizing by Domain Groups	74
7.9.1	About Domain Groups	74
7.9.2	Key Features	75
7.9.3	Creating a Domain Group	75
7.9.4	Minimizing (Collapsing) Domain Groups	77
7.9.5	Restoring Domain Groups	77
7.9.6	Maximizing Domain Groups	77
7.9.7	Auto-Placement of Neighbors	78
7.9.8	Grouping into a New Domain Group	78
7.9.9	Ungrouping a Domain Group	79
7.9.10	Clearing Contents	80
7.9.11	Changing a Domain Group Icon	81
7.9.12	Keyboard Support for Domain Groups	81
7.10	Connecting Applications	82
7.10.1	Automatic Connections	82
7.10.2	Connection Target Highlights	82
7.10.3	Automatically Creating Objects	83
7.10.4	Auto Redraw	83
7.10.5	Manually Connecting	83
7.10.6	eDir-to-eDir Connections	84

7.10.7	Multiple Driver Connections	84
7.10.8	Straightening Connections	86
7.10.9	Reconnecting	87
7.10.10	Driver Icons	87
7.10.11	Selected Drivers	87
7.10.12	Auto-Layout of Imported Objects	87
7.10.13	Keyboard Support for Connections	88
7.11	Aligning and Laying Out Components	88
7.11.1	Alignment Hints	89
7.11.2	Using Rulers	90
7.11.3	Using a Grid	92
7.11.4	Distributing Applications	93
7.11.5	Auto-Layouts	94
7.11.6	Layouts to Use for Imports	94
7.12	Editing Multiple Objects	95
7.13	Modeling Active Directory Domain Controllers	95
7.13.1	Configuring a Connection	95
7.13.2	Discovering Controllers	95
7.13.3	Information about Domain Controllers	96
7.14	Saving Your Model	96

8 Workspaces, Perspectives, and Views 99

8.1	Specifying the Workspace	99
8.1.1	Project File Format	100
8.1.2	Editing by Using a Related Editor	101
8.2	Managing Perspectives	105
8.2.1	Opening a Perspective	106
8.2.2	Resetting a Perspective	107
8.2.3	Closing a Perspective	107
8.3	Managing Views	107
8.3.1	Opening a View	108
8.3.2	Moving a View	108
8.3.3	Minimizing and Maximizing Views	109
8.3.4	Closing a View	109
8.3.5	Reopening a View Window	110
8.3.6	Using Fast Views	111
8.4	The Project View	113
8.4.1	Viewing and Editing Projects: The Project View	113
8.4.2	Tasks You Can Perform from the Project View	114
8.4.3	What You See in the Project View	116
8.5	The Outline View	121
8.5.1	Viewing Contents of the Active Editor: The Outline View	121
8.5.2	Tasks You Can Perform in the Outline View	122
8.5.3	The Outline View Tab	123
8.6	The Navigator View	135
8.7	The Policy Set View	138
8.8	The Properties View	139
8.8.1	The Spinner Control	140
8.8.2	The DN Cell Editor	140
8.8.3	Editing Boolean Values	140
8.8.4	Keyboard Tips	141
8.8.5	Resizing Column Widths	141
8.8.6	Summary Table	141
8.9	The Search Results View	142
8.10	The Tasks View	143

8.11	The Trace and Error Log Views	144
8.12	The Dataflow View	144

9 Configuring Objects in Designer 145

9.1	Viewing Object Properties	145
9.1.1	Properties View	145
9.1.2	Properties Dialog Box	146
9.1.3	Operations Relating to Properties	147
9.2	Configuring a Domain Group	148
9.3	Configuring Identity Vaults	148
9.4	Configuring Servers	150
9.5	Configuring Driver Sets	152
9.5.1	Driver Set General Options	153
9.5.2	Driver Set Global Configuration Values	154
9.5.3	Driver Set Log Levels	154
9.5.4	Driver Set Server List	155
9.5.5	Driver Set Trace	156
9.6	Configuring Drivers	158
9.6.1	Driver General Settings	159
9.6.2	Driver Global Configuration Values	160
9.6.3	Driver Log Level	160
9.6.4	Driver Configuration	161
9.6.5	Driver Manifest	163
9.6.6	Driver Named Passwords	163
9.6.7	Editing or Creating a Driver iManager Icon	164
9.6.8	Driver Trace Levels	165
9.6.9	Engine Control Values	167
9.7	Configuring Jobs	169
9.7.1	General	169
9.7.2	Trace	170
9.8	Configuring Application Properties	170
9.8.1	General	171
9.8.2	Environment	172
9.8.3	Administrator	172
9.8.4	Connectivity	172
9.8.5	AD Domain	174
9.9	Configuring Policies	174
9.9.1	Editing a Policy Name	175
9.9.2	Viewing References	175
9.9.3	Editing a Credential Provisioning Policy	175
9.10	Adding Prompts to a Driver Configuration File	175
9.11	Editing a Notification Template Name	177
9.12	Synchronizing Passwords	177

10 Managing Versions 179

10.1	Key Differences in Identity Manager Versions	180
10.2	Changing the Identity Manager Version	180
10.3	Tracking Versions of Identity Manager	181
10.4	Information in the Driver Configuration Wizard	183
10.5	Checking Projects for Version Issues	185
10.6	Adjusting the UI Based on the Version Number	186

11	Managing the Schema	187
11.1	Accessing the Manage Schema Tool	188
11.2	Creating Classes and Attributes	189
11.2.1	Creating eDirectory Classes	189
11.2.2	Creating eDirectory Attributes	194
11.3	Modifying the Schema	201
11.3.1	Deleting Schema Definitions	202
11.3.2	Modifying Classes or Attributes	202
11.3.3	Renaming Schema Definitions	202
11.4	Deploying the Schema into the eDirectory Tree	204
11.5	Exporting the Schema to a File	206
11.5.1	Exporting the Schema to a .sch File	206
11.5.2	Exporting the Schema to an LDIF File	208
11.6	Importing the Schema	209
11.6.1	Importing the Schema from the eDirectory Tree	209
11.6.2	Importing the Schema from a File	211
11.7	Managing a Copy of an Application Schema	216
11.7.1	Editing an Application's Schema	217
11.7.2	Refreshing the Application Schema	219
11.8	Mapping eDirectory to an LDAP Schema	220
12	Managing the Flow of Data	221
12.1	The Dataflow View	221
12.1.1	Accessing the Dataflow View	221
12.1.2	Flow Arrows in the Modeler	222
12.1.3	Viewing How Attributes Are Synchronized	224
12.1.4	Changing the Data Flow	224
12.2	The Dataflow Editor	227
12.2.1	Filtering Views	228
12.2.2	Filtering Identity Vaults and Applications	230
12.2.3	Pinning the Identity Vault	230
12.2.4	Expanding and Collapsing the Identity Vault	231
12.2.5	Switching to an eDirectory Tree Icon	233
12.2.6	Viewing an eDir-to-eDir Driver	234
12.2.7	Keyboard Support	235
12.3	Adding Items in the Dataflow Editor	235
12.3.1	Adding an Identity Vault in the Dataflow Editor	235
12.3.2	Adding a Driver in the Dataflow Editor	236
12.3.3	Adding an Application in the Dataflow Editor	236
12.3.4	Adding Classes and Attributes	237
12.3.5	Adding Non-Filter Attributes	238
12.4	Editing Items	239
12.4.1	Editing within the Dataflow Editor	240
12.4.2	Editing Non-Filter Attributes	241
12.4.3	Managing Schema	241
12.4.4	Removing a Flow	242
12.4.5	Changing Data Flows	242
12.5	Generating HTML Reports	243
12.6	Integrating Passwords	245

13	Creating and Managing Policies	247
14	Setting Up E-Mail Notification Templates	249
14.1	Viewing Notification Templates	249
14.2	Editing a Notification Template	251
14.3	Adding a Notification Template	255
14.4	Policy Builder and Notification Templates	256
14.5	Configuring the E-Mail Server	256
15	Importing into Designer	257
15.1	Importing Projects	257
15.1.1	Importing a Project from the Identity Vault	258
15.1.2	Importing a Project from the File System	262
15.1.3	Importing a Project from iManager	264
15.2	Importing a Driver Set or Driver from the Identity Vault	266
15.2.1	Associating a Server to the Identity Vault	266
15.2.2	Importing a Driver Set from the Identity Vault	268
15.2.3	Importing a Driver from the Identity Vault	273
15.3	Importing a Driver Configuration File	275
15.3.1	Importing an Identity Manager Project from the File System	275
15.3.2	Importing a Driver Configuration from a File in the Modeler View	275
15.3.3	Importing from a File through the Outline View	278
15.4	Importing Channels, Policies, and Schema Items from the Identity Vault	281
15.4.1	Importing a Channel	282
15.4.2	Importing a Policy	285
15.4.3	Importing a Schema	288
15.5	Using the Compare Feature When Importing	291
15.5.1	Using Compare when Importing a Driver Object	292
15.5.2	Using Compare On a Channel Object	295
15.5.3	Using Compare On a Policy	295
15.5.4	Matching Attributes with Designer Properties	295
15.6	Error Messages and Solutions	298
16	Documenting Projects	299
16.1	Creating a Document Style	299
16.2	Editing a Document Style for Your Needs	300
16.2.1	Editing a Style Template	301
16.2.2	Editing Sections of a Style	301
16.3	Generating a Document	302
16.4	Using Your Template Style for Other Projects	306
16.4.1	Start with the Style and Choose the Project	306
16.4.2	Start with the Project and Choose the Style	307
16.4.3	Document a Section of the Style	308
16.4.4	Document Multiple Sections of the Style	310
16.5	Customizing Styles to Include or Exclude Information	311
16.5.1	Limiting Documentation for the Identity Vault Schema and Application Schema	311
16.5.2	Deleting Extra Identity Vault Schema from Your Generated Report	315
16.5.3	Using Project Configuration to Limit Information	318
16.6	Advanced Editing of a Document Style	320
16.6.1	What's In the Advanced Editing Mode	320
16.6.2	A Walk-through Tutorial	327
16.6.3	Selecting a Language for Generated Documents	334

16.6.4	Double-Byte Font Support	335
16.6.5	Saving a Style File in the Styles Template Library	337
16.7	Troubleshooting Issues When Generating Documents	337
17	Managing Entitlements	339
17.1	Designing Entitlements	339
17.1.1	Terminology	340
17.1.2	Entitlement Prerequisites	340
17.1.3	Identity Manager Drivers with Preconfigurations that Support Entitlements	341
17.1.4	Enabling Entitlements On Identity Manager Drivers	341
17.2	Creating Entitlements through the Entitlement Wizard	342
17.2.1	Valueless Entitlements	343
17.2.2	Valued Entitlement that Queries an External Application	345
17.2.3	Administrator-Defined Entitlements with Lists	351
17.2.4	Administrator-Defined Entitlements without Lists	353
17.3	Editing and Viewing Entitlements	354
17.3.1	Entitlement XML Source and XML Tree Views	356
17.3.2	Using the Novell Entitlement DTD	359
17.4	Managing Entitlements	359
18	Scheduling Jobs	361
18.1	Creating a Job	361
18.2	Editing a Job	363
18.2.1	Job Editor Selections Under the General Tab	363
18.2.2	Job Editor Selections Under the Job Parameter Tab	367
18.2.3	Job Editor Selections Under the Scheduler Tab	371
18.2.4	Job Editor Selections Under the Notification Tab	373
19	Deploying and Exporting	375
19.1	Preparing to Deploy	375
19.2	Deploying a Project to an Identity Vault	376
19.3	Deploying a Driver Set to an Identity Vault	376
19.3.1	eDir-to-eDir Deployments and SSL/TLS	379
19.4	Deploying a Driver to an Identity Vault	379
19.5	Deploying a Channel to an Identity Vault	381
19.6	Deploying a Policy to an Identity Vault	382
19.7	Using the Compare Feature When Deploying	383
19.7.1	Using Compare when Deploying a Driver Object	384
19.7.2	Using Compare Before Deploying a Channel Object	387
19.7.3	Using Compare Before Deploying a Policy	387
19.7.4	Matching Attributes with Designer Properties	387
19.8	Troubleshooting Deployed Objects	387
19.9	Exporting a Project	387
19.10	Exporting To a File	390
19.10.1	The Export Context Menu	390
19.10.2	Exporting Configuration Files from the Modeler View	391
19.10.3	Exporting Configuration Files from the Outline View	391
20	The Novell XML Editor	395
20.1	About the Novell XML Editor	395
20.1.1	Creating XML Files	395

20.1.2	Validating Files	400
20.1.3	Outline View	400
20.1.4	XPath Navigator	400
20.2	Using the Source Editor	401
20.3	Using the Tree Editor	402
20.4	Attaching Schema or DTD	403
20.5	Setting XML Editor Preferences	405
21	Tools	407
21.1	Converting Earlier Projects	407
21.1.1	Converting Projects That Were Open	408
21.1.2	Converting Projects That Were Not Open	408
21.1.3	Running Later Projects on Earlier Designer Versions	412
21.2	Opening a Web Browser	413
21.3	Launching iManager	413
21.4	Checking Your Projects	414
21.4.1	How to Check Your Project	414
21.4.2	Customizing the Project Checker	418
21.4.3	Items That Are Checked	421
21.5	Managing Directory Objects	423
21.5.1	Tool-Based Method	424
21.5.2	Task-Based Method	425
21.5.3	Browse, View, or Modify Object Attributes	425
21.5.4	Advanced Editing	427
21.6	Configuring TLS for eDir-to-eDir Drivers	427
21.6.1	Prerequisites	427
21.6.2	Enabling TLS	428
21.6.3	Creating Certificates	430
21.7	Viewing DSTrace Log Files	431
21.8	Working with Generic Resources	434
21.8.1	Creating a Generic Resource Object	434
21.8.2	Editing a Generic Resource Object	435
21.9	Updating Designer	435
22	Editing Icons for Drivers and Applications	437
23	Setting Preferences	441
23.1	Finding Preference Pages	441
23.2	Preferences: General	441
23.2.1	Appearance	442
23.2.2	Capabilities	444
23.2.3	Compare/Patch	444
23.2.4	Content Types	444
23.2.5	Editors	445
23.2.6	Keys	447
23.2.7	Perspectives	447
23.2.8	Startup and Shutdown	447
23.2.9	Web Browser	448
23.2.10	Welcome	448
23.3	Preferences: Designer for IDM	448
23.3.1	Versions, Updates, Prompts, and Browsers	449
23.3.2	Configuration	451
23.3.3	Document Generation	455

23.3.4	Entitlements	457
23.3.5	iManager	458
23.3.6	Import/Deploy	458
23.3.7	Changing the Language Setting	460
23.3.8	Modeler	461
23.3.9	Policy Builder	465
23.3.10	Project Checker	466
23.3.11	Schema	466
23.3.12	Simulation	466
23.3.13	Trace	467
23.4	Preferences: Help	469
23.4.1	Help Server	469
23.5	Preferences: Novell	470
23.6	Preferences: Provisioning	470
23.7	Preferences: Web and XML	470
23.7.1	CSS Files	470
23.7.2	HTML Files	474
23.7.3	JavaScript Files	479
23.7.4	Task Tags	482
23.7.5	XML Catalog	482
23.7.6	XML Files	484

24 Troubleshooting Designer 491

24.1	Viewing the Error Log	491
24.1.1	Using Menus	491
24.1.2	Browsing the File System	492
24.2	Turning on Trace Messages	492
24.3	Checking Loaded Plug-Ins	494
24.4	Deploying Identity Manager Objects	495
24.4.1	An Example Deploy Error	496
24.4.2	Deploying a Driver with an Empty Server List	497
24.4.3	Error Messages and Solutions	498
24.5	Issues When Generating Documents	509
24.5.1	Memory Issues	509
24.5.2	Editing Extra Identity Vault Schema Items from Your Report	511
24.6	No F1 Help in Maximized Editors	513
24.7	Freeing Heap Memory	513
24.8	Reporting Bugs and Giving Feedback	514

A Designer: An Effective Tool 517

A.1	Core Application Services and Framework	517
A.2	Installation	518
A.3	Enterprise Modeler	518
A.4	Project View	519
A.5	Outline View	519
A.6	Import and Deploy	519
A.7	XML Editor	520
A.8	Configuration Management	520
A.9	Policy Management	520
A.9.1	Policy Flow View	520
A.9.2	Filter Editor	521
A.9.3	Schema Map Editor	521
A.9.4	Policy Builder	521

A.10	Policy Simulation	522
A.11	Dataflow View and Editor	522
A.12	Entitlements	522
A.13	E-Mail Notification Templates	523
A.14	Document Generator	523
A.15	Directory Tools	523
A.16	Version Handling	524
A.17	Project Converter	524
A.18	Project Checker	524
B Document Generator Core Support Templates		525
B.1	dgSection.xml	525
B.2	dgFormat.xml	526
B.3	idmConfig.xml	528
B.4	idmUtil.xml	529
C Designer's Architecture		533
C.1	Built on Eclipse Technology	533
C.2	Core Model Uses the Eclipse Modeling Framework	533
D Quickview: Menu Options		535
D.1	Options on the File Menu	535
D.2	Options on the Edit Menu	537
D.3	Options on the View Menu	538
D.4	Options on the Project Menu	539
D.5	Options on the Tools Menu	539
D.6	Options on the Live Menu	540
D.7	Options on the Model Menu	541
D.8	Options on the Window Menu	546
D.9	Options on the Help Menu	552
E Modeler Operations		553
E.1	Modeler Space Operations	553
E.2	Identity Vault Operations	554
E.3	Driver Set Operations	557
E.4	Driver Operations	559
E.5	Application Operations	562
E.6	Submenus	565
E.7	Keyboard Support	566
F Key Terms		569

Designer for Identity Manager: Help

Designer for Identity Manager can help you design, test, document, and deploy Identity Manager-based solutions in a highly productive environment.

Newcomers can use wizards to build Identity Management solutions. Veterans and expert users can bypass the wizards and interact directly at any level of detail.

Audience

This guide is for enterprise IT professionals and developers, consultants, sales engineers, architects or system designers, and system administrators.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comment feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html and enter your comments there.

Documentation Updates

For the most recent version of the *Designer for Identity Manager: Administration Guide*, visit the [Identity Manager Web site \(http://www.novell.com/documentation/idm35/\)](http://www.novell.com/documentation/idm35/).

Documentation Conventions

In Novell® documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as Linux* or UNIX*, should use forward slashes as required by your software.

Additional Documentation

- ◆ *Identity Manager 3.5.1 Installation Guide*
- ◆ *Novell Identity Manager 3.5.1 Administration Guide*
- ◆ *Understanding Policies for Identity Manager 3.5.1*
- ◆ *Policies in Designer 2.1*
- ◆ *Novell Credential Provisioning Policies for Identity Manager 3.5.1*
- ◆ *Identity Manager 3.5.1 DTD Reference*

- ◆ Identity Manager 3.5 User Application: Design Guide (<http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/dgpro/data/bookinfo.html>)
- ◆ Driver guides (<http://www.novell.com/documentation/idm35drivers/index.html>)

What's New

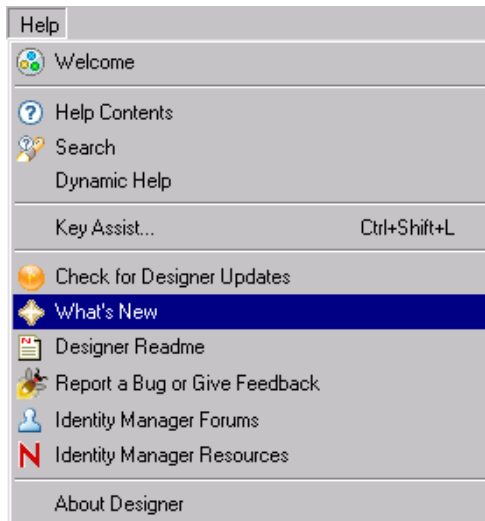
1

- ♦ [Section 1.1, “What’s New in Designer 2.1,” on page 17](#)

1.1 What’s New in Designer 2.1

To find out what’s new in Designer 2.1:

- 1 Run Designer for Identity Manager.



- 2 Click *Help* > *What’s New*.

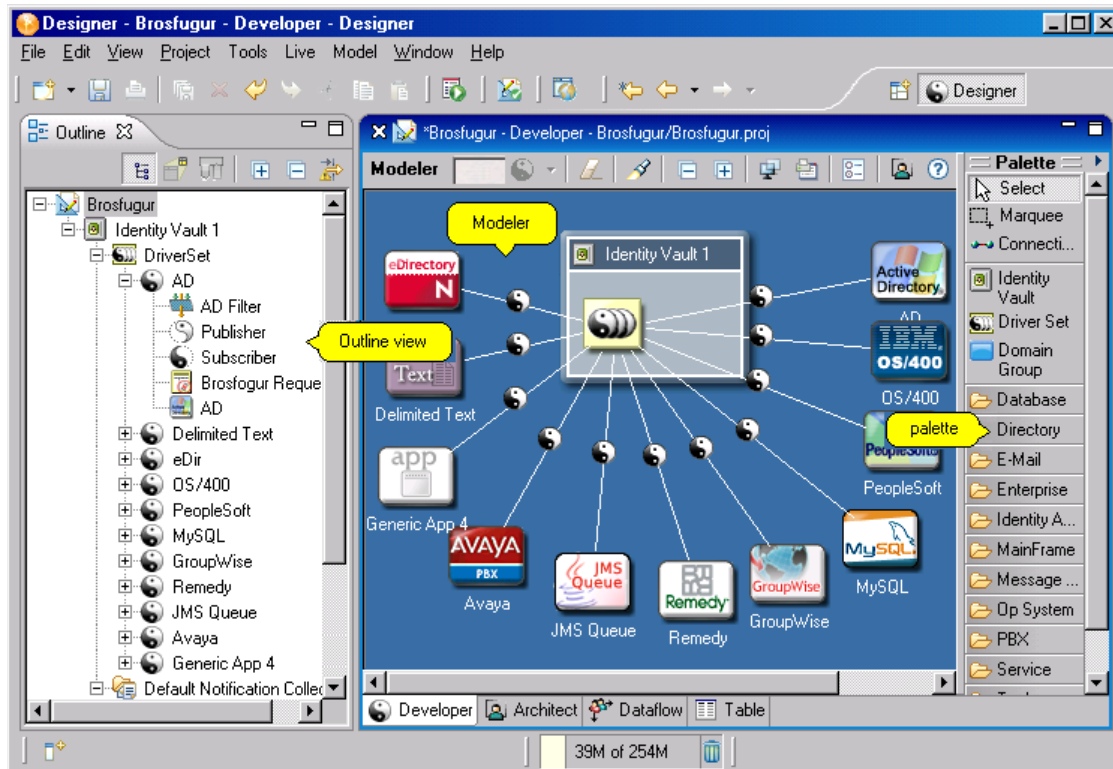
You can also click *Help* > *Help Contents* > *Designer for Identity Manager* > *What’s New in Designer*.

Designer Concepts

2

Designer is a powerful, graphical toolset that helps you configure and deploy Identity Manager.

Figure 2-1 The Outline View, Modeler, and Palette in Designer



An Identity Vault stores data and definitions. The Identity Manager engine manages the flow of data in and out of the Identity Vault as well as between other systems and directories. Drivers tie the systems together and ensure that communication flows smoothly, regardless of formats used. Designer's user interface uses an abstraction of the data in order to remain flexible without impacting existing data and definitions, until you decide to deploy the model that you have designed and tested.

- ◆ [Section 2.1, "Target Audience,"](#) on page 20
- ◆ [Section 2.2, "Identity Manager: Overview,"](#) on page 20
- ◆ [Section 2.3, "Designer's Features,"](#) on page 22
- ◆ [Section 2.4, "Highly Productive Environment,"](#) on page 22
- ◆ [Section 2.5, "Tools for Developers,"](#) on page 23
- ◆ [Section 2.6, "Features That Require Identity Manager 3.x Engines,"](#) on page 23
- ◆ [Section 2.7, "Designer and iManager,"](#) on page 26

2.1 Target Audience

Designer was created for the following audiences:

- ◆ Enterprise IT developers
- ◆ Consultants
- ◆ Sales engineers
- ◆ Architects or system designers
- ◆ System administrators

This tool is aimed at information technology professionals who

- ◆ Have a strong understanding of directories, databases, and their information environment
- ◆ Act in the role of a designer or architect of identity-based solutions

You don't need to be a developer or programmer to fully make use of every aspect of this tool. We provide many capabilities for developers to extend this tool to suit their own needs.

Wizards make this tool easy to learn and use for building Identity Management solutions. Experienced users can bypass the wizards and interact directly at any level of detail.

You can also use Designer as an effective and valuable tool to help communicate key Identity Solution concepts and design to strategic decision makers in the organization. You can use the visual Modeler as well as documentation that captures and displays Designer data.

2.2 Identity Manager: Overview

- ◆ [Section 2.2.1, “Managing Identities, Access, and Groups,” on page 20](#)
- ◆ [Section 2.2.2, “How Identity Manager Works,” on page 21](#)

2.2.1 Managing Identities, Access, and Groups

Identity Manager (formerly known as DirXML[®]) is a cross-platform service that helps you manage identity information across select systems on your corporate network or your partners' networks. For more information on the Novell[®] solution, visit the [Novell Identity Manager Web site \(http://www.novell.com/products/identitymanager\)](http://www.novell.com/products/identitymanager).

Identity Manager helps you manage identity information in any system for which Novell provides a special connector (and for connectors that you or a third party write). Novell provides connectors that interface with popular applications, databases and directories, including PeopleSoft*, SAP* HR, GroupWise[®], Microsoft* Exchange, Lotus* Notes*, Oracle*, and Microsoft Active Directory*. (An asterisk--*--denotes a third-party trademark.)

Identity Manager helps you securely manage the access needs of your ever-changing user community. You can manage the full user life cycle and deliver first-day access to essential resources. For example, when you create a new user in your HR database (or a customer or partner in the customer's related application), all of the user's accounts are generated automatically, according to your business rules, and any necessary resources or services are supplied.

You can also quickly and consistently manage groups. For example, you can automatically grant access rights to everyone with a given role. You can also synchronize multiple passwords into a

single login, modify or revoke access rights instantly, and even support compliance with government regulations.

Former employees pose a common threat to many businesses, but Novell Identity Manager closes this security loophole. The moment you change an employee's status to dismissed in an authoritative data source, Identity Manager terminates the user's accounts and access rights in all specified systems, and provides an audit report of the successfully revoked user access rights. Your confidential resources remain safe, and you eliminate countless hours of manual, error-prone administration.

With its Metadirectory capabilities, Identity Manager enables an identity integration foundation for unifying identities across platforms, systems and organizational boundaries. It permits departments to maintain control of the identity data that they are responsible for.

At the same time, the enterprise as a whole gains a single virtual identity for each user by

- ♦ Linking scattered identity information
- ♦ Setting rules that govern automatic updates to all designated systems when a change is made in an authoritative source

This foundation serves as the underlying architecture for password management, user provisioning, white pages, role-based administration, and other capabilities.

Since August 2002, Novell has been in the Leader quadrant of Gartner Metadirectory Magic Quadrant. Novell offers the most flexible identity management solution on the market. This solution adapts to each customer's needs and business rules, enables enterprises to automate processes for quick and efficient deployment, and provides the flexibility to customize and adapt the solution as requirements evolve.

2.2.2 How Identity Manager Works

To control when and how identity information is exchanged between these systems, you configure the systems' respective connectors by creating various types of rules. You base these rules on your company's needs and on its relationship with partners and employees.

Identity Manager uses Extensible Markup Language (XML) and extensible Stylesheet Language Transformations (XSLT), the two formats in which DirXML rules have traditionally been represented. Identity Manager 3.5 and later include features that exceed traditional DirXML capabilities by significantly simplifying the rule-building (business policy) process and by simplifying password management.

Novell Identity Manager runs on eDirectory™ 8.7.3 or later and supports all of the platforms that eDirectory supports, including NetWare®, Microsoft Windows* NT/2000, Red Hat* Linux, Solaris*, AIX* and HP-UX*. (For specific version numbers, visit the [Novell eDirectory Web site \(http://www.novell.com/products/edirectory/sysreqs.html\)](http://www.novell.com/products/edirectory/sysreqs.html)).

For an article on how Identity Manager works, see [Too Many Directories? \(http://www.novell.com/connectionmagazine/2000/05/DirXML50i.final.pdf\)](http://www.novell.com/connectionmagazine/2000/05/DirXML50i.final.pdf).

Also, see the [Novell Identity Manager \(http://www.novell.com/products/identitymanager/\)](http://www.novell.com/products/identitymanager/) Web site.

2.3 Designer's Features

Designer enables you to do the following:

- ◆ Carry projects with you, work offline, and save projects to disk.
- ◆ Keep a project open for an extended period of time.
- ◆ Work on multiple projects at the same time.
- ◆ Experience higher performance while designing and developing a model. Processing is performed on the local CPU.
- ◆ Enjoy a smoother interface with fewer clicks and fewer sub-windows to resize or close.
- ◆ Benefit from an included development/debugging environment and integration with other client-based tools for editing and debugging.

For a list of features, and details about them, see [Appendix A, "Designer: An Effective Tool,"](#) on [page 517](#).

2.4 Highly Productive Environment

The following help you be highly productive in your environment:

- ◆ Rich-client environment: work in a highly productive rich-client environment, with a native look and feel
- ◆ Disconnected mode: work well in a disconnected mobile environment when you're "on the go"
- ◆ Strong visual editors, minimal pop-ups, and well-synchronized views laid out to maximize productivity
- ◆ Powerful window, view, perspective, and tool management
- ◆ Reduction of steps to configure solutions
- ◆ Wizards to help you get started and configure projects
- ◆ Auto-creation of objects, auto-value, auto-connection, auto-layouts
- ◆ Strong copy/paste within and across editors
- ◆ Full keyboard support in all editors in a way that only a client can perform
- ◆ Full undo/redo in most editors and views
- ◆ Ability to scale the UI from basic to more advanced views
- ◆ Ability to filter key views to see as much or as little as you need to see
- ◆ Integrated XML editor with syntax highlighting, auto-suggest, and search and replace
- ◆ Very efficient, compact, powerful, and productive off-line schema management tools
- ◆ Value history on many fields to reduce typing
- ◆ Many preferences and settings that tailor the UI to how you want to use the product
- ◆ Logs and traces that are easy to access and view
- ◆ Thorough contextual help and powerful searchable help system
- ◆ Quick and very easy installation
- ◆ Auto-Update
Auto-update installation notifies you of any updates and easily pulls them in.

2.5 Tools for Developers

- ◆ Easily add and model something not in the shipping version.

For example, you can add your own applications, drivers, resources, and icons.

- ◆ Configure Designer to use a different editor.

Configure all file types (for example, `.xml` and `.txt`) to use your editor of choice. Eclipse-based editors work best, but you can also include various artifacts (for example, word processing documents and spreadsheets). The native editor is automatically integrated into Designer if the platform supports it.

- ◆ Develop and debug in Java*.

If you install Designer plug-ins into a full Eclipse install, you can do Java development and debugging, ANT, C#, and UML modeling, all in the same tool alongside Designer. This has particular value to Identity Manager driver writers (Java or C) who want the tools all together.

- ◆ Use public APIs.

Identity Manager uses fully published public Eclipse APIs, an underlying project data model that is consistent with open industry standards in its format, and published Eclipse extension points.

Though not public yet, a strong set of APIs and extension points is being developed to make it easy to add your own integrated wizard, editor, process, view, menu, toolbar item, or help page. For the most part, you can use the public APIs and extension points that Eclipse already provides. We are further augmenting that with additional APIs and extension points, based on Eclipse standards, to give you more power and options.

We use these APIs ourselves. Many internal developers are currently using them. When they are stable enough, they will be made public. We are not appending APIs as an afterthought, but building them in from the beginning. We fully support the open model, patterns, and API standards as outlined by Eclipse, rather than taking a proprietary approach.

In the future, with an off-the-shelf Eclipse book and knowledge of our extension points, you should be able to contribute very rich content to Designer. You will have an abundance of books, resources, and forums in the Eclipse community to address most of your needs. Some developers are having success doing this now, even without access to our APIs.

2.6 Features That Require Identity Manager 3.x Engines

The following features in Designer 2.1 won't work on Identity Manager engines that were installed earlier than Identity Manager 3.0:

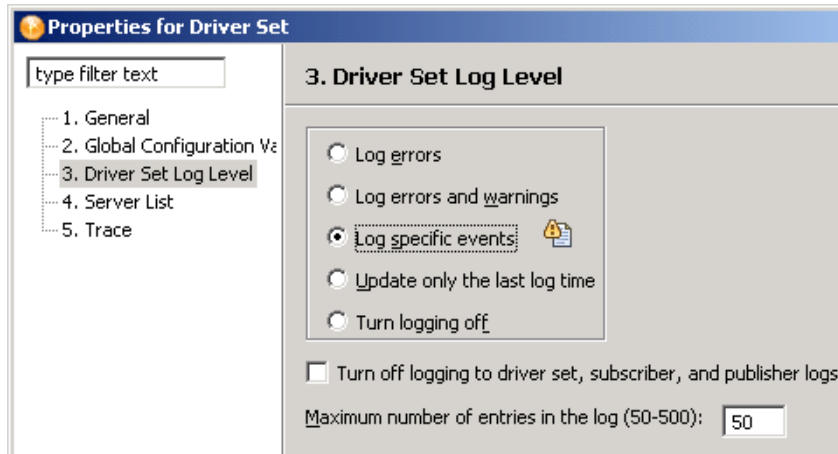
- ◆ Enhancements in Driver Set and Driver Log Events
- ◆ Changes in the Policy Script
- ◆ Support for entitlements
- ◆ Support for workflow-based provisioning

2.6.1 Driver Set and Driver Log Events

Driver Set Log Events

To identify non-functional properties for a driver set:


- 1 Right-click a driver set in the Modeler.
- 2 Select *Properties*.
- 3 Select *Driver Set Log Level > Log Specific Events*.



- 4 Click the *Expand* icon.



- 5 Select options.

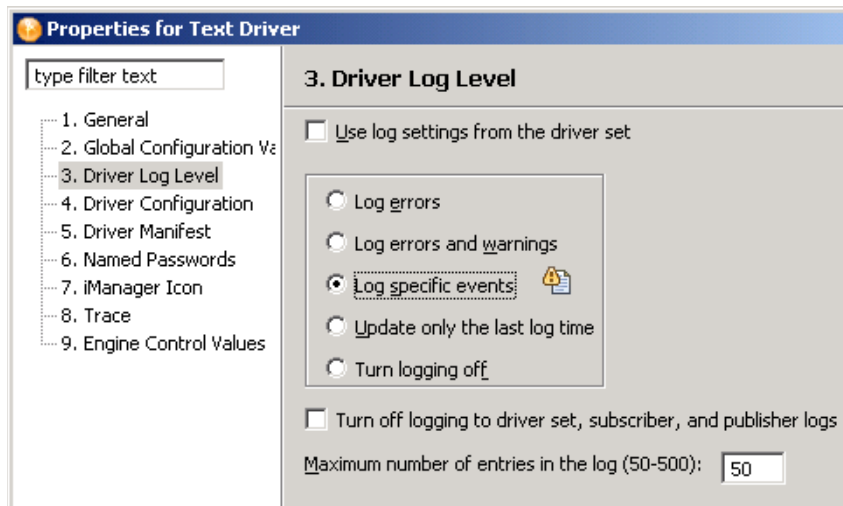
Items that have the earlier-version icon  won't work on Identity Manager engines earlier than 3.0.

Driver Log Events

To identify non-functional properties for a driver:

- 1 Right-click a driver in the Modeler.
- 2 Select *Properties*.

3 Select *Driver Log Level* > *Log Specific Events*.




If this option is dimmed, the driver is using properties from the driver set.

4 Click the *Expand* icon.



5 Select options.

Items that have the earlier-version icon  won't work on Identity Manager engines earlier than 3.0.

2.6.2 Policy Script

The following changes were made in the Policy Script for Identity Manager 3.

- ◆ Added `do-implement-entitlement`. Updated `do-for-each` to include implied `do-implement-entitlement`.
- ◆ Updated `token-entitlement`, `token-added-entitlement`, and `token-removed-entitlement` to return `entitlement-impl` in a `node-set` context and to change behavior relative to granted vs. revoked entitlements.
- ◆ Updated `if-entitlement` to change behavior relative to granted vs. revoked entitlements.
- ◆ Changed all usages of “nodeset” to “node-set” to make terminology consistent.
- ◆ Updated `comment` to add support for multiple comments and comment names.

2.6.3 Support for Entitlements

Entitlements created in Designer 1.1 won't work on Identity Manager engines earlier than Identity Manager 3.0.

The following table compares entitlements in Identity Manager 2 with entitlements in Identity Manager 3.0.

Table 2-1 Comparison: Entitlements in Identity Manager 2 and Identity Manager 3

Identity Manager 2	Identity Manager 3
Entitlement definitions existed only as XML in the driver manifest on a DirXML [®] Driver object.	Entitlement definitions are stored as a DirXML Entitlement object contained by a DirXML Driver object.
Entitlements are implemented by using driver policies.	Entitlements are implemented by using driver policies.
Role-based entitlements can use dynamic membership (default) or static include/exclude (additional).	Role-based entitlements can use dynamic membership (default) or static include/exclude (additional).
No workflow entitlements	Provides workflow entitlements.
Entitlements are granted and revoked through the addition of an aux class's attribute to an object.	Entitlements are granted and revoked through the addition of an aux class's attribute to an object.
No status tracking.	Status tracking through the Identity Manager engine into one of the aux class's attributes.
Only one Entitlement Service Driver can run in a driver set.	Only one Entitlement Service Driver can run in a driver set.
A driver set can be associated with only one server.	A driver set can be associated with multiple servers.

You can access the Entitlements Wizard from the Modeler or from the Outline view.

From the Modeler:

- 1 Right-click a driver.
- 2 Select *Entitlements > Add Entitlements*.

From the Outline View:

- 1 Right-click a driver.
- 2 Select *Add Entitlements*.

2.6.4 Support for Workflow-Based Provisioning

Workflow-based provisioning is the process of managing user access to secure resources in an organization. Users request resources, and one or more individuals (including delegates or proxies) with approval rights can approve or deny the request. Users can view the status of requests.

Support for workflow-based provisioning is a key feature of Identity Manager 3. It is not supported in Identity Manager 2.

2.7 Designer and iManager

iManager's primary use is for administration. iManager will continue to be updated with new functionality for managing and monitoring deployed solutions.

iManager's Web-based environment continues to have the following advantages:

- ◆ Remote access
- ◆ Centralized administration
- ◆ Support for roles
- ◆ Integration with other web-based tools

iManager and Designer have similarities, but their features and user experience are optimized for their respective target users and environments. They are compatible. You can export information (for example, a driver set or driver) from one to the other. Also, several key common UI elements are now similar so that you can move between the tools effectively.

Planning Identity Solutions

3

- ♦ [Section 3.1, “Security: Best Practices,” on page 29](#)
- ♦ [Section 3.2, “Solving Identity Issues,” on page 30](#)
- ♦ [Section 3.3, “How to Use Designer,” on page 34](#)

3.1 Security: Best Practices

When using Designer for Identity Manager, observe best practices.

Limiting Rights

Before giving a consultant an Identity Vault administrator password, limit the rights assigned to that administrator to areas of the tree that the consultant must access. Doing so protects sensitive data from being misused, damaged, or unintentionally compromised.

Changing Passwords

After a consultant has completed work, change the password of the user (for example, admin) that the consultant used. If you created a special user (for example, consadmin) for that consultant to use, delete that user or change passwords. Doing so restricts access and brings closure to the consulting process.

Deleting or Saving .proj Files

Delete the project files (.proj) or save them to a company directory.

Designer .proj files are to remain at the company’s project site. A consultant does not take the files after completing a project.

Deleting Files That Aren’t Needed

After project files, log files, and trace files are no longer needed, delete them. These files might contain sensitive information.

Verifying Files

Before discarding or surplus a laptop, verify that project files have been cleaned. Otherwise, someone might discover sensitive information.

Securing Connections

Ensure that the connection from Designer to the Identity Vault server is physically secure. Otherwise, someone could monitor the wire and pull sensitive information.

Handling Documents

When you create documents by using the Document Generator, take care with those documents. These documents can contain passwords and sensitive data in clear text.

Working with Encrypted Attributes

If Designer needs to read or write to an eDirectory™ attribute, you can mark the attribute to be encrypted and allow the attribute to be read over the wire in clear text. This means that you can have encrypted attributes, but they cannot be retrieved securely.

Handling Sensitive Passwords

Do not store passwords that are sensitive.

At this time, Designer projects are not encrypted. Passwords are only encoded. Therefore, do not share Designer projects that have saved passwords.

To save a password for a session, but not save it to the project:

- 1 In an expanded Outline view, right-click an Identity Vault.
- 2 Select *Properties*.
- 3 On the Configuration page, type a password, then click *OK*.

You can enter a password once per session. After you close the project, the password is lost.

To save a password to the project, complete Step 1 through Step 3, select *Save*, then click *OK*.

Security When Using the User Application

For information on security issues relating to the User Application, see the [Security section \(http://www.novell.com/documentation/idm/index.html?page=/documentation/idm/agpro/data/b2gx72y.html\)](http://www.novell.com/documentation/idm/index.html?page=/documentation/idm/agpro/data/b2gx72y.html) in the *Identity Manager User Application: Administration Guide (http://www.novell.com/documentation/idm/index.html)*.

3.2 Solving Identity Issues

- 1 Identify target goals.

Make sure that you and others working on the project use the same terminology. For example, password synchronization and single sign-on are two different technologies.

Also, make sure that the meaning and usage of “metadirectory” are clear as you communicate with others on the project.

Clearly write down the big picture of what you are trying to do.

- 2 Discover the business process.

The process that you are trying to automate with Identity Manager is often already defined and written down.

Be careful. Even if a process exists, people commonly don’t fully understand how it works.

Clearly write down what the business process is, and clarify the process to others.

- 3 Identify all systems.

Identify the systems and all connectivity interfaces with existing systems. Doing this can lead to an increase or decrease in the originally planned systems that you had in mind.

Keep in mind that telephony systems often don’t have an easy API interface.

SQL-based systems sometimes have a very complex database relationship model. It's possible that nobody in your environment understands how the model actually works.

Identify how many Identity Vaults you will need, where they will be located, and what they will connect with.

What to Do In Designer: You can use the Designer Architect Modeler while you are doing this process. It's very helpful to build a diagram of your enterprise and capture key information about each of your systems and each of the users that administer those systems.

As you start a project, keep in mind that the "people management" aspect is vital to your ultimate success. As you identify systems, you need to start the wheels in motion for getting access to those systems in the various departments of the organization.

4 Define high-level data sync flow.

Define the different information types that you want to synchronize. It's common to have different employee types (for example, internal or external).

Be aware of the relationships and dependencies between information of different types (for example, employee, manager, organization, or location).

Doing this step well is critical in helping you to define your data and policy needs.

What to Do In Designer: You can design these data flows in the Dataflow view while you are in the Architect mode or Developer mode.

5 Define objects and attributes to synchronize.

After knowing the data and relationships, you can break the data into specific Identity Vault and Application object types and attributes.

Define merging behavior, passthrough, notification, and reset of attributes.

Identify and derive additional operations based on an original event.

Identify attributes for match, creation, and placement behaviors.

What to Do In Designer: You can use the Task view to capture all of this information or add it to a section in a Document Generation form. One advantage for having it in the Task view is that you can then mark off your tasks as you move into the implementation phase and write the policy to deliver on your requirements.

You can also include any file (word processing document, spreadsheet, presentation) in your Designer project through the Navigator view. By default, the Navigator view launches the appropriate native application to edit and view that document in a way that integrates into Designer, as long as the platform supports it. If you prefer to, you can also configure Designer to use a different editor. This view gives you many options for the types of documents you can include and edit in a Designer project.

6 Write a Requirements Assessment Document.

Document all of the above findings.

Write a high-level acceptance proposal.

Describe the infrastructure.

Create a project timeline.

What to Do In Designer: You can use Designer's Document Generator to capture your high-level design, systems, and basic data flows. You can also fill in sections in the document or create your own sections that describe the project.

7 Fill in the technical details.

Take the high-level diagram that you created in the Architect mode and transition it to a detailed diagram that you can configure your entire solution with.

Configure servers and authentication information on the Identity Vaults.

Create driver sets and drivers. Determine how many driver sets are on each vault. Define what drivers are associated with each driver set, and what servers the driver set is running on. Run the Driver Configuration Wizard on each driver.

Develop driver policies. Make sure that you define and use Global Configuration Values (GCVs) as much as possible. Doing this leads to easier maintenance.

What to Do In Designer: With the Modeler in Developer mode, use the Outline view, Policy Flow view, and properties dialog boxes on all objects, Filter editor, Schema Map editor, E-mail Template editor, Policy Builder editor, and other functionality.

NOTE: The preceding six steps assume that you are building an identity solution from the beginning.

However, Designer is also intended for working on existing solutions. You can create a project and import existing driver sets, drivers, policies, and configurations. Designer lets you easily diagram and manage all components as one complete solution. You can edit existing policies or configurations, add new elements, and then redeploy.

Using Designer on existing solutions has several advantages. You can do the following:

- ◆ Work in a high-productivity development environment.
- ◆ Easily work off-line.
- ◆ Have everything on your local file system for versioning and backup.
- ◆ Simulate.
- ◆ Generate documentation for your projects.

8 Unit test.

Use project-proposal documents and requirements as a basis.

Test all policies. In addition to the Designer tools, you can deploy the project (or parts of it) into a test environment to test more fully.

Generate a report on results.

What to Do In Designer: Use the Designer Policy Simulator to do much of your policy simulation and debugging. It is very easy to use, helps you catch a lot of issues, and is ideal for off-line development.

9 Build a production environment.

As much as possible, mirror existing production environments.

Make sure of the following:

- ◆ All Identity Manager engines and drivers are installed.
- ◆ All servers have correct IP addresses.
- ◆ The applications your drivers will connect to are in place.

As an extra precaution, you might want to deploy to a full test environment or staging environment first. If this is the case, all of the above steps apply to configuring that environment.

What to Do In Designer: To make the next steps easier, you can click a button, get a full PDF document of all of the configuration details of your project, and use that document as you set up your machines and software.

10 Make sure that the data is clean before you deploy.

Ensure the following:

- ◆ Each user has a unique and correctly formatted ID.
- ◆ All users and other objects are where you expect them to be.
- ◆ No duplicate objects exist.
- ◆ No extra or unnecessary objects exist.

11 Deploy to production.

Make sure that the production systems are working well. For example, machines are up and running.

Before you deploy, make sure you have done adequate testing on your project.

If you have previously configured the project to deploy to a separate test or staging environment, you need to change the IP addresses on each Identity Vault (and maybe some GCVs) to successfully deploy to the production environment.

After you deploy, quickly test and redeploy all or any part of the project with any fixes you need.

What to Do In Designer: In the future, more will be done in Designer to help in this area. For now, the following tips can help in this process.

- ◆ Do a full copy of the project in the Project View.
This lets you fill in the new IP addresses for production and leave the old ones alone, so that you can have a project called Staging and another called Production. The disadvantage is that if you change something such as a policy in one, you need to change it in the other to keep the policies in sync. Designer does let you copy/paste between projects to make synchronization easier.
- ◆ Always use GCVs to keep your policies easily portable.
If you have a value that is environment-dependent, you only need to edit a GCV when switching to a different environment.
- ◆ Use the Modeler's Table editor.
This editor is a quick and easy way to get to objects of a certain type (for example, all of your Identity Vaults or all of your servers), see their values, and edit them.
- ◆ Document your project.
Document Generation is always available to help you understand your configurations.

12 Document, back up, and maintain the production system.

What to Do In Designer: Run the Document Generator and save the document.

Make a copy of your project in the Project view. A copy is useful in case you ever need to return to that state or set up a production system again for any reason.

Over time, you will certainly add systems and policies, and make changes in your production environment. You can use iManager for the management of your system (especially if the change is more of an administrative task) and then reimport those changes into your local Designer project.

If the changes you need to make are more suited for a development environment, (for example, policy debugging, policy development, or adding new systems) then continuing to use Designer is the best practice. After each change, you can redeploy the project or just the elements that have changed.

As you version and extend your project over time, remember to always use Document Generation as your paper trail.

3.3 How to Use Designer

- [Section 3.3.1, “Designing New Solutions,” on page 34](#)
- [Section 3.3.2, “Redesigning Existing Solutions,” on page 34](#)
- [Section 3.3.3, “Working on Live Systems,” on page 35](#)

3.3.1 Designing New Solutions

- 1 Launch Designer.
- 2 Using the New Project Wizard, open or create a project.
- 3 Drag and drop items from the Modeler Palette.
- 4 Edit the items’ settings and configurations (for example, names, IP addresses, and passwords).
- 5 Using the Policy Builder and other policy editors, write a policy for sharing data between systems.
- 6 Check your work.
- 7 Use the Document Generator to generate project documentation, and review the documentation with others.
- 8 Deploy the project to live systems.

3.3.2 Redesigning Existing Solutions

- 1 Launch Designer.
- 2 Open or create a new project.
- 3 Import Identity Management elements (drivers, driver sets) from working directories.
- 4 Edit the element settings and configurations (for example, names, IP addresses, and passwords).
- 5 Using the Policy Builder and other policy editors, write a policy for sharing data between systems.
- 6 Check your work.
- 7 Use the Document Generator to generate project documentation, and review the documentation with others
- 8 Deploy the project back to live systems.

3.3.3 Working on Live Systems

The fundamental model/paradigm of Designer is to import solutions, configure them locally in the Designer environment, and then deploy when a solution is ready. This approach was taken to optimize the experience for the following:

- ◆ Off-line work
- ◆ Highly productive development
- ◆ A safer model of configuring and testing before touching your live system

The tool is aimed at developers and architects as the primary audience.

For convenience, Designer enables you to do key operations that a developer typically needs to do, so that the developer can be effective: browse live trees, check driver status, start/stop drivers, and configure driver security. Functionality in the Live Operations options takes immediate effect on the end systems. These operations are clearly grouped in the Designer UI, so that you can tell if you're doing something live.

Installing Designer

4

- ♦ [Section 4.1, “Requirements,” on page 37](#)
- ♦ [Section 4.2, “Installing,” on page 37](#)
- ♦ [Section 4.3, “Uninstalling Designer,” on page 40](#)

4.1 Requirements

- ♦ [Section 4.1.1, “System Requirements,” on page 37](#)
- ♦ [Section 4.1.2, “Platform Requirements,” on page 37](#)

4.1.1 System Requirements

- ♦ Minimum resolution: 1024 x 768. The recommended resolution for Designer is 1280 x 1024.
- ♦ 512 MB RAM (recommended).
- ♦ 1 GB available disk space (recommended).
- ♦ 1 GHz processing speed.
- ♦ Gettext utilities (Linux only).

When you install support packages for Designer (NDSbase and NICI), certain Linux core utilities are needed. The GNU gettext utilities provide a framework for internationalized and multilingual messages. Before installing Designer, make sure that you have installed this package. You can use YaST to check for dependencies and installed packages.

4.1.2 Platform Requirements

You can install Designer on the following:

- ♦ Windows 2000 Professional SP4
- ♦ Windows XP SP2
- ♦ Windows Server 2003
- ♦ Red Hat Fedora* Core 5
- ♦ Red Hat Linux 4.0
- ♦ Novell® Linux Desktop 9
- ♦ SUSE® Linux 10.1
- ♦ SUSE Linux Enterprise Desktop 10
- ♦ SUSE Linux Enterprise Server 10

4.2 Installing

- ♦ [Section 4.2.1, “Installing Designer If Eclipse Is Not Already Installed,” on page 38](#)
- ♦ [Section 4.2.2, “Installing Designer If Eclipse Is Already Installed on Your System,” on page 40](#)

4.2.1 Installing Designer If Eclipse Is Not Already Installed

- ♦ “Downloading the Files” on page 38
- ♦ “Installing on Windows” on page 38
- ♦ “Installing on Windows Vista” on page 39
- ♦ “Installing on Linux” on page 40

Because Designer runs in an Eclipse environment, Designer installs Eclipse on your workstation.

Downloading the Files

- 1 Download the correct `.iso` image or `.zip` file for your system.

Downloads are available at the following locations:

- ♦ Novell Downloads (<http://download.novell.com/index.jsp>)
- ♦ Cool Solutions (<http://www.novell.com/cool solutions/dirxml/designer/>)

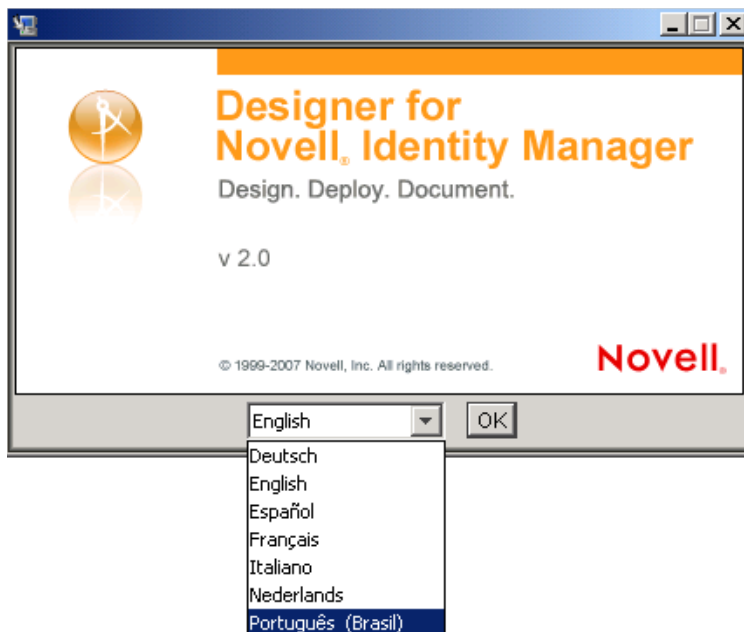
- 2 If you downloaded a `.iso` image, burn that image file to an Identity Manager CD or DVD.

Installing on Windows

- 1 From the `designer_install` directory, run `install.exe`.

The `\nt` directory also contains an `install.exe` file. However, that file is for the Metadirectory system. Use the `install.exe` file found in the `designer_install` directory.

- 2 Select a language to use during the installation, then click *OK*.

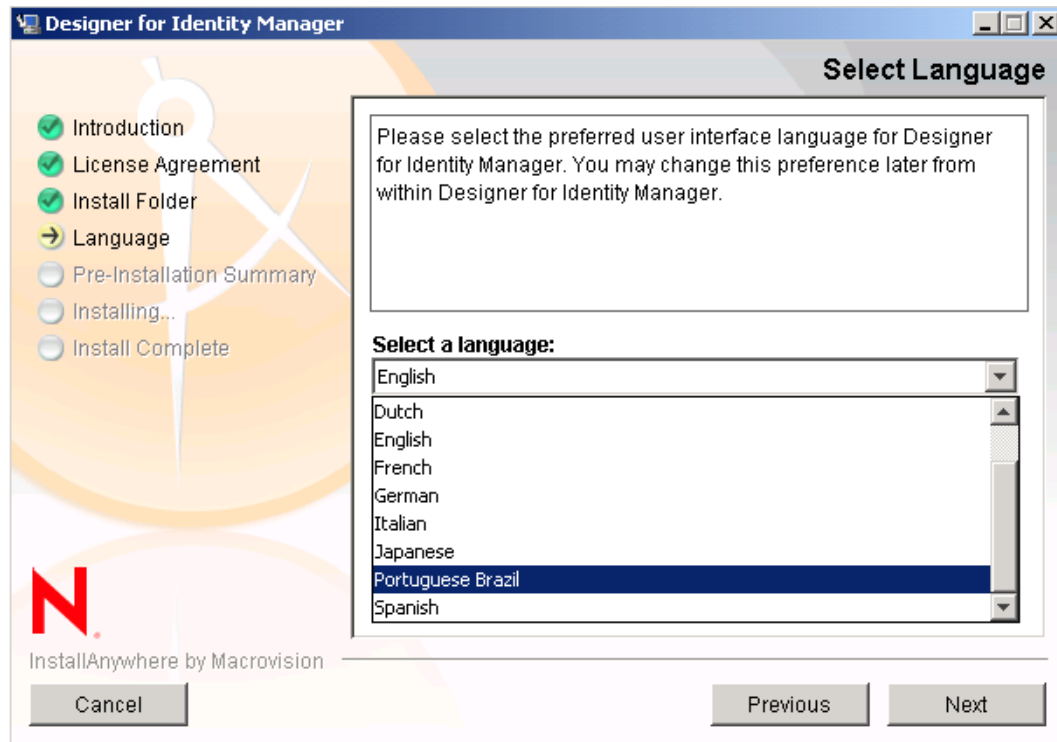


The default language is English. You can select a different language from the drop-down list.

The languages displayed in the language-selection dialog box are filtered, based on the default system locale. If the default system locale is English (or another Latin-based character set), only languages that use the Latin character set are displayed.

If the default system locale is set to Japanese (or any double byte-locale), only the double-byte locales, such as Chinese, Korean, and Japanese, are displayed in the list. The exception to this is English, which is always in the list, regardless of the default system locale.

- 3 Review the Introduction, accept the license agreement, select a location for the Designer files, then select shortcuts to Designer.
- 4 Select a language for the Designer user interface.



The default is the language that you selected to use for the installation. You can select a different language from the drop-down list.

After installation, you can change the language. See [Section 23.3.7, “Changing the Language Setting,” on page 460](#).

- 5 Review the summary of choices, then install Designer.
- 6 Review known issues in the Readme, then complete the installation.
- 7 To run Designer, click the *Designer* icon on the desktop.

Installing on Windows Vista

Designer 2.1 is not officially supported or tested on Windows Vista. However, our initial impression is that Designer runs well on Vista.

During the install, you see a message stating that you are installing on an unsupported Windows version. You can continue with the installation by selecting *OK*.

Windows Vista has implemented a new User Account Control feature that prevents applications from running as Administrator unless you specifically allow it. If Designer does not launch after installation on Windows Vista:

- 1 Right-click the Designer shortcut.
- 2 Select *Run as Administrator*.

Installing on Linux

Installing eDirectory™ base components and NICI requires `root` privileges. The install program prompts you for the `root` password so that you can install these components.

To install on Linux:

- 1 From the Identity Manager CD or `.zip` file, run `/linux/designer/install.bin`.
The `/linux` directory also contains an `install.bin` file. However, that file is for the Metadirectory system. Use the `install.bin` file found in the `designer` directory.
- 2 To run Designer, click the *Designer* icon on the desktop.

4.2.2 Installing Designer If Eclipse Is Already Installed on Your System

Because of the interaction between Designer and third-party Eclipse plug-ins, Novell does not support installing Designer into an existing Eclipse installation.

For information on installing Designer into an Eclipse environment, see the [CoolSolutions Web site](http://www.novell.com/coolsolutions/dirxml/designer/existinginstall_instructions.html) (http://www.novell.com/coolsolutions/dirxml/designer/existinginstall_instructions.html).

4.3 Uninstalling Designer

- ♦ “Uninstalling on Windows” on page 40
- ♦ “Uninstalling on Linux” on page 40

Uninstalling on Windows

- 1 In the Control Panel, select *Add/Remove Programs*.
- 2 Click *Designer for Identity Manager > Change/Remove > Uninstall > Yes*.

To easily uninstall on English-language workstations, select Uninstall from the Start menu. For example, on Windows 2000, click *Start > Programs > Designer for Identity Manager > Uninstall*.

Uninstalling on Linux

- 1 Make sure that you have the correct privileges to uninstall.
- 2 Run `Uninstall_Designer_for_Identity_Manager`.
This file is in `[path you chose to install into]/designer/UninstallDesigner`.

Getting Help

5

Information about Designer is in the local help system, in the `readme.txt` file, and in the online documentation. Information in the help system is not localized. If you need localized information, refer to the *Designer 2.1 for Identity Manager 3.5.1* guide on the [Novell Documentation \(http://www.novell.com/documentation/designer21/index.html\)](http://www.novell.com/documentation/designer21/index.html) Web site.

- ◆ Section 5.1, “Help from the Welcome Page,” on page 41
- ◆ Section 5.2, “Browsing for Information,” on page 43
- ◆ Section 5.3, “Getting Context-Sensitive Help,” on page 43
- ◆ Section 5.4, “Help from the Web,” on page 44
- ◆ Section 5.5, “Viewing Demos,” on page 45
- ◆ Section 5.6, “Searching for Information,” on page 45
- ◆ Section 5.7, “Customizing the Browser and Help Server,” on page 46

5.1 Help from the Welcome Page

When Designer starts, a Welcome page appears and makes it easy for you to get started.

Figure 5-1 The Welcome Page



Table 5-1 Options on the Welcome Page

Option	Description
Overview	Provides an overview of Designer and Identity Manager.
What's New	<ul style="list-style-type: none">◆ Describes what's new in this release.◆ Enables you to update Designer.◆ Links to the readme file.
Tutorials	Launches Flash demos of Designer tasks.
Web Resources	Provides links to online information and downloads.
Run Designer	Takes you to Designer's Modeler.

The icons on the toolbar take you to the Welcome page, toggle between previous and next pages, and enable you to customize the Welcome page.

Figure 5-2 Icons on the Toolbar



You can close the Welcome page at any time by clicking the *X* next to *Welcome*.

Figure 5-3 The Close Icon

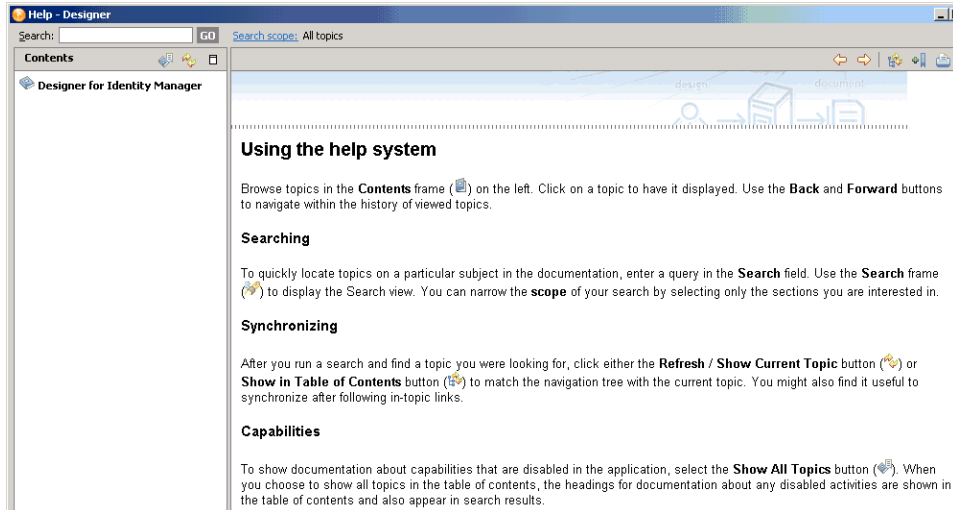


To return to the Welcome page, click *Help > Welcome*.

5.2 Browsing for Information

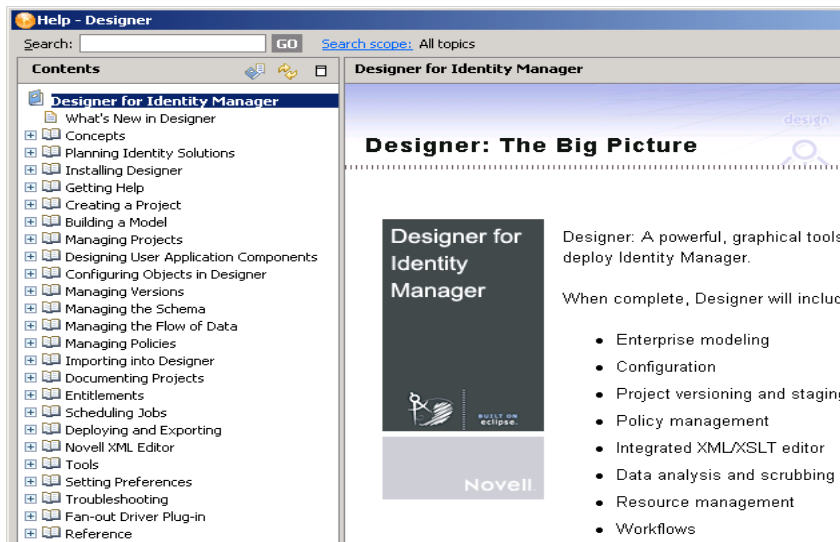
To navigate all of the help topics, you can use a built-in Web browser:

- 1 From the main menu, select *Help > Help Contents*.




The text under *Using the help system* describes the icons that help you navigate and use the Help system.

- 2 In the Contents pane, select *Designer for Identity Manager*.



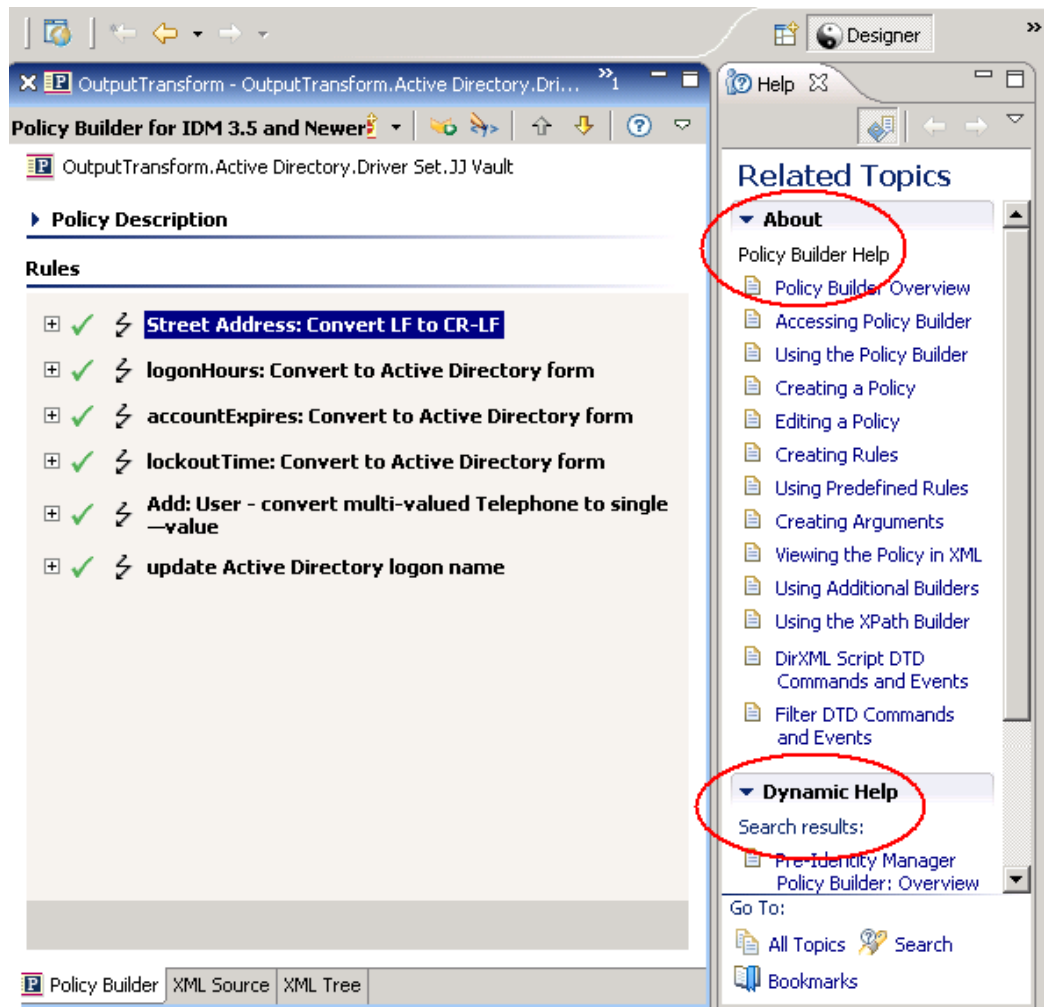
- 3 Navigate to the topic that you're interested in.

5.3 Getting Context-Sensitive Help

- 1 Press F1 or click the help icon .

NOTE: If you maximize an editor (for example, the Modeler), help topics do not display when you press F1. To view the help, minimize the editor.

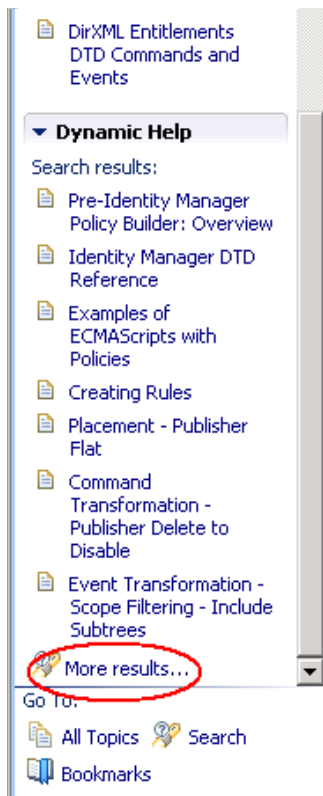
- 2 Click links under *About* and *Dynamic Help*.



5.4 Help from the Web

To view results found through Google* or in the Eclipse help system, click *More results* > *Web Search*.

Figure 5-4 Related Topics in Google and Eclipse



5.5 Viewing Demos

To see how Designer works, view the Flash demos that are available on the Welcome page:

- 1 Select *Help > Welcome*.
- 2 Select the *Tutorials* icon.



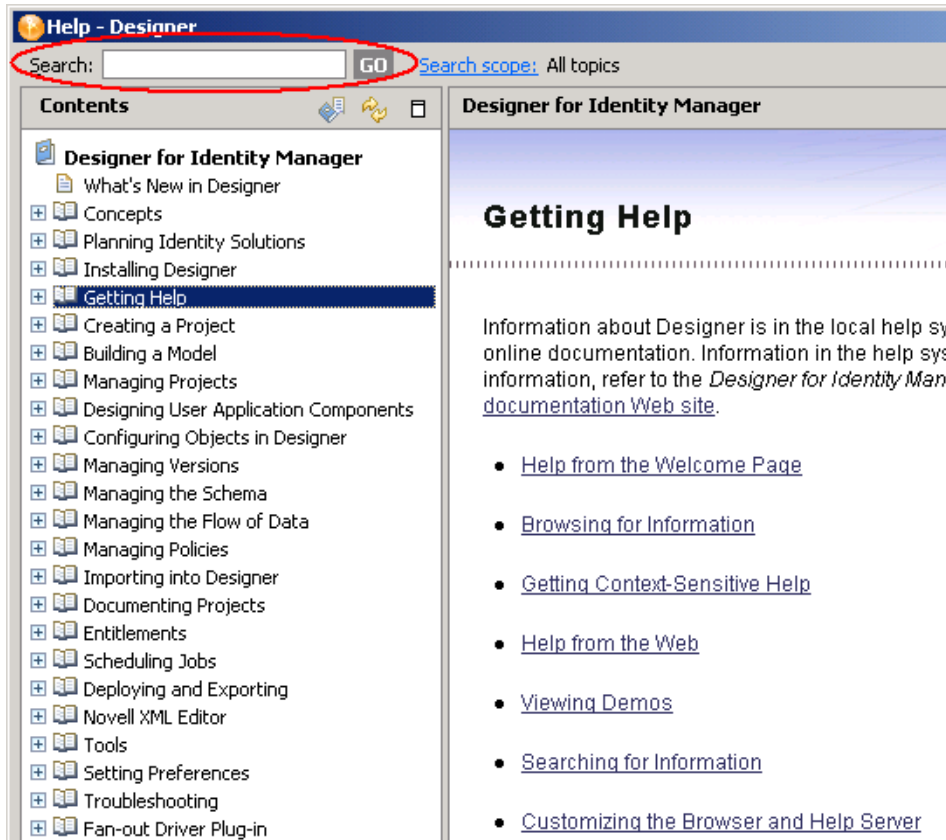
- 3 Select a demo:
 - ♦ Model Projects
 - ♦ Import
 - ♦ Create a Workflow
 - ♦ E-Mail Templates
 - ♦ Manage and Simulate Policies
 - ♦ Document Generation Overview

5.6 Searching for Information

- 1 Select *Help > Search*.

2 Type a topic in the *Search* field, then click *Go*.

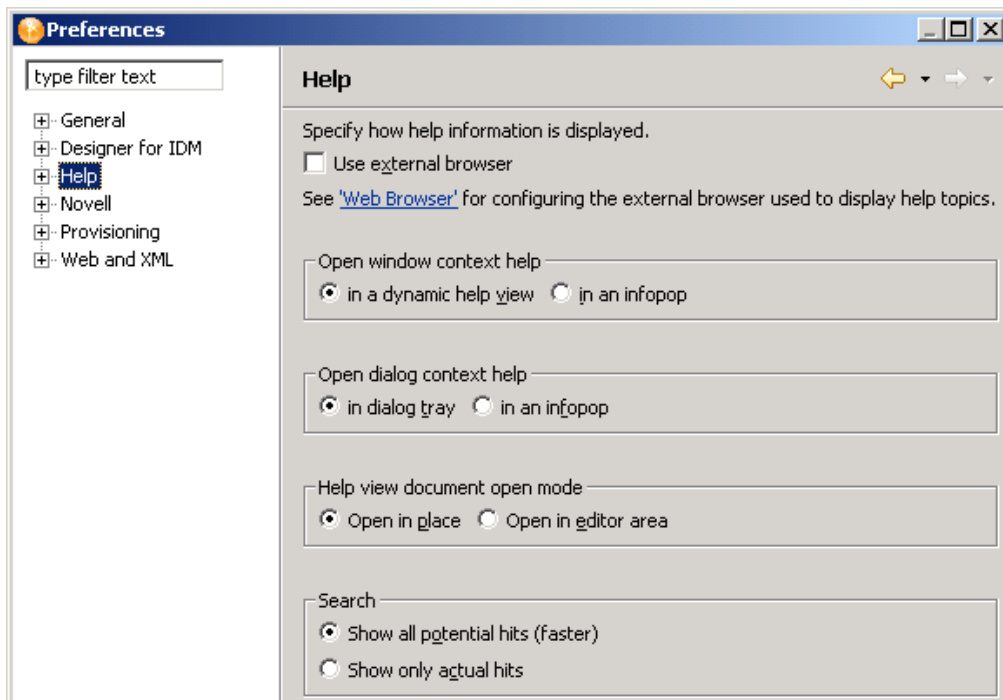
You can also select *Help > Help Contents*, then use the Search option above the Contents pane, or click *Search* at the bottom of the context-sensitive Help view.



5.7 Customizing the Browser and Help Server

1 From the main menu, click *Window > Preferences*.

2 Select *Help*, make changes, then click *OK*.

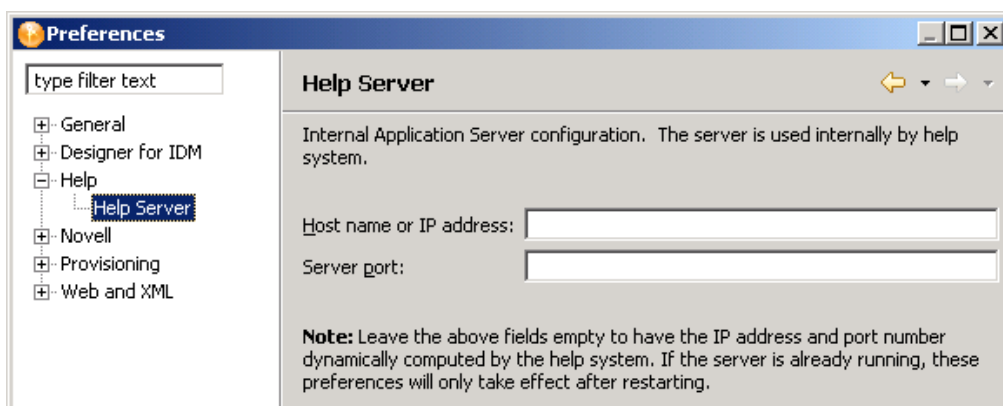


Because Designer is built on Eclipse, you are able to use Help Server functionality. It is a development tool for building your own tools. If you have a team environment where the entire team is using Eclipse, you might want to set up a common Help server, and then use that server for help files and content in your intranet.

For more information on Help and the Help server:

- ♦ See the Help section in Preferences.
- ♦ Go to the Eclipse Web site.

3 (Optional) Expand *Help*, then click *Help Server*.



4 (Optional) Make changes concerning the Help Server, then click *OK*.

Creating a Project

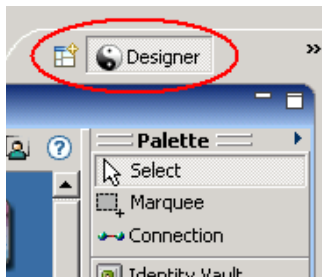
6

IMPORTANT: Projects created in Designer releases earlier than Designer 3.0 work in Designer 3.0. However, projects created in Designer 2.1 or 3.0 don't work in Designer 1.1 or earlier releases.

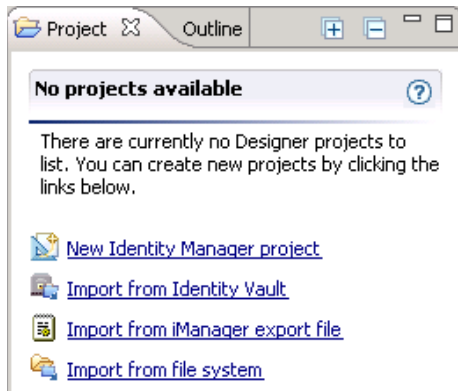
- ♦ [Section 6.1, “When No Project Exists,” on page 49](#)
- ♦ [Section 6.2, “When You Want to Create an Additional Project,” on page 50](#)
- ♦ [Section 6.3, “When You Want to Import a Project,” on page 51](#)

6.1 When No Project Exists

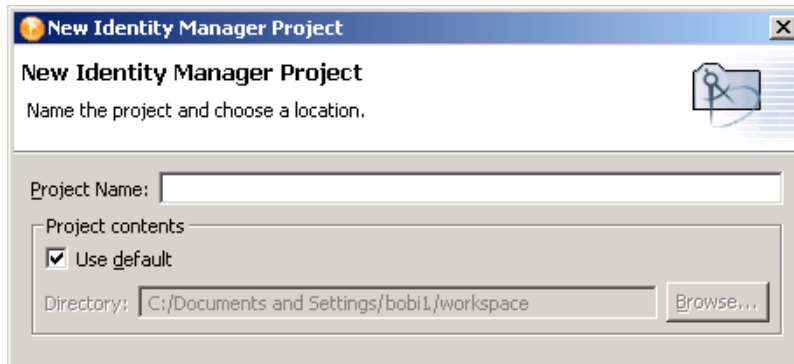
- 1 Make sure that the Designer perspective (in the upper right corner) is selected.



- 2 If you are just starting Designer and have no projects in the Project tab, you see the following window:



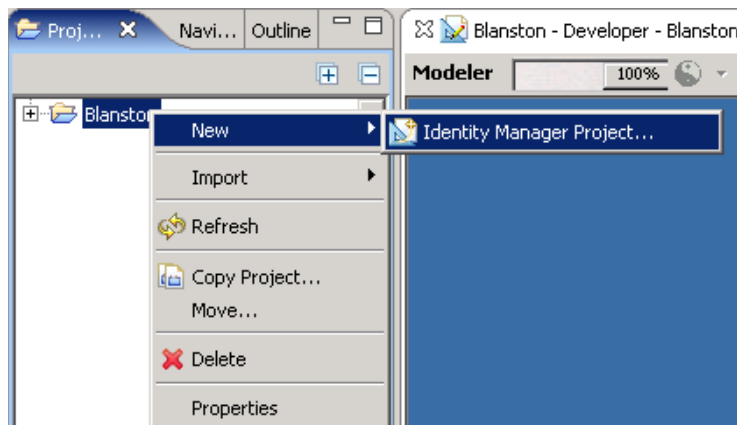
- 3 Click *New Identity Manager Project* to launch the Identity Manager Project Wizard.



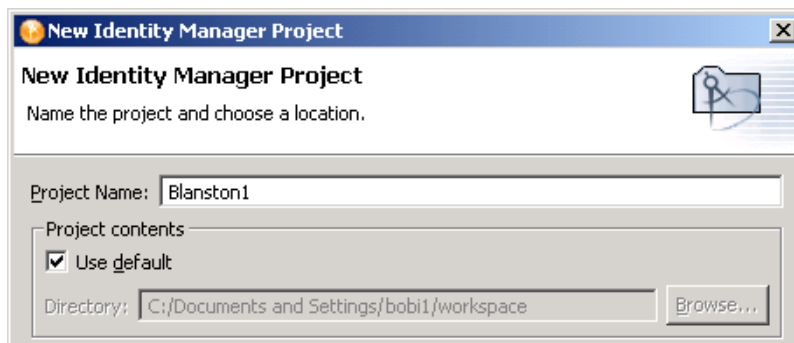
- 4 Name the project, then click *Finish*.

6.2 When You Want to Create an Additional Project

- 1 Right-click in the Project view pane, then click *New > Identity Manager Project*.



- 2 In the Identity Manager Project Wizard, name the project.

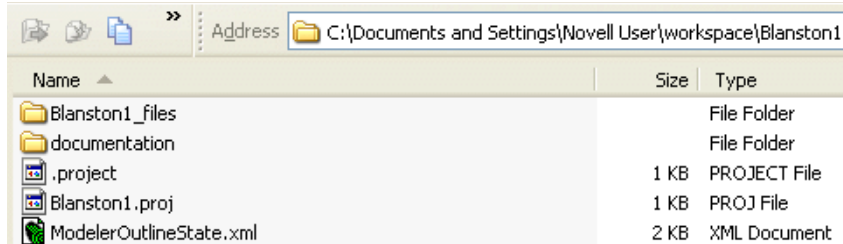


Designer stores the project in a local directory. You specified this directory when you installed Designer. Typically, this default directory is `c:\Documents and`

Settings\Username\workspace. To specify a different directory, deselect *Use Default*, then browse to and select the desired directory.

3 Click *Finish*.

The project is stored as a directory structure with the project name as the initial directory containing files with a .proj and a .project extension. In this example, the project is stored in the c:\Documents and Settings\Novell User\workspace\Blanston1 directory on a Windows workstation.



The project name appears in the Project view. When you select the System Model icon under the project name, Designer opens the Modeler (an editor) for the new project.

For information on saving a project, see “Saving” in [Section 8.4.2, “Tasks You Can Perform from the Project View,”](#) on page 114.

6.3 When You Want to Import a Project

To import a project from an Identity Vault or from the File System, see [Chapter 15, “Importing into Designer,”](#) on page 257.

- ◆ [Section 7.1, “Basic Tasks,” on page 53](#)
- ◆ [Section 7.2, “Accessing the Modeler,” on page 54](#)
- ◆ [Section 7.3, “Selecting a Modeling Mode,” on page 55](#)
- ◆ [Section 7.4, “Working from the Palette,” on page 62](#)
- ◆ [Section 7.5, “Copying and Pasting,” on page 68](#)
- ◆ [Section 7.6, “Moving Items,” on page 72](#)
- ◆ [Section 7.7, “In-Line Editing,” on page 73](#)
- ◆ [Section 7.8, “Tooltips and Toolbar,” on page 73](#)
- ◆ [Section 7.9, “Organizing by Domain Groups,” on page 74](#)
- ◆ [Section 7.10, “Connecting Applications,” on page 82](#)
- ◆ [Section 7.11, “Aligning and Laying Out Components,” on page 88](#)
- ◆ [Section 7.12, “Editing Multiple Objects,” on page 95](#)
- ◆ [Section 7.13, “Modeling Active Directory Domain Controllers,” on page 95](#)
- ◆ [Section 7.14, “Saving Your Model,” on page 96](#)

7.1 Basic Tasks

This section lists basic tasks for creating a model after you have created a project.

1 Select a project.

If your project doesn't appear in the Modeler, open the Project view (*Window > Show View > Project*), expand the project, then double-click *System Model*.

2 Drag an Identity Vault object from the palette to the Modeler.

When you create an Identity Vault or server in Designer 2.1, the default Identity Manager engine version is 3.5. Designer assumes that the Identity Vault has 3.5 capabilities. You can successfully deploy and run 3.5 projects only on Identity Manager 3.5 servers.

You can easily change the engine version by selecting a version from the Server DN field.

Before you deploy a project, you must associate a server with the Identity Vault. You do this through the Identity Vault properties. See [Section 9.3, “Configuring Identity Vaults,” on page 148](#).

You can add multiple Identity Vaults.

3 Configure a driver set.

Each Identity Vault contains a driver set. See [Section 9.5, “Configuring Driver Sets,” on page 152](#).

4 Add applications.

Drag applications from the palette to the Modeler. See [Section 9.8, “Configuring Application Properties,” on page 170](#).

5 Configure drivers.

Driver connections are automatically drawn between the application and the driver set. See [Section 9.6, “Configuring Drivers,”](#) on page 158.

6 Develop and customize your model.

Develop according to what you planned in [Section 3.2, “Solving Identity Issues,”](#) on page 30.

7 Save your model (design).

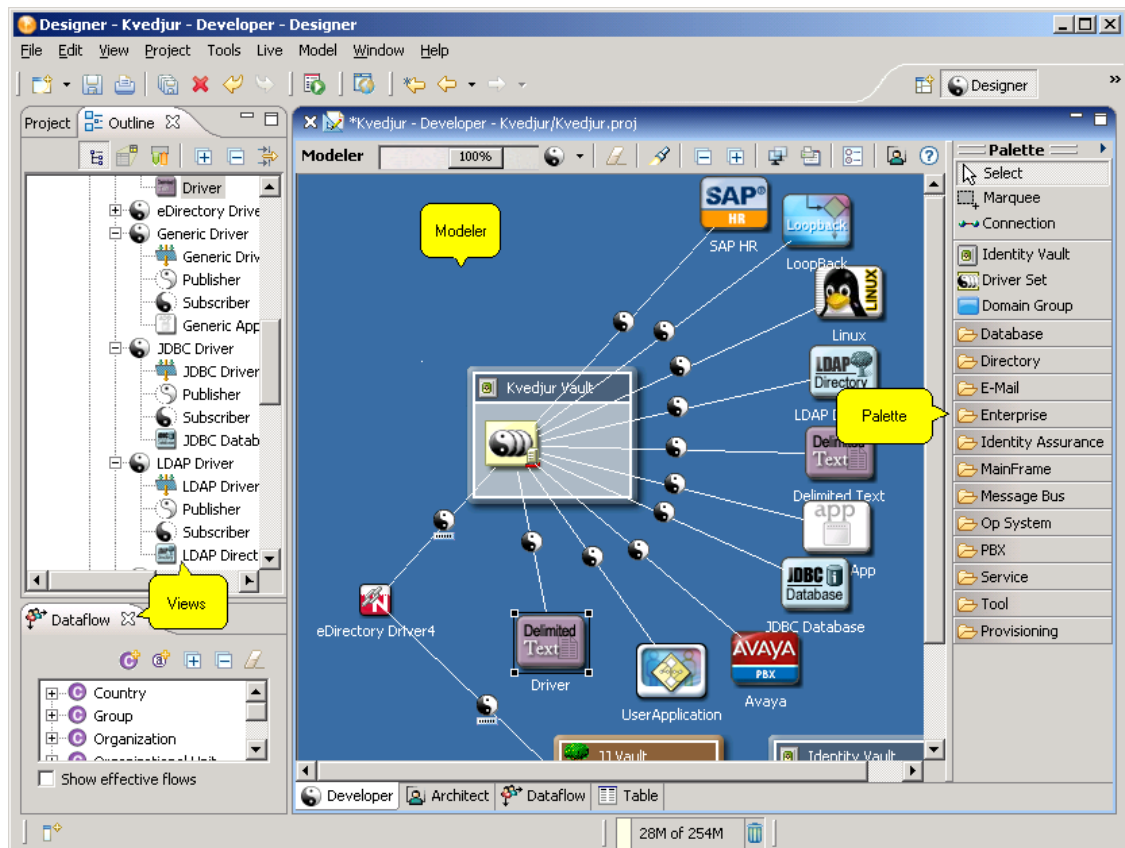
Do one of the following:

- From the main menu, select *File > Save* (or *Save All*).
- From the main menu, select *File > Close > Yes*.
- Click the *X* in the Modeler’s tab, then select *Yes*.

7.2 Accessing the Modeler

The Modeler space is the main working area. It is an editor where you design projects. It is the main work space and primary means of interacting with Designer. All other editors, views, and dialog boxes support and provide functionality for the Modeler.

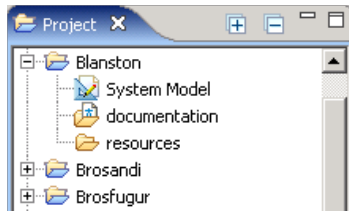
Figure 7-1 Designer’s Modeler



To get started, you create a project and drag items from the palette into the Modeler space. Then you arrange and configure the items.

If the Modeler doesn't display:

- 1 Expand a project in the Project view.



If you haven't yet created a project, create one.

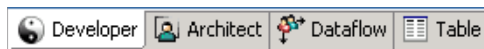
- 2 Double-click *System Model*.

7.3 Selecting a Modeling Mode

- ♦ [Section 7.3.1, “Developer Mode,” on page 56](#)
- ♦ [Section 7.3.2, “Architect Mode,” on page 57](#)
- ♦ [Section 7.3.3, “Dataflow Mode,” on page 61](#)
- ♦ [Section 7.3.4, “Table Mode,” on page 62](#)

The Modeler has tabs along the bottom, so that you can switch among different modeling modes. The modes have different advantages, depending on the task you're trying to do and the role that you are acting in.

Figure 7-2 *Modeler Modes*

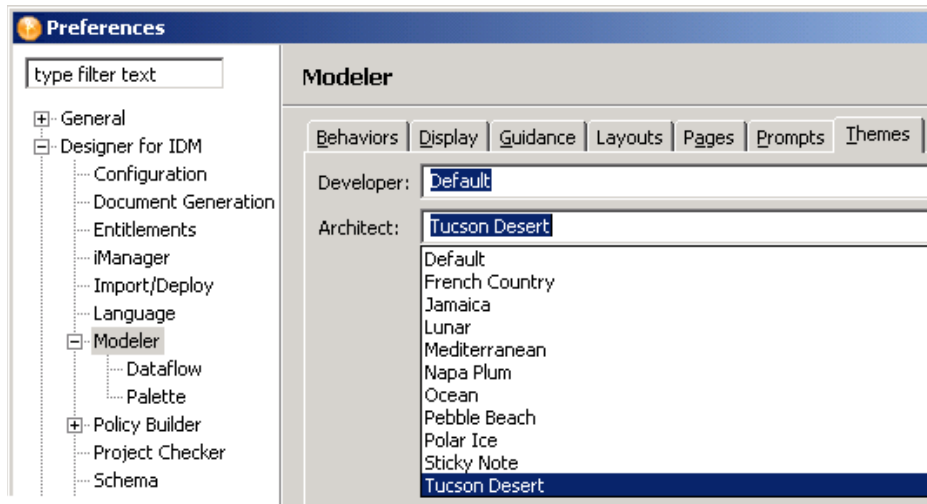


The modes are synchronized with each other with selection, data, and content. They are also synchronized with the Outline View and Thumbnail view.

As you switch modes in the Modeler editor, the editor tab at the top displays the mode that you are in.

By default, the theme preference is different for each mode. To configure each theme independently in the Modeler preferences:

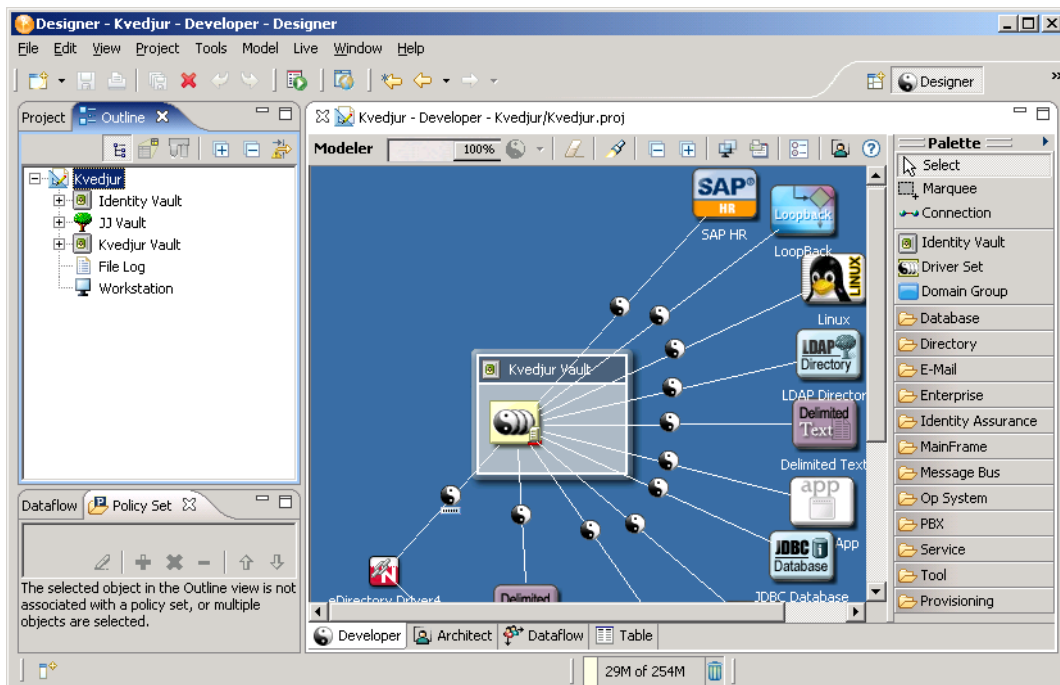
- 1 Click *Window > Preferences > Designer for IDM > Modeler*.



- 2 Click the *Themes* tab.
- 3 Select a theme, then click *OK*.

7.3.1 Developer Mode

Figure 7-3 Developer Mode



Use Developer mode to do all low-level operations with driver sets, drivers, policies, and applications. This mode lets you manage all of the visual elements and configuration details that you need to fully build and deploy an identity solution.

In Developer mode, the palette organizes the applications and systems into categories. You can customize them to show up as one alphabetical list by using the Modeler Preferences. See [“Palette” on page 465](#).

Working with Labels

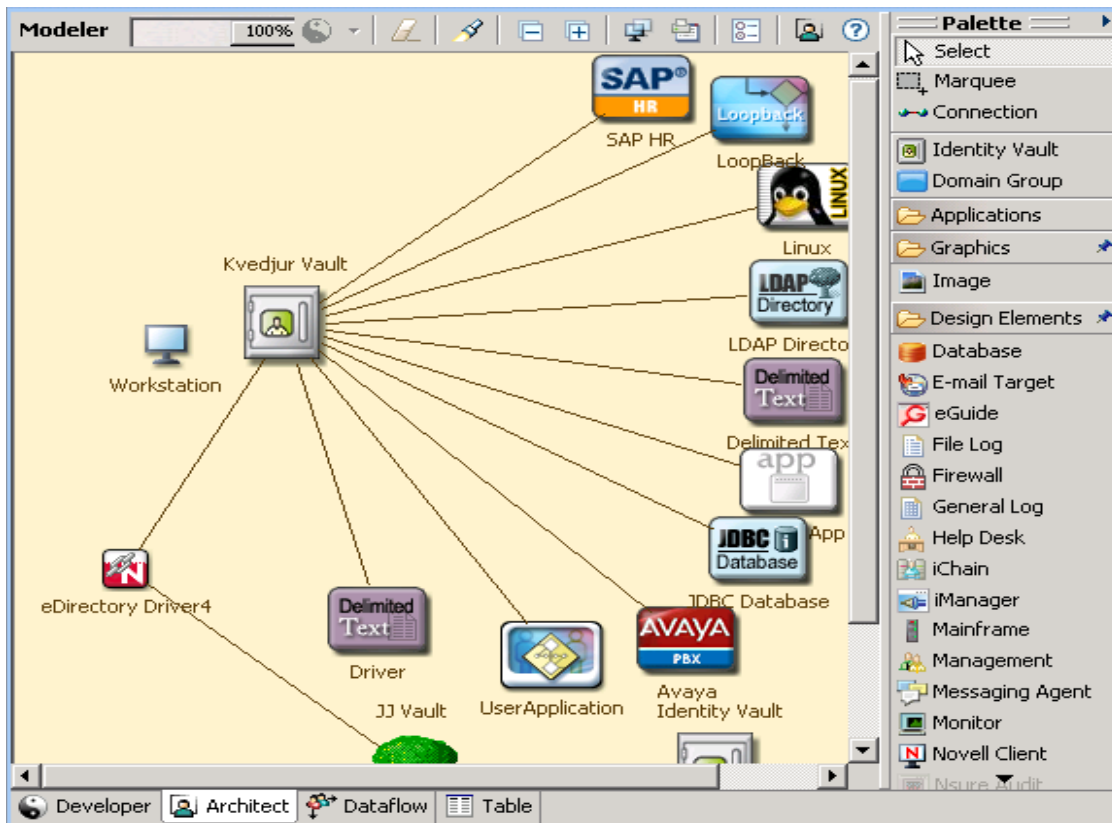
Figure 7-4 An Application's Label



By default in both Developer and Architect modes, labels appear under application icons in the Modeler. They also appear above Identity Vaults in Architect mode. To configure these labels to not appear, use the Modeler Preferences. See [Section 23.3.8, “Modeler,” on page 461](#).

7.3.2 Architect Mode

Figure 7-5 Architect Mode



Use Architect mode to work at a higher level. Because a higher level doesn't show drivers, driver sets, or policies, you focus more on systems. This mode helps you do large-scale design, which is more intuitive to architects and business strategists.

It is quite likely that you will start in this mode when you begin each project. You will probably spend time putting together an accurate diagram of your enterprise as you consult with various people throughout your organization. As you do so, you should capture key information on each system, such as the owner, contact information, machine environment, software versions, and authentication credentials. As you go through this process, you will also define your project requirements, start thinking about your data, and capture that information in your project.

When the time is right, you can switch to the Developer mode and delve into the technical details of building a working solution. Depending on the size of your project and makeup of your team, you could have architects and designers build high-level solutions with Designer in Architect mode, and then send the project to identity developers who understand all the details of writing policies and configuring systems. They can share the same project.

In Architect mode, you can connect any design element with any other design element, application, image, or Identity Vault. The connecting lines enable you to express any relationship, making Architect mode a general-purpose, high-level business modeler. The Architect-mode lines don't display when you switch to the Developer mode.

The design elements have connectivity information tied to them. You can use design elements to do live operations or to remotely control other elements that are in your environment but are not necessarily included in your Identity Manager infrastructure.

The Palette in Architect Mode

In Architect mode, the palette alphabetically lists all applications in one folder and design elements in another folder.

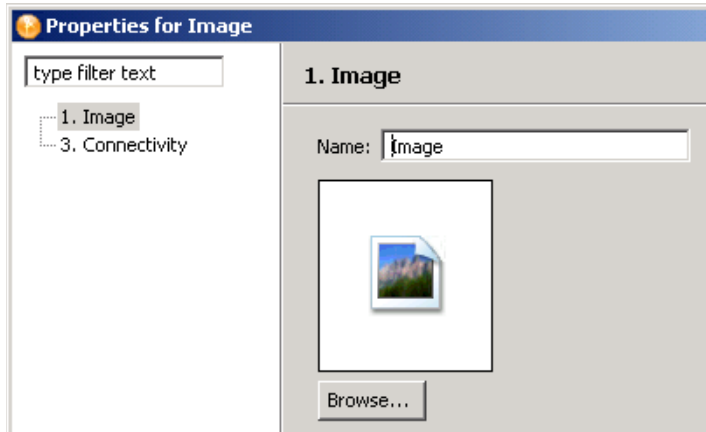
The Graphics folder has an Image icon. When you drag this icon to the Modeler, Designer displays a generic graphic:

Figure 7-6 *The Image Icon*



To edit the properties of this icon:

- 1 Right-click the icon, then select *Properties*.



- 2 In the *Name* field, replace *Image* with a caption.
- 3 Browse to and select a replacement graphic, then click *OK*.

You might need to reduce the size of the graphic before importing it.

After the image is in the Modeler, you can drag it, change it, connect lines to it, align or distribute it, or delete it.

High-Level Data Flows in Architect Mode

To set data flows in Architect mode:

- 1 Right-click the line between an application and an Identity Vault.
- 2 Select *Dataflow*.
- 3 Specify synchronization and notification events, then click *OK*.

This option is used the same way as in Developer mode except that in Architect mode, Designer automatically configures all the details (schema, filters, and mapping policies) for you. You won't see the Data Flow Wizard for these details. Before deployment, you can edit the details by using Developer mode.

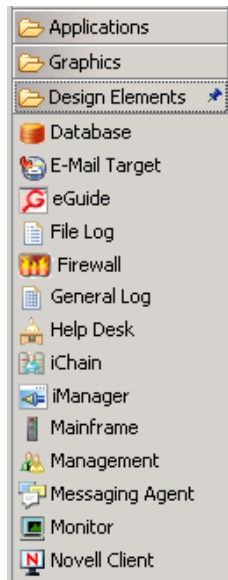
Tasks

You can do the following tasks in Architect mode:

- ♦ Straighten connections (edges). See [Section 7.11, "Aligning and Laying Out Components," on page 88](#).
- ♦ View Password Sync icons and edit synchronization. See [Section 12.6, "Integrating Passwords," on page 245](#).
- ♦ Auto-connect eDir-to-eDir.
- ♦ When deleting the driver line, view a prompt to confirm drivers being deleted.
- ♦ Display design elements in your model.

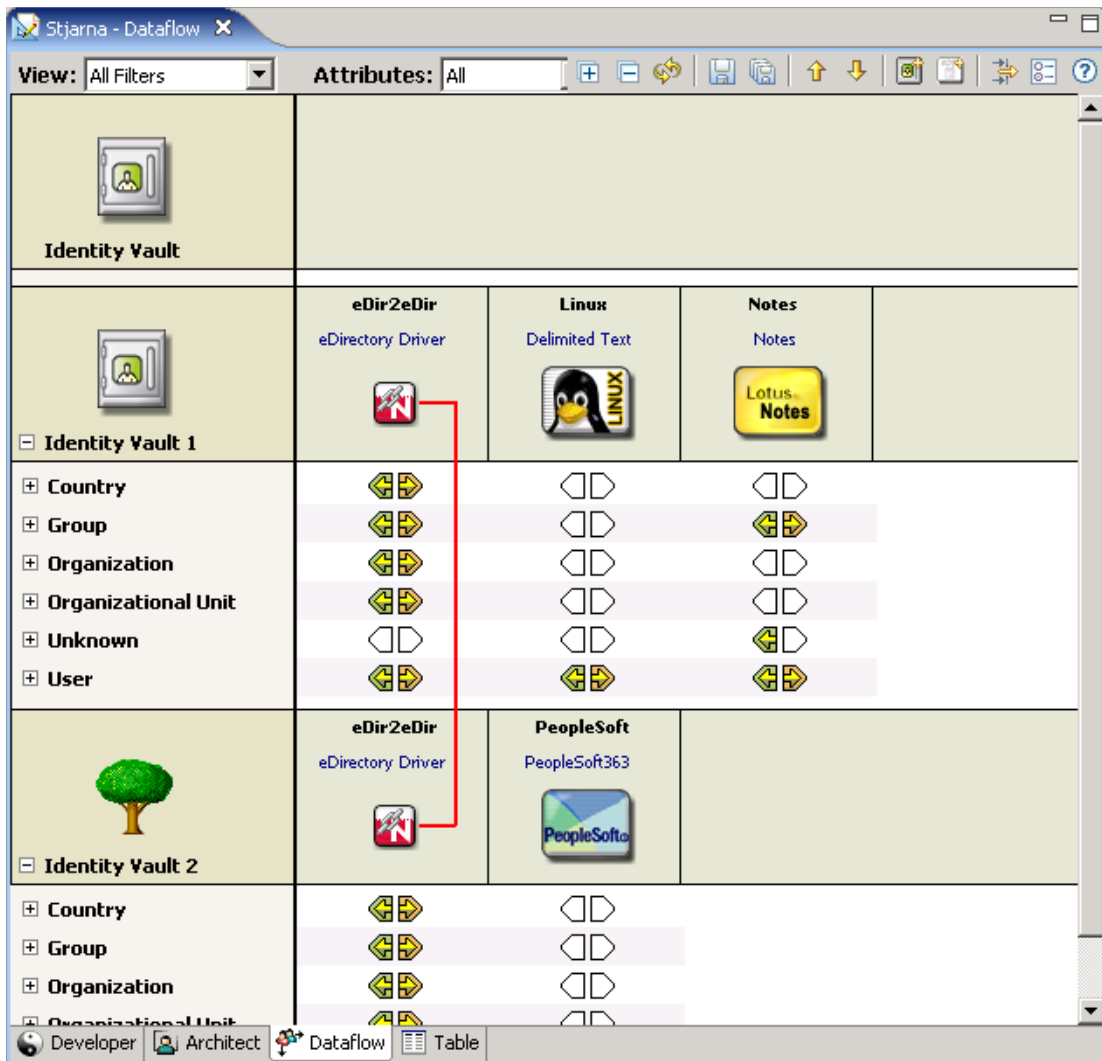
Open the *Design Elements* folder on the palette, drag design elements onto the Modeler, and connect the design elements.

Figure 7-7 *Items in the Design Elements Folder*



7.3.3 Dataflow Mode

Figure 7-8 Dataflow Mode



The Dataflow mode launches the Dataflow Editor, so that you can see all filters that control how data flows between systems and Identity Vaults.

The Dataflow mode is synchronized with the Modeler and with the Outline view when you add, delete, change, or synchronize objects. Also, you can see how passwords flow from each server. See [Chapter 12, “Managing the Flow of Data,”](#) on page 221.

7.3.4 Table Mode

Figure 7-9 Table Mode

Name	Type	Container	Details
Linux	Application	Modeler Workspace	
Notes	Application	Modeler Workspace	
PeopleSoft	Application	Modeler Workspace	
eDir2eDir	Application	Modeler Workspace	
Modeler Workspace	Domain	Stjarna	
eDirectory Driver	Driver	DriverSet	Log Off, Limit 50, Trace None, File None,
eDirectory Driver	Driver	DriverSet	Log Off, Limit 50, Trace 0, File , Trace Na
Notes	Driver	DriverSet	Log Off, Limit 50, Trace None, File None,
PeopleSoft363	Driver	DriverSet	Log Off, Limit 50, Trace None, File None,
Delimited Text	Driver	DriverSet	Log Off, Limit 50, Trace None, File None,
Driver Set	DriverSet	Identity Vault	Context: None, Log Off, Limit 50, Trace M
DriverSet	DriverSet	Identity Vault 1	Context: None, Log Off, Limit 50, Trace M
DriverSet	DriverSet	Identity Vault 2	Context: None, Log Off, Limit 50, Trace M
Delimited Text Filter	FilterPolicy	Delimited Text	Size: 915
eDirectory Driver Filter	FilterPolicy	eDirectory Driver	Size: 7240
eDirectory Driver Filter	FilterPolicy	eDirectory Driver	Size: 7240

Table mode provides a global table editor, which provides a list of all design elements in the project. You can scroll this table to quickly scan essential information. You can efficiently find all items of a particular type and edit their settings.

To edit an entry in the table, double-click a line, or right-click, click *Open With*, then select an editor.

When you select an item in the table, Designer synchronizes the selection with the Outline view, so that you can view the selection's container.

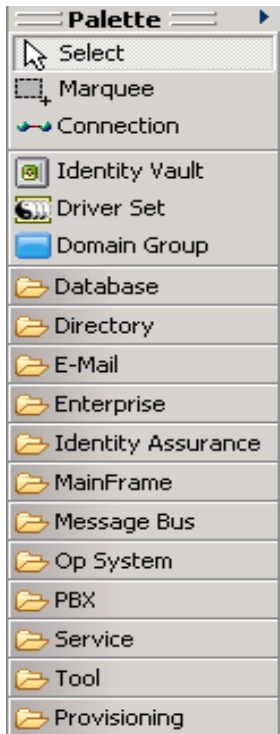
To sort the lists, click a column header.

7.4 Working from the Palette

- ◆ [Section 7.4.1, “About the Palette,” on page 63](#)
- ◆ [Section 7.4.2, “Palette Operations,” on page 63](#)
- ◆ [Section 7.4.3, “Using Generic Applications,” on page 64](#)
- ◆ [Section 7.4.4, “Fly-Out Palette,” on page 64](#)
- ◆ [Section 7.4.5, “Resizing the Palette,” on page 65](#)
- ◆ [Section 7.4.6, “Docking the Palette,” on page 66](#)
- ◆ [Section 7.4.7, “Arranging Folders and Applications,” on page 66](#)
- ◆ [Section 7.4.8, “Changing the Layout,” on page 67](#)
- ◆ [Section 7.4.9, “Keyboard Support for the Palette,” on page 68](#)

7.4.1 About the Palette

Figure 7-10 Designer's Palette



The palette is the source of all of the items that you add into the Modeler. To build a model, do one of the following:

- ◆ Drag and drop items from the palette to the Modeler space. When you drag and drop an application, it auto-connects to the closest driver set.
- ◆ Click an item in the palette, then click in the Modeler space where you want the item to go.

7.4.2 Palette Operations

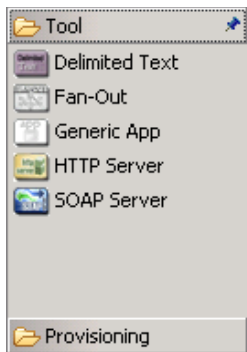
Table 7-1 Palette Operations

Operation	Description
Connection	Connects items in the Modeler space.
Identity Vault	Places an Identity Vault in the Modeler space.
Driver Set	Places an eDirectory Driver Set object in an Identity Vault. All applications that you want to connect use a Driver Set object as a hub between the two applications.
Domain Group	Lets you group and organize items in the Modeler space.

Operation	Description
Folders	Applications are organized within folders or drawers. To open or close a folder, click it. To hold the folder in place and make sure that it doesn't fully collapse (even when you open other folders), click the pin. An unpinned folder automatically closes when you open another folder.
Applications	The various applications that you can connect are grouped into folders by type. You can drag and drop these applications to the Modeler space and begin editing them. The Modeler automatically adds a connecting line, which represents a driver.
Scrolling Arrows	Small directional arrows. If a folder has many items, or if the screen area is restricted, scrolling arrows appear. To scroll through the contents of a folder, click the arrows.

7.4.3 Using Generic Applications

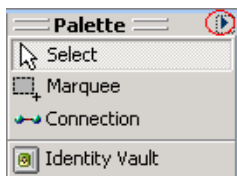
Figure 7-11 *The Generic App Option on the Palette*



Scenario: A Generic Application. Fridrik creates a project with his own items and graphics, in his own version of Designer. He transfers the project to you, but you are using a different version of Designer, which doesn't understand those items. Your version renders the transferred objects as Generic applications.

7.4.4 Fly-Out Palette

Figure 7-12 *The Palette's Control Arrow*



To hide the palette, click the small control arrow on the palette. The palette collapses.

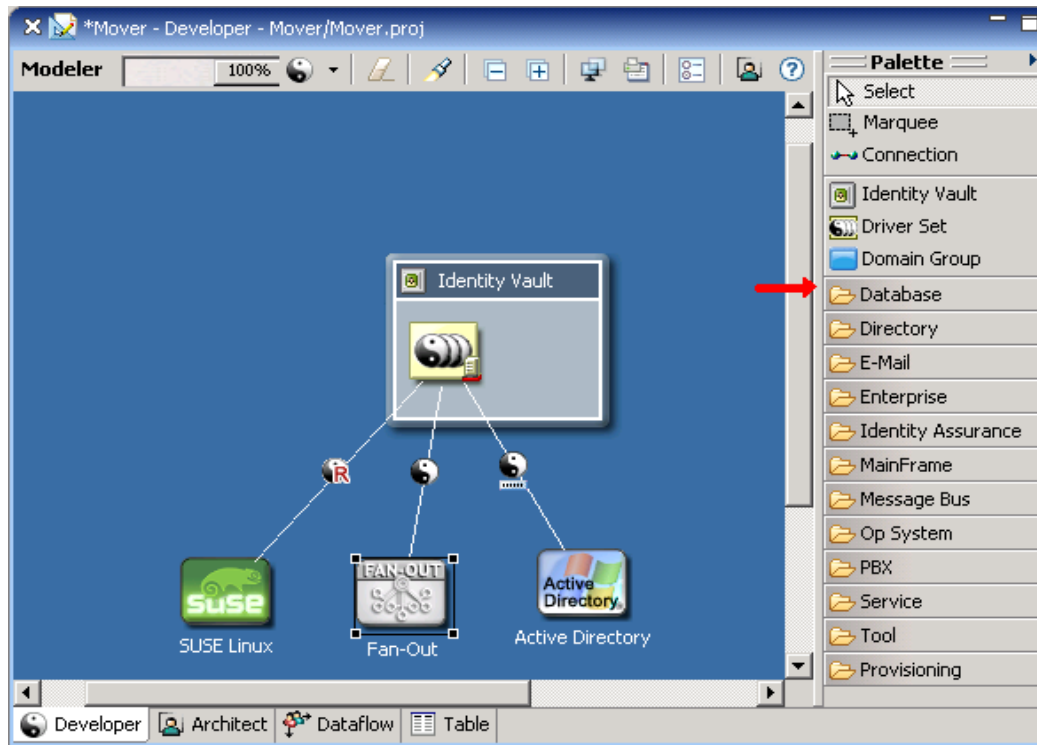
To open the palette again and keep it open, click the arrow.

To temporarily open the palette again, hover the cursor over the collapsed palette, below the control arrow. The palette quickly expands. This is fly-out mode.

To change the palette from fly-out mode, click the control arrow again. The state persists and is restored the next time you run the application.

7.4.5 Resizing the Palette

- 1 Click the palette's thick border that faces the Modeler space.



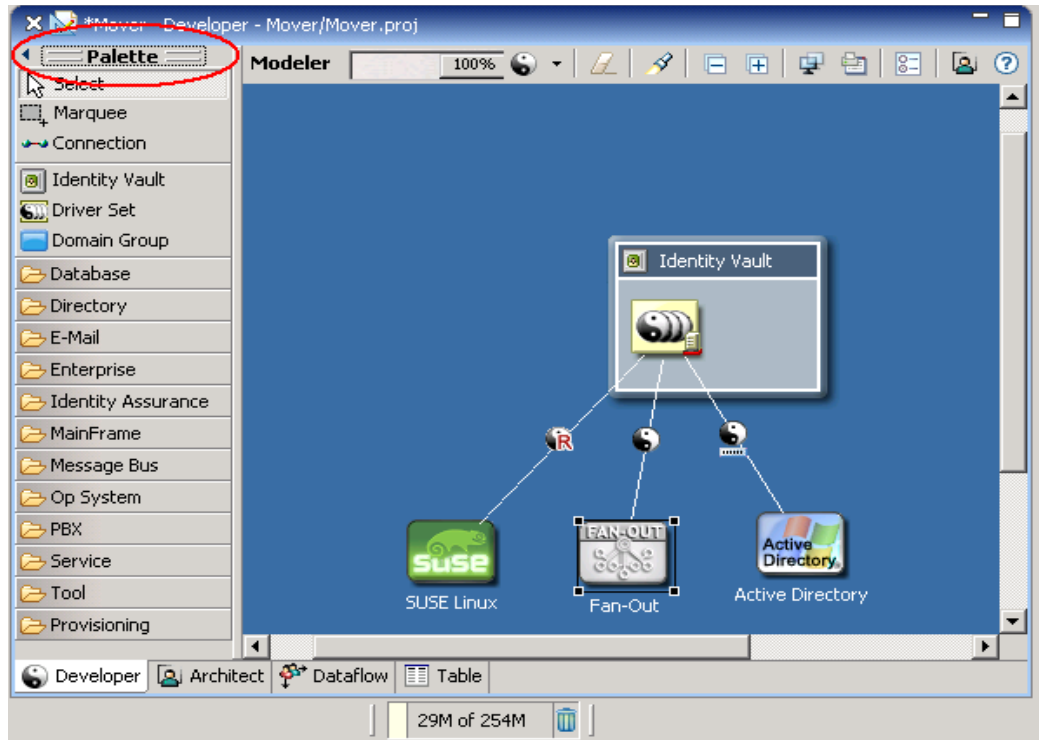
- 2 Drag the line.

The size persists and is restored the next time you run the application.

7.4.6 Docking the Palette

To dock the palette on the left or right of the Modeler space:

- 1 Click the top palette header.



- 2 Drag the palette to the desired location.

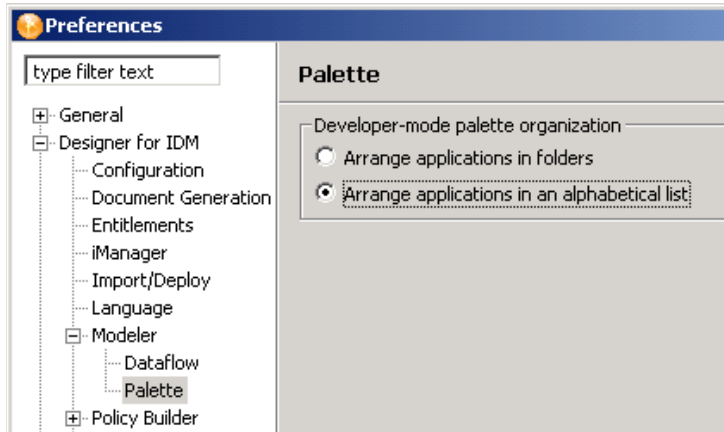
The location persists and is restored the next time you run the application.

7.4.7 Arranging Folders and Applications

By default, applications are placed in folders.

To arrange applications alphabetically instead of in folders:

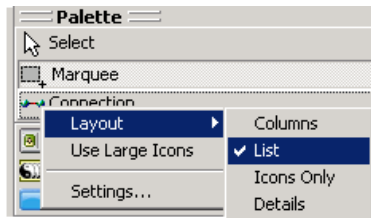
- 1 Click *Window > Preferences > Designer for IDM > Modeler > Palette*.



- 2 Select *Arrange applications in alphabetical list*, then click *OK*.

7.4.8 Changing the Layout

- 1 Right-click the palette.



- 2 Select *Layout*.
- 3 Select an option.

Setting	Description
<i>Layout: Columns</i>	Displays folders and applications in columns.
<i>Layout: List</i>	Arranges folders and applications in a list.
<i>Layout: Icons Only</i>	Removes descriptive labels.
<i>Layout: Details</i>	Briefly describes palette items.
<i>Use Large Icons</i>	Toggles the size of icons used for applications.
<i>Settings</i>	Enables you set the layout and icon size in one dialog box. Controls how folders (drawers) behave.

7.4.9 Keyboard Support for the Palette

Table 7-2 *Shortcut Keys for the Palette*

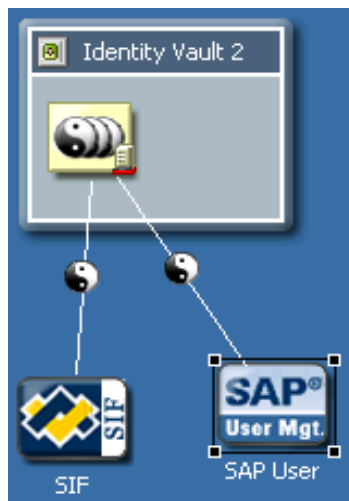
Keystroke	Description
Left-arrow	Collapses an open folder. The focus must be on the folder, not the application.
Right-arrow	Opens a collapsed folder. Moves into an open folder.
Up-arrow	Moves up to the next folder.
Down-arrow	Moves down to the next folder.

7.5 Copying and Pasting

- ◆ [Section 7.5.1, “Copying Applications,”](#) on page 68
- ◆ [Section 7.5.2, “Copying a Driver Set,”](#) on page 69
- ◆ [Section 7.5.3, “Copying an Identity Vault,”](#) on page 70
- ◆ [Section 7.5.4, “Copying a Domain Group,”](#) on page 71
- ◆ [Section 7.5.5, “Copying between Editors,”](#) on page 71

7.5.1 Copying Applications

Figure 7-13 *Applications to Copy*



You can copy and paste the following within the same editor or to another editor:

- ◆ Applications, including custom applications
- ◆ Disconnected applications
- ◆ Driver icons

1 Select an application or driver icon.

- 2 Press Ctrl+C, then Ctrl+V.

The copy and paste operations are also accessible from the Clipboard context menu. (Currently, they aren't accessible from the main menus.)

When you copy an application in the same editor, Designer copies all of the application's attributes, and copies all subelements. Therefore, all drivers that the application is connected to are copied, and all policies that the drivers contain are also copied. The new application defaults to connect to the same driver sets that the previous application connected to.

To copy an application to a different driver set (in the same editor or in another editor):

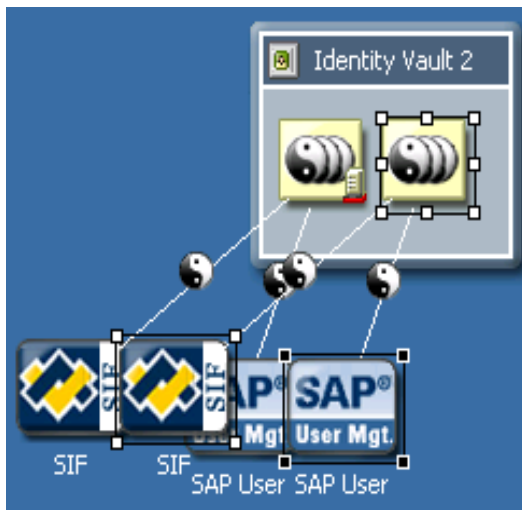
- 1 Select the application.
- 2 Press Ctrl+C.
- 3 Select the target driver set that the application will connect to.
- 4 Press Ctrl+V.

If you copy and paste an application without selecting a target driver set, Designer makes a copy and connects it to the current driver set.

You can select multiple applications and then copy and paste them.

7.5.2 Copying a Driver Set

Figure 7-14 Driver Sets



You can copy and paste driver sets within the same Identity Vault or to another Identity Vault in the same editor or in another editor.

- 1 Select a driver set.
- 2 Press Ctrl+C, then Ctrl+V.

When you copy a driver set in the same editor, Designer copies all of the attributes of the driver set, including the following:

- ♦ All drivers that the driver set is connected to

- ◆ All policies that the drivers contain
- ◆ All target applications

To copy to a different editor:

- 1 Select a driver set.
- 2 Press Ctrl+C.
- 3 Select the target Identity Vault in the other Modeler editor where you want the driver set to be copied to.
- 4 Press Ctrl+V.

By default, the new driver set is created in the same Identity Vault as the one that it was copied from. However, if you select another Identity Vault, the driver set is copied there.

After you copy and paste, you might need to move the pasted objects to a better location so that they don't cover up an existing object. To do this, leave the objects selected after you paste them, then move them. Another way to easily select objects:

- 1 Right-click a driver set.
- 2 Select *Select All Connected Applications*.
- 3 Move one of the selected applications.

All connected applications move together.

When you copy a driver set, it has the same settings, except for the selected servers, which are blank. This exception occurs because the Identity Manager engine does not allow more than one driver set on an Identity Vault to be associated with the same server. Therefore, you need to set up the servers for the new driver set. If you copy an Identity Vault, Designer copies the driver sets. The new driver set has the same server settings set up for you.

You can select multiple driver sets and then copy and paste them. To copy and paste multi-driver connections, you must copy the driver set or Identity Vault that contains them. In Designer 2.1, if you copy the application that has a multi-driver connection, the application and only one of its drivers are copied.

7.5.3 Copying an Identity Vault

You can copy and paste Identity Vaults within the same editor, to another editor in the same Modeler space, or in a specific Domain Group.

- 1 Select an Identity Vault.
- 2 Press Ctrl+C.
- 3 Select nothing or the target Domain Group (in the same editor or another) where you want the Identity Vault to be copied to.

If you select nothing, the new Identity Vault is copied to the right of the previous Identity Vault in the current editor.

- 4 Press Ctrl+V.

The new Identity Vault appears to the right of the previous Identity Vault and is the same size as the one that is it being copied from.

When you copy an Identity Vault, Designer copies all of the elements of the Identity Vault. The elements include servers, e-mail templates, driver sets, and connected applications.

You can select multiple Identity Vaults and then copy and paste them.

7.5.4 Copying a Domain Group

You can copy and paste Domain Groups within the same editor or to another editor in the same Modeler space or in a specific Domain Group.

- 1 Select a Domain Group.
- 2 Press Ctrl+C.
- 3 Select the location for the new Domain Group.

If you select nothing, the new Domain Group is copied to the right of the previous Domain Group in the current editor.

- 4 Press Ctrl+V.

The new Domain Group appears to the right of the previous Domain Group, and is the same size as the one it was copied from.

When you copy a Domain Group in the same editor, Designer copies all of the attributes of the Domain Group. However, Designer doesn't copy all subelements.

You can select multiple Domain Groups and then copy and paste them.

7.5.5 Copying between Editors

To easily copy and paste between two editors:

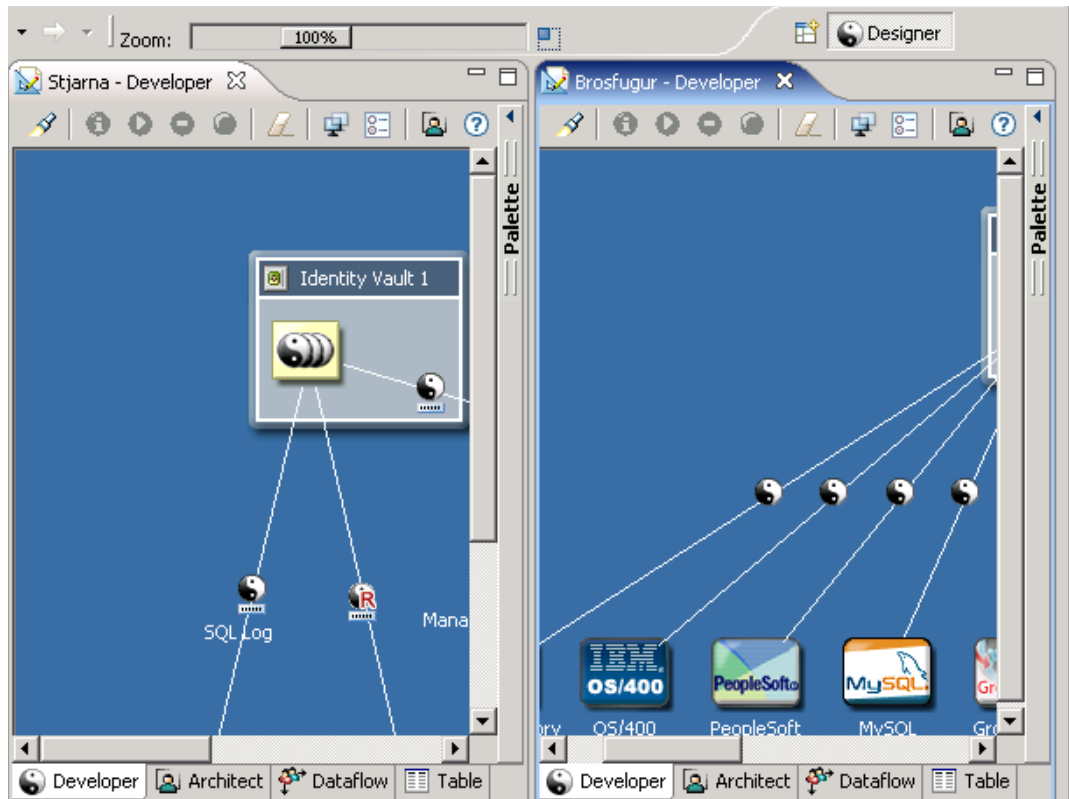
- 1 Using the Project view, open two projects.
One project is active. The second project's tab displays at the top of the Modeler.
- 2 Close the palette by clicking the control arrow on the palette's title bar.



- 3 Click the second project's tab and drag it to the Modeler's right border.

The tab changes to a folder icon until it arrives near the border, where the folder changes to an arrow.

4 Release the mouse.



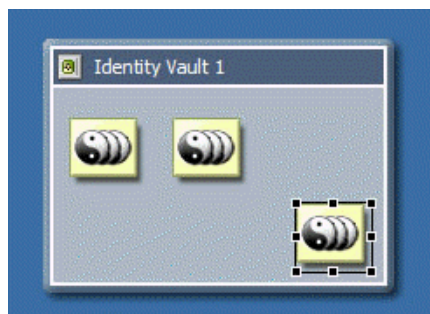
5 Copy items from one editor to the other.

7.6 Moving Items

After an item is in the Modeler space, you can move it by dragging it to a new location. The Modeler prevents you from placing objects where they don't belong. For example, you cannot move a driver set out of an Identity Vault to the Modeling space, or drop an application inside of an Identity Vault. You can always drag objects into a Domain Group, or drag a driver set from one vault into another.

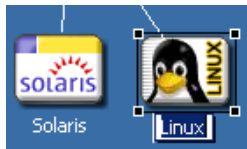
If you drag a driver set into an Identity Vault, the Identity Vault automatically grows or shrinks to fit the driver set, so you don't need to manually resize the vault. This behavior can be turned on or off in Preferences. See [Section 23.3.8, "Modeler,"](#) on page 461.

Figure 7-15 Example Driver Sets in an Expanded Identity Vault



7.7 In-Line Editing

Figure 7-16 An In-Line Edit



To edit the names of objects, do one of the following:

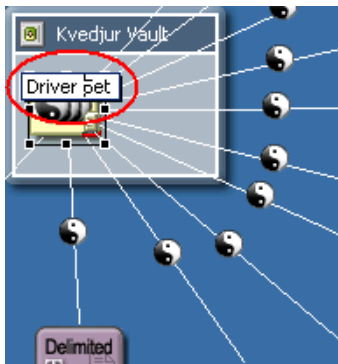
- ◆ Select the item, press F2, then edit the label.
- ◆ Double-click the item, then edit the *Name* field.

You can do an in-line edit for any type of item in the Modeler, including the driver lines.

7.8 Tooltips and Toolbar

As you move the mouse over objects in the Modeler, a tooltip appears with the name of the object.

Figure 7-17 A Tooltip



The Modeler also provides a tool bar.

Figure 7-18 The Modeler Toolbar



The Modeler toolbar enables you to quickly find often-used features:

- ◆ Search
- ◆ Find a driver's status (also available from the *Live* menu when you select a driver set or Identity Vault)
- ◆ Start, stop, or restart a driver (also available from the *Live* menu when you select a driver set or Identity Vault)
- ◆ Clear all items
- ◆ Save a snapshot of the model
- ◆ Launch Modeler preferences

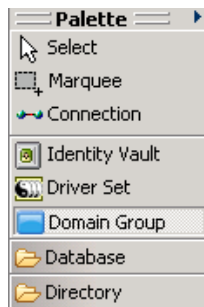
- View demos on how to use the Designer
- Get help

7.9 Organizing by Domain Groups

- [Section 7.9.1, “About Domain Groups,” on page 74](#)
- [Section 7.9.2, “Key Features,” on page 75](#)
- [Section 7.9.3, “Creating a Domain Group,” on page 75](#)
- [Section 7.9.4, “Minimizing \(Collapsing\) Domain Groups,” on page 77](#)
- [Section 7.9.5, “Restoring Domain Groups,” on page 77](#)
- [Section 7.9.6, “Maximizing Domain Groups,” on page 77](#)
- [Section 7.9.7, “Auto-Placement of Neighbors,” on page 78](#)
- [Section 7.9.8, “Grouping into a New Domain Group,” on page 78](#)
- [Section 7.9.9, “Ungrouping a Domain Group,” on page 79](#)
- [Section 7.9.10, “Clearing Contents,” on page 80](#)
- [Section 7.9.11, “Changing a Domain Group Icon,” on page 81](#)
- [Section 7.9.12, “Keyboard Support for Domain Groups,” on page 81](#)

7.9.1 About Domain Groups

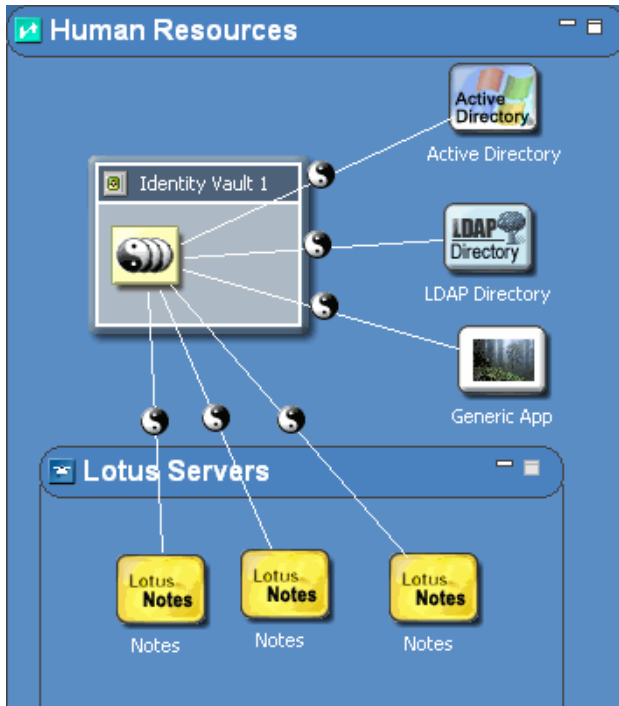
Figure 7-19 *The Domain Group Option on the Palette*



Domain Groups enable you to organize your model into logical groupings that help to keep your diagram clean. Domain Groups have no technical function, and they have no impact on how items and relationships are stored in the Identity Vault. This option is just a tool to help you better organize and view items in the Modeler.

Using Domain Groups is the key to modeling your entire enterprise, no matter how large it is. You can create a model that is manageable, useful, and logical, according to how you want to organize and diagram your enterprise.

Figure 7-20 A Domain Group in the Modeler



7.9.2 Key Features

- ◆ Change a group name through the Properties view.
- ◆ Drag and drop items in and out of groups.
- ◆ Minimize or restore groups.
- ◆ Move everything in a group.
- ◆ Remove everything in a group.
- ◆ Nest groups within groups (no limit).
- ◆ Resize groups. A minimum size is enforced.
- ◆ Ungroup. Remove the group but leave the children.

7.9.3 Creating a Domain Group

- 1 Drag and drop a Domain Group from the palette to the Modeler space.
- 2 Organize items inside Domain Group items.

To add another Domain Group, right-click inside the Domain Group, then select *Add Domain Group*.

To add an Identity Vault, do one of the following:

- ◆ Drag an Identity Vault from the palette.
- ◆ Right-click in the Domain Group, then select *Add Identity Vault*.

The Add Server to Identity Vault dialog box appears. If you select *Specify a Server*, Designer provides a dialog box that enables you to select an eDirectory™ server or specify a server manually.

To add a driver set:

- 1 Right-click inside an Identity Vault.
- 2 Select *Add Driver Set*.

To add an application:

- 1 Right-click a Driver Set object.
- 2 Select *Add Connected Application*.

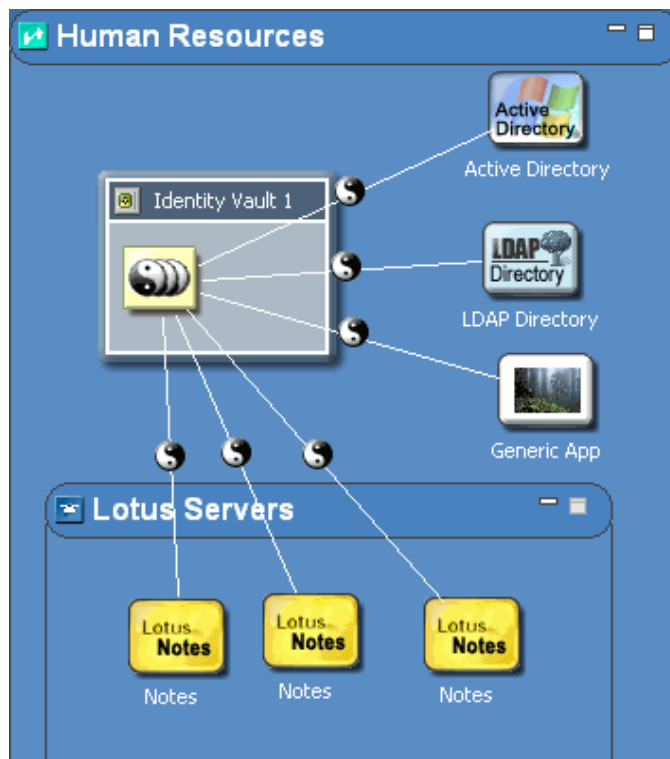
The application is added to the right of the right-most connected application. If this is the first application, it is placed under the driver set.

The application defaults to a generic application type. To change the type:

- 1 Right-click the application, then select *Properties*.
- 2 Select a different application, then click *OK*.

When you add selected items to a Domain Group, the Domain Group expands.


Figure 7-21 A Domain Group



If you move an item to the edge of the Domain Group, the boundaries expand, so that the items remain inside the Domain Group. You can drag an item from the Domain Group to remove it from the group.

You can have nested domains. If you expand a nested domain, the outer (hosting) domain auto-increases in size. You aren't required to manually resize parent domains. By expanding, the hosting domain displays the nested domain, so that the nested domain isn't cut off.

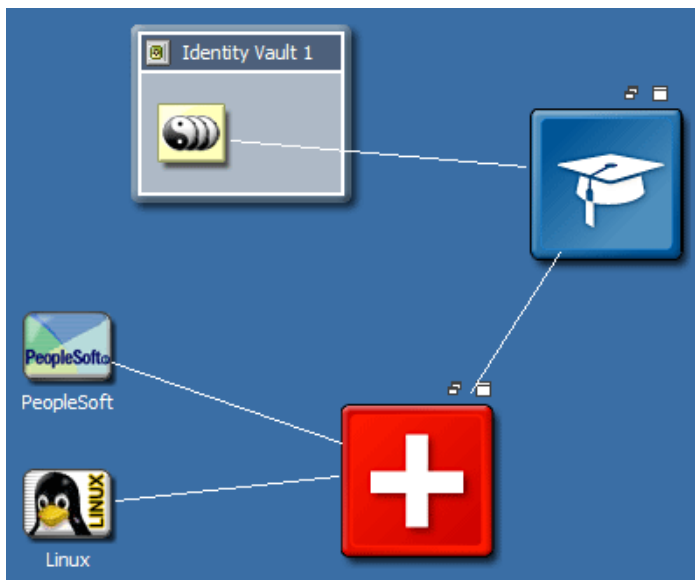
7.9.4 Minimizing (Collapsing) Domain Groups

To minimize a Domain Group, click the *Minimize*  button. When a Domain Group is minimized, it defaults to a random icon. (See “[Changing a Domain Group Icon](#)” on page 81.) The icon and minimized state of the group are saved in the Project file.

When a group is minimized, you can't see its contents, nor can you drag new items into the group. However, you can move, rename, or delete it.


When you minimize a group, lines that were connected to items in the group now connect to the group. This functionality enables you to see that there is a relationship with items in the group and items outside the group. Depending on your objects, their relationships, and state of other related groups, multiple lines might collapse into one line.

Figure 7-22 A Collapsed Group



When you expand the group, the lines are moved back to the actual items they connect with. This functionality works for any level of nesting of groups.

7.9.5 Restoring Domain Groups

To restore the Domain Group to its original size, click the *Restore*  button.

7.9.6 Maximizing Domain Groups

To maximize a Domain Group, click the *Maximize*  button. The group expands to a much larger size. To return it to the original size, click the *Restore* button.

You can maximize only first-level groups. For inner groups, the Maximize function is disabled.

7.9.7 Auto-Placement of Neighbors

To push or pull the neighboring items when you expand or contract Domain Groups, hold down the Ctrl key while you expand or contract the Domain Group. Any item that is to the right or below a Domain Group is affected.

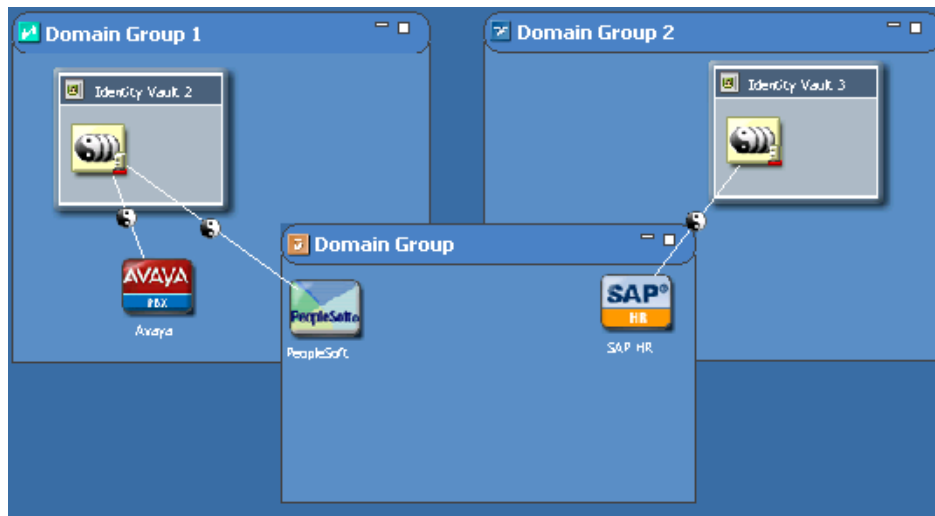
7.9.8 Grouping into a New Domain Group

- 1 In the Modeler, select multiple items.
- 2 Right-click, then select *Add to Group*.

The Modeler creates a new Domain Group and adds those items, preserving their relative spacing to each other. This process removes the items from wherever they previously existed and places them in the proper area in the new group.

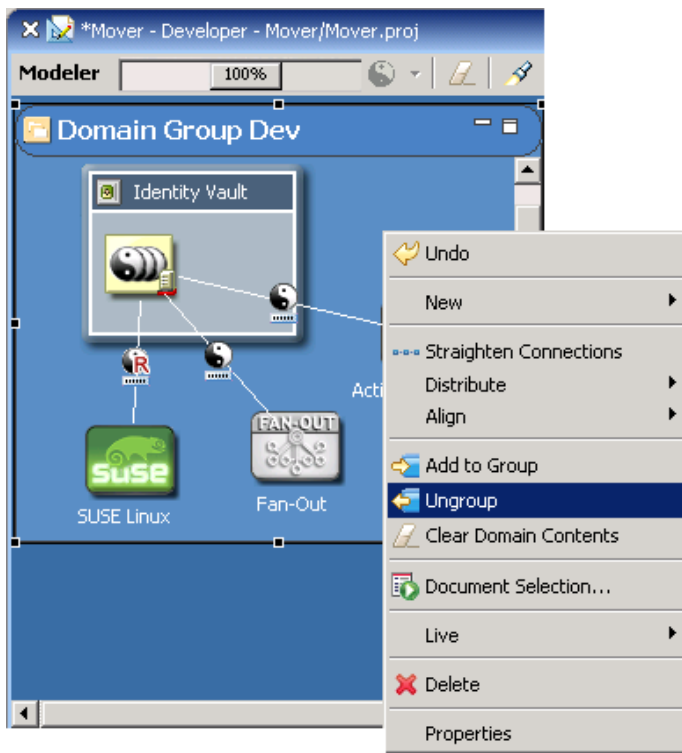
The following figure illustrates two Applications that have been added to a new Domain Group and removed from their previous groups.

Figure 7-23 Grouping into a New Domain Group



7.9.9 Ungrouping a Domain Group

Figure 7-24 Ungrouping a Domain Group

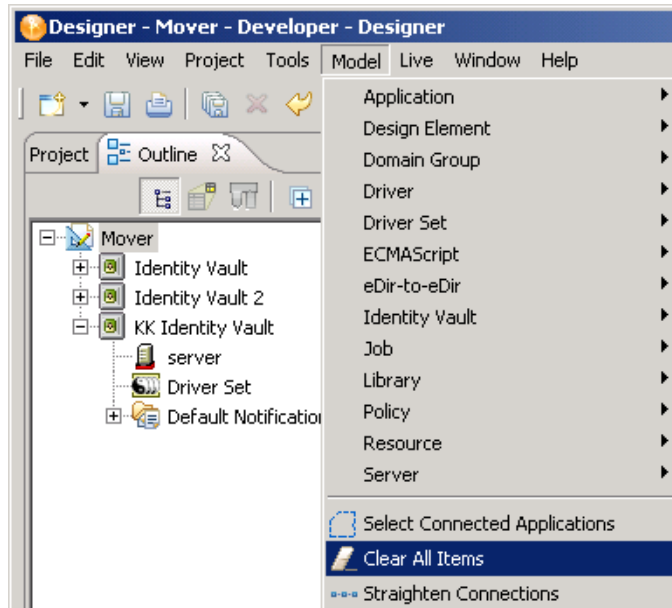


To ungroup a Domain Group, right-click it, then select *Ungroup*.

This process removes the Domain Group but leaves all contents where they are, so that they won't be deleted. This is just a way to ungroup the items. Depending on what level you are in the Modeler, the ungrouped items are automatically added to the parent group or to the main Modeling space.

7.9.10 Clearing Contents

Figure 7-25 *Clear All Items*



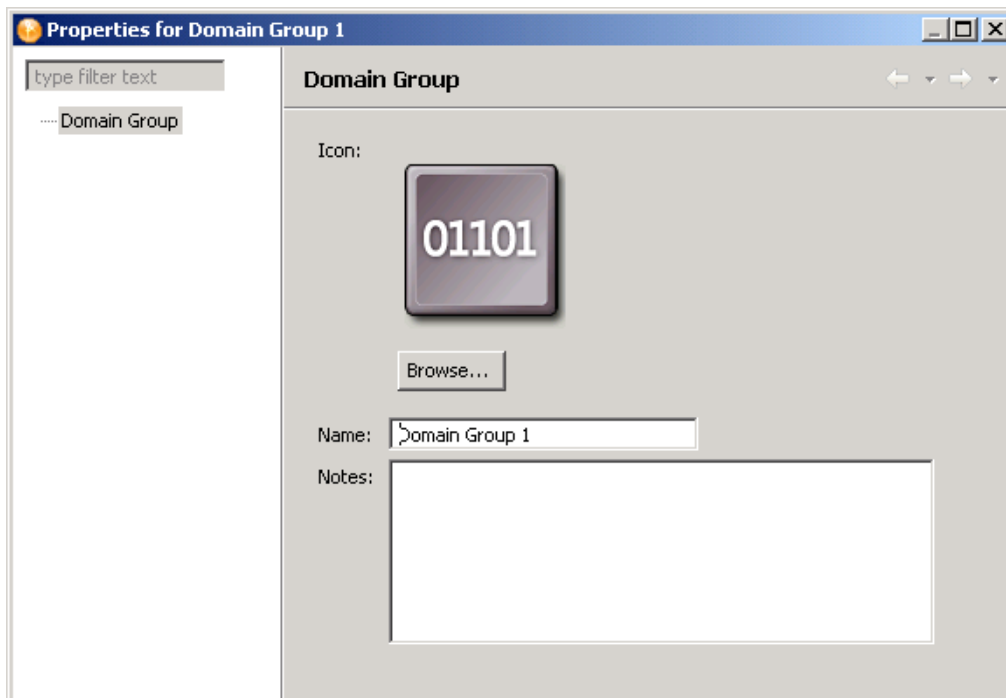
To remove all contents from the Modeler, click *Model*, then select *Clear All Items*.

To remove all contents from a Domain Group, right-click, then select *Clear Domain Contents*.

Designer prompts you before clearing the Modeler space.

7.9.11 Changing a Domain Group Icon

- 1 Right-click a Group Domain item in the Modeler, then select *Properties*.



- 2 Browse to and select an image (for example, `finance.png`).

Icons for Domain Group components reside in the `com.novell.designer.core\icons\Group` directory. By default, Designer opens the `Group` directory.

Designer supports `.GIF`, `.JPEG`, `.PNG`, and Windows `.BMP` formats. You can add your own icons to the `Group` directory.

- 3 Click *Open*, then click *OK*.

The minimized 16x16 version of the image also now appears in the Domain Group title bar.

As you add Domain Group items, Designer randomly assigns icons from the `Group` directory to the new Domain Group.

7.9.12 Keyboard Support for Domain Groups

Table 7-3 *Shortcut Keys for Domain Groups*

Keystroke	Description
Alt+Down-arrow	Navigates into a Domain Group
Alt+Up-arrow	Navigates out of a Domain Group
Delete	Deletes the selected items

7.10 Connecting Applications

- ◆ [Section 7.10.1, “Automatic Connections,” on page 82](#)
- ◆ [Section 7.10.2, “Connection Target Highlights,” on page 82](#)
- ◆ [Section 7.10.3, “Automatically Creating Objects,” on page 83](#)
- ◆ [Section 7.10.4, “Auto Redraw,” on page 83](#)
- ◆ [Section 7.10.5, “Manually Connecting,” on page 83](#)
- ◆ [Section 7.10.6, “eDir-to-eDir Connections,” on page 84](#)
- ◆ [Section 7.10.7, “Multiple Driver Connections,” on page 84](#)
- ◆ [Section 7.10.8, “Straightening Connections,” on page 86](#)
- ◆ [Section 7.10.9, “Reconnecting,” on page 87](#)
- ◆ [Section 7.10.10, “Driver Icons,” on page 87](#)
- ◆ [Section 7.10.11, “Selected Drivers,” on page 87](#)
- ◆ [Section 7.10.12, “Auto-Layout of Imported Objects,” on page 87](#)
- ◆ [Section 7.10.13, “Keyboard Support for Connections,” on page 88](#)

7.10.1 Automatic Connections

When you drag an application into the Modeler space, and the Modeler contains a driver set, Designer automatically draws a connecting line between the Driver Set object and the application.

When you use the palette’s Connection function to connect an application to an Identity Vault, you can begin or end your driver line at the Identity Vault. The line automatically connects to a driver set in an Identity Vault.

If the Identity Vault contains more than one driver set, the Connection function connects the driver line to the first driver set. This functionality also works for multi-driver connections.

All multi-driver driver lines are bendable. You can lay them out so that the lines don’t overlap at any angle. Also, you can reconnect multi-driver connections.

If an Identity Vault has multi-driver connections in a Domain Group and you minimize that Domain Group, a single collapsed line represents all of the multi-driver connections.

7.10.2 Connection Target Highlights

When you drag an application across the Modeler space, the closest Identity Vault and closest driver set in that Identity Vault are highlighted. The highlights indicate the item that the application will connect with when you drop the application.

Figure 7-26 Connected Objects



7.10.3 Automatically Creating Objects

If you drop an application into the Modeler space, and that space has no Identity Vaults, Designer automatically creates an Identity Vault.

7.10.4 Auto Redraw

If you move items, lines are automatically redrawn.

7.10.5 Manually Connecting

To manually connect an application to a driver set:

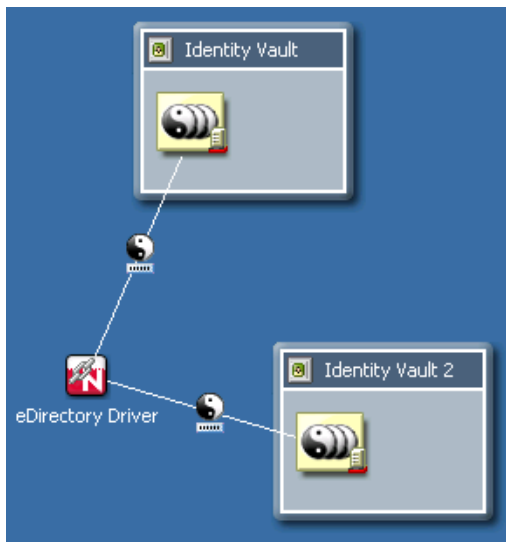
- 1 Click *Connection* in the palette.
- 2 Draw a line between the application and the driver set.

To reconnect an application, select the driver line, then drag one end of the line to another driver set or application.

The drag gesture gravitates the line towards the nearest connectable point. This functionality helps you know what you can connect to and where you can connect the item. If you try to connect to something that isn't allowed, the cursor usually indicates so, or nothing happens when you drop the item.

7.10.6 eDir-to-eDir Connections

Figure 7-27 eDir-to-eDir Connections



An eDir-to-eDir connection is a special type of connection. It is used frequently in Identity Manager environments. This connection is a way to configure two eDirectory drivers to communicate directly with each other. (No other drivers are able to communicate directly with any other type of driver.) This type of connection is most commonly used for synchronizing a local directory tree with a Metadirectory Identity Vault.

To create an eDir-to-eDir connection, do one of the following:

- ◆ Drag a line between two Identity Vaults
- ◆ Drag a line between two driver sets

When you connect a line between two eDirectory applications, the line automatically turns into an eDir-to-eDir connection. See the illustration in [Section 12.2.6, “Viewing an eDir-to-eDir Driver,” on page 234](#).

To disconnect an eDir-to-eDir connection, right-click an eDir item, then select *Disconnect eDir-to-eDir*. Designer creates two new eDirectory applications and redirects each driver to its respective application. A new driver is not created. No data is lost. Designer keeps the same drivers.

If you delete one side of an eDir-to-eDir connection, Designer converts the remaining half into a regular driver connection to an eDirectory application.

7.10.7 Multiple Driver Connections

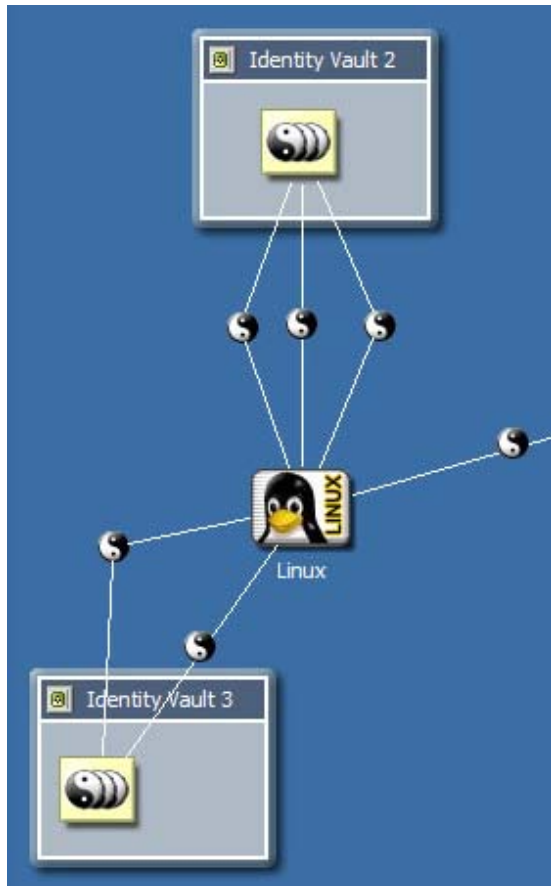
To connect more than one driver from a driver set to an application:

- 1 Select *Connection* in the palette.
- 2 Connect the driver set and the application again and again.

Each time you connect, a new line is added. All lines are bendable, so that the lines don't overlap. To get the model to look optimal, you probably need to move the application slightly from its default position.

You can also connect more than one driver to a single application. This actually causes the application to act as a hub. Each driver can connect to and authenticate to the application or system the same or differently, depending on your needs. Each driver can access the same part of the application or system or different parts (for example, different tables in a database). The Modeler lets you diagram a layout according to your needs.

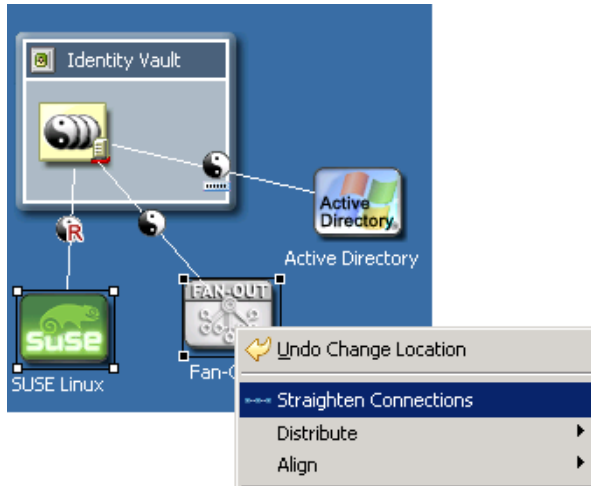
Figure 7-28 Multi-Driver Connections



7.10.8 Straightening Connections

To straighten connecting lines:

- 1 Press Ctrl, then select one or more applications in the Modeler.



- 2 Right-click, then select *Straighten Connections*.

What is straightened depends on what you select:

Table 7-4 *Straightened Connections*

Selected Item	What Is Straightened
A driver	That driver's line
An application	The connecting driver's line
An Identity Vault	All lines that originate from that driver set in that Identity Vault
A Domain Group	Everything in the Domain Group
A project (selected by clicking the Modeler's background)	Everything in that project

Lines are straightened only if they are less than 20 pixels from a north, west, south, or east alignment. The intent of this operation is to quickly nudge lines that are almost straight, so that they become perfectly straight.

This nudging removes the tedium of meticulously dragging items into perfect alignment and being concerned with the pixels. If a line isn't almost straight, it is left alone. In fact, the Straighten Connection operation is disabled unless the selected items qualify to be straightened. If some of the selected items qualify but others don't, the operation is still enabled, but only eligible lines are straightened.




7.10.9 Reconnecting

To reconnect components, do one of the following:

- ◆ Drag the end of a line (driver) from one application to another.
- ◆ Drag the end of a line (driver) from one driver set to another.

7.10.10 Driver Icons

Table 7-5 *Driver Icons*

Icon	Description
	A driver. The entire line between a driver set object and an application represents a driver.
	A remote driver.
	A firewall. Indicates that the driver is communicating across a firewall.

The following figure illustrates these icons.

Figure 7-29 *Driver Icons*



To see, turn on, or turn off driver icons:

- 1 Right-click a driver line.
- 2 Select an option (for example, *Mark as Firewall*) to turn on or turn off.

7.10.11 Selected Drivers

As you move the mouse over a driver, the line thickens so that it is more obvious. You can click and interact with this line.

7.10.12 Auto-Layout of Imported Objects

When you import objects from the directory, they are automatically laid out, connected with lines, and assigned an icon that matches objects and relationships as closely as possible.

For example, if you import a Driver Set object, Designer imports all of the drivers and connects them with lines. Also, each driver points to an application icon. Application icons include the following:

- ◆ The exact Application icon (for example, Avaya* or PeopleSoft*)
- ◆ The image stored on the driver
The image is embedded in a square application icon.
- ◆ A generic application icon

If no image is stored on the driver, Designer supplies an icon for one of the following applications:

- ◆ Generic
- ◆ JDBC*
- ◆ LDAP
- ◆ Delimited Text

The auto-layout mechanism uses the layout topology that you have selected. The default is Fan Out - Bottom. You can customize this setting in Preferences. See [Section 23.3.8, “Modeler,” on page 461](#).

7.10.13 Keyboard Support for Connections

Table 7-6 Shortcut Keys for Connections

Keystroke	Description
/	Navigates to the item's next connection
\	Navigates to the item's previous connection

7.11 Aligning and Laying Out Components

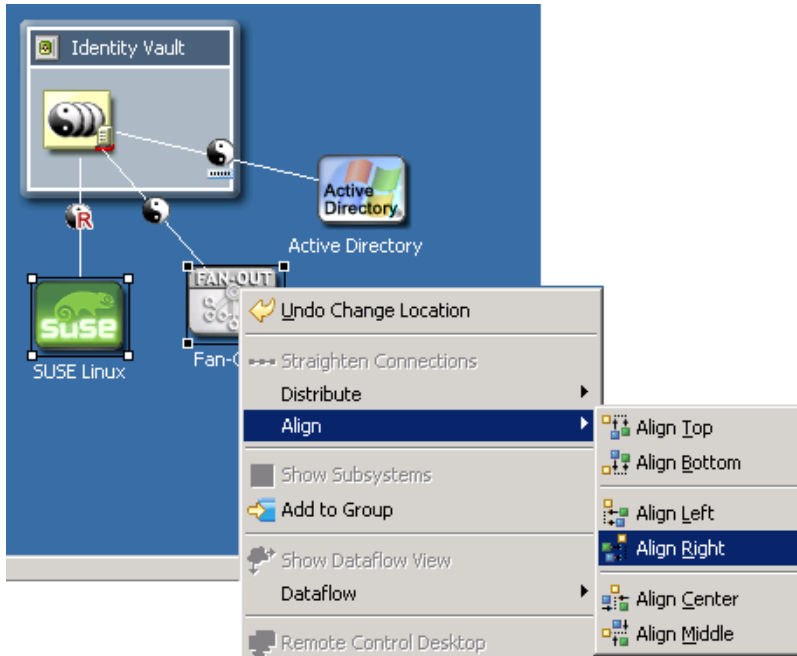
- ◆ [Section 7.11.1, “Alignment Hints,” on page 89](#)
- ◆ [Section 7.11.2, “Using Rulers,” on page 90](#)
- ◆ [Section 7.11.3, “Using a Grid,” on page 92](#)
- ◆ [Section 7.11.4, “Distributing Applications,” on page 93](#)
- ◆ [Section 7.11.5, “Auto-Layouts,” on page 94](#)
- ◆ [Section 7.11.6, “Layouts to Use for Imports,” on page 94](#)

Alignments place objects in the same horizontal or vertical plane. Alignments help you see relationships in your model. You can align or attach items to the left, center, or right of alignment guides.

When you move the guide, attached items move with it, staying attached in the same relative positions.

To align components:

- 1 Press Ctrl, then select more than one item.



- 2 Right-click, then select *Align*.

- 3 Select an alignment option.

You can also attach an item by dragging it to a guide. After you wait a moment, the guide line is highlighted, indicating that the item is attached. You can align within the same group but not across groups.

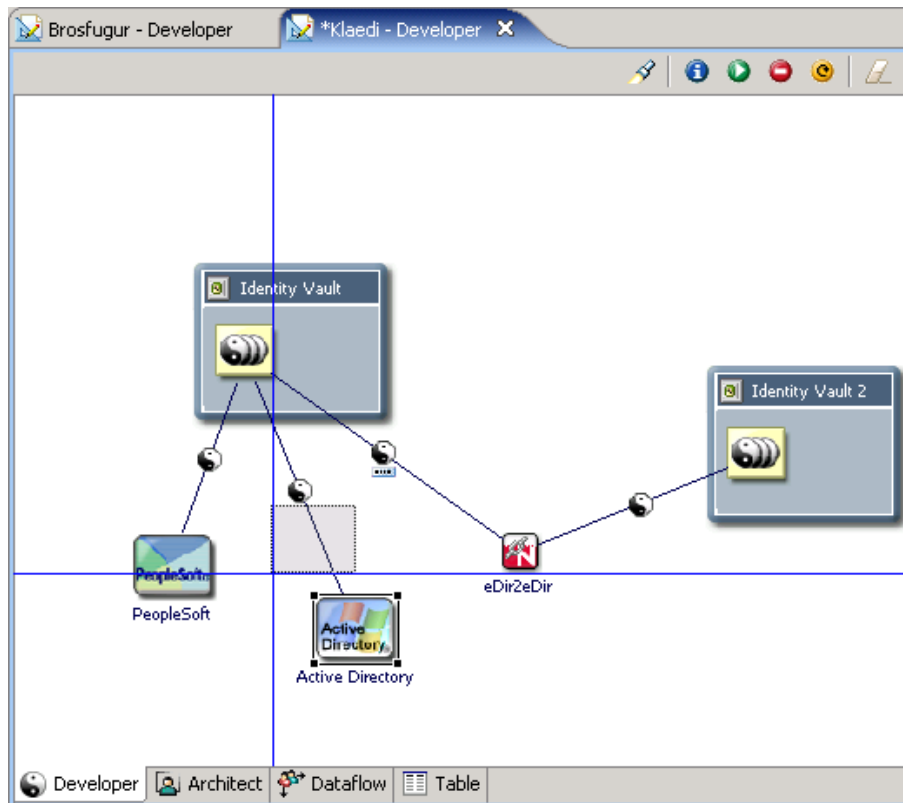
Guides that you set up are restored the next time that you run Designer. You don't need to re-create them.

Also, the alignments and attachments (left, center, or right) are stored in the project on a per-item basis, so that they are also restored.

7.11.1 Alignment Hints

Horizontal and vertical “hint” lines automatically show as you drag items into vertical or horizontal alignment with neighboring items.

Figure 7-30 Alignment Hints



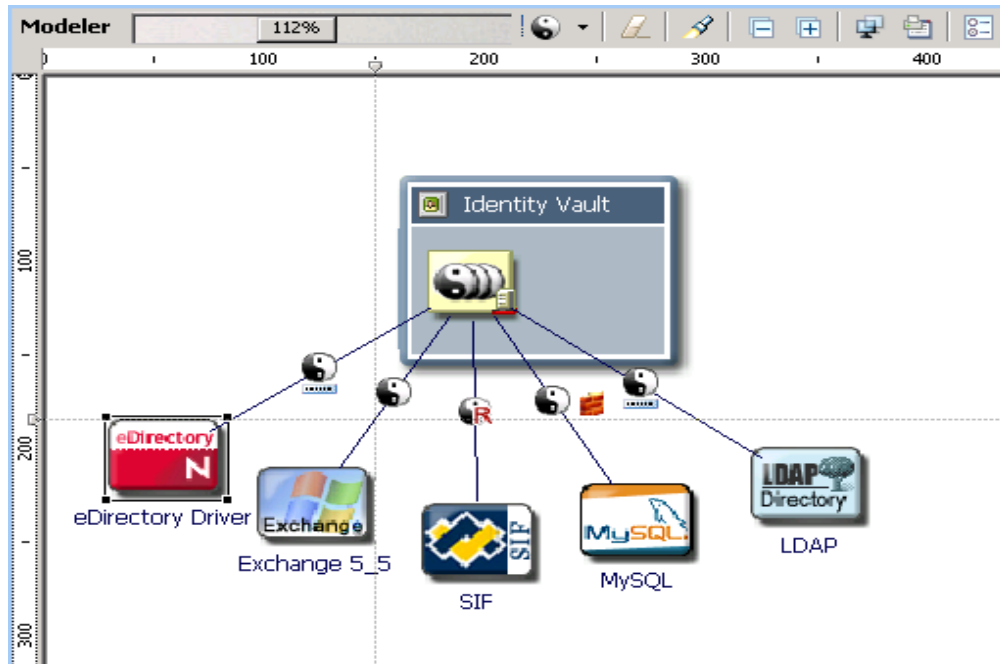
The Alignment Hints feature is off by default. To turn it on, click *View > Alignment Hints*.

Horizontal and vertical hint lines automatically show as you drag items into vertical or horizontal alignment with neighboring items.

7.11.2 Using Rulers

To turn on the horizontal and vertical rulers:

- 1 Click the Modeler space to make it active.
- 2 Click *View > Rulers*.



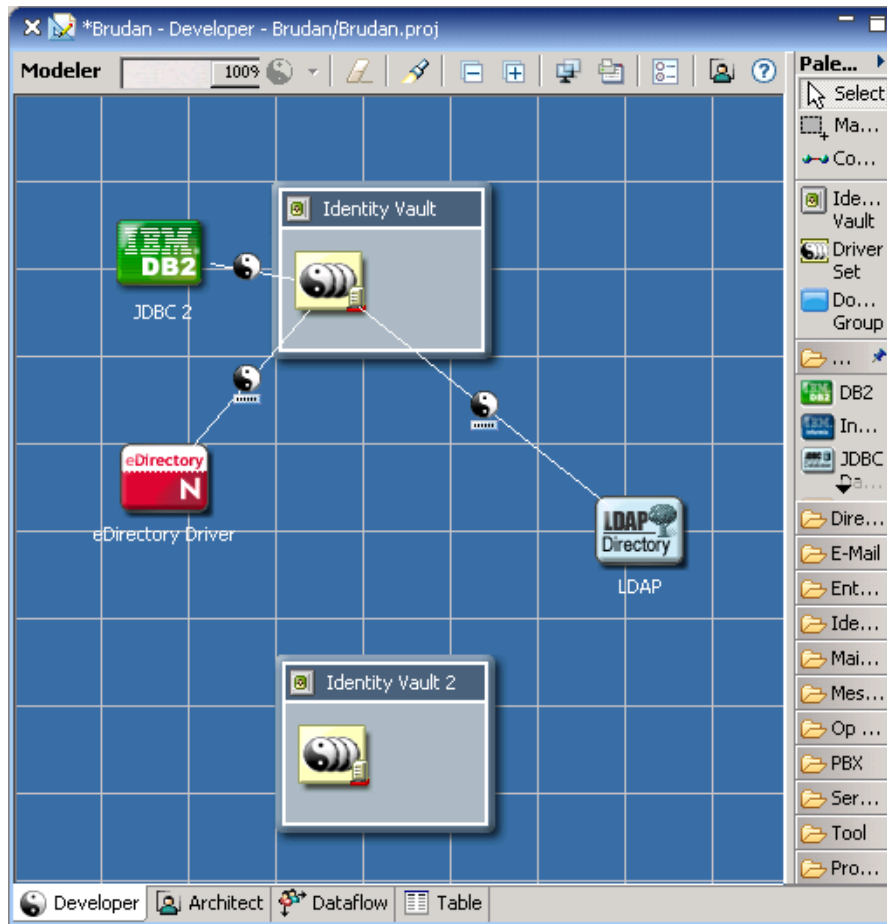
To create a guide (line), click either ruler. If the Modeler background is dark blue, you might want to change the color so that the guides are easier to see.

To anchor items to a guide, drag the items in the model to the line.

To simultaneously move all anchored items, drag the line.

7.11.3 Using a Grid

Figure 7-31 The Modeler's Grid



When the grid is on, the snap-to-grid functionality is on.

To turn grid lines on and off:

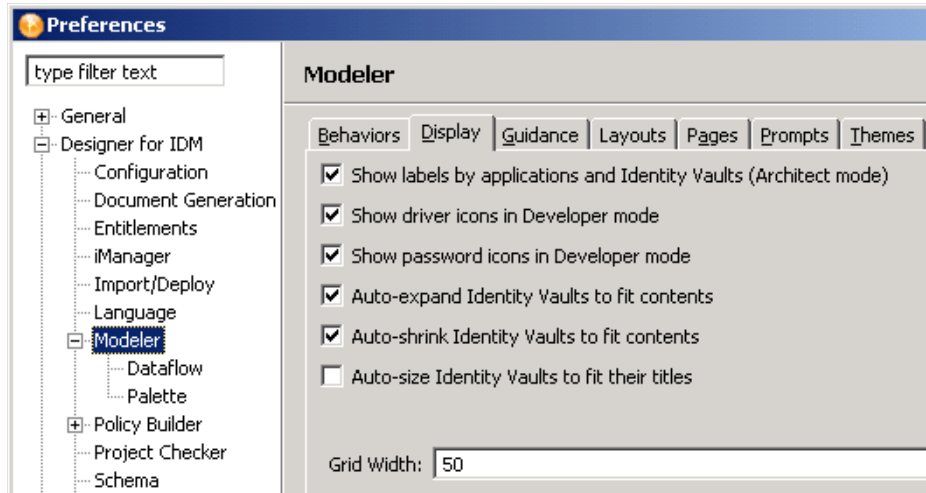
- 1 Click the Modeler, so that the Modeler is the active view.
- 2 Click *View > Grid*.

To coerce objects to not align with the grid, temporarily turn off snap-to-grid by holding down the Alt key. (Linux doesn't support this functionality.)

To constrain items to north-south or east-west coordinates, press Shift while dragging the items.

To change the grid size:

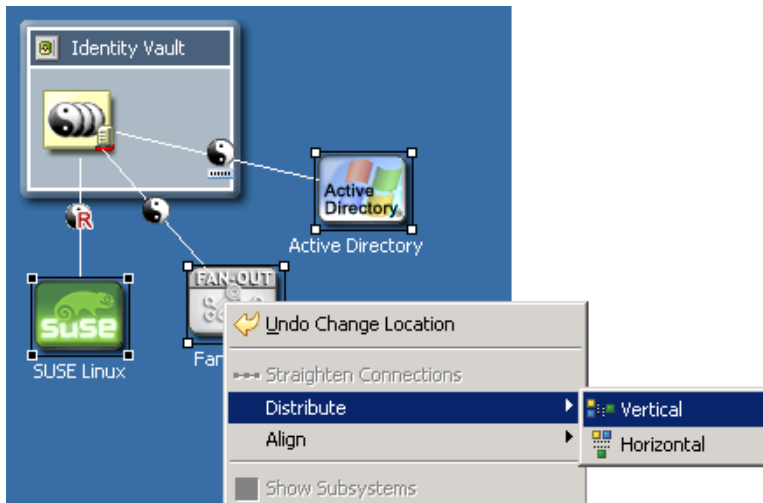
- 1 Click *Window > Preferences > Designer for IDM > Modeler > Display*.



- 2 Type a value in the *Grid Width* field.

7.11.4 Distributing Applications

Figure 7-32 *Distributing Applications*



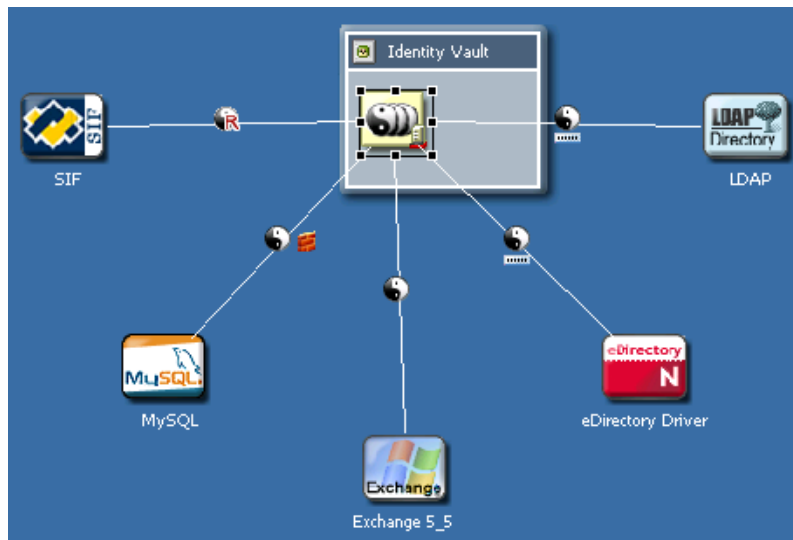
To equally distribute (space) applications horizontally or vertically:

- 1 Press *Ctrl*, then select three or more items.
- 2 Right-click, then select *Distribute*.
- 3 Select a distribution (for example, *Vertical*).

7.11.5 Auto-Layouts

Designer ships with a number of predefined layout topologies: circle, half-circle, star, box, and different fan-out layouts.

Figure 7-33 A Circle Layout



These layouts are set on a per-driver-set basis. Therefore, each driver set can have its own layout.

To select a layout:

- 1 Right-click a driver set, then select *Arrange Applications*.
- 2 Select an arrangement (for example, *Fan Out - Left*).

If your model has an incorrect layout, the layout options are dimmed.

After you set a layout, applications that you connect will automatically snap into that layout. Certain connected objects (for example, multi-driver connections, eDir-to-eDir connections, and applications that are connected but reside in a different Domain Group) are ignored. They aren't included in the layout, and they don't disturb it.

An option on the Arrange Applications submenu on the Modeler's context menu enables you to expand or contract the layout arrangement. This option makes all spokes of the layout longer or shorter when you drag a slider.

7.11.6 Layouts to Use for Imports

To specify what layout to use on new driver sets that you import:

- 1 Select *Window > Preferences > Designer for IDM*.
- 2 Click *Modeler > Layouts*.
- 3 Select an arrangement (for example, *Half Circle*), then click *OK*.

7.12 Editing Multiple Objects

You can open multiple objects and edit them at the same time. These objects must be of the same type (for example, policies).

To find out whether you can edit an object, right-click it. If *Edit* displays among the menu items, you can edit that object.

- 1 In the Outline view, expand the project that contains the objects that you want to edit.
- 2 Select the objects.
- 3 Right-click, then select *Edit*.
- 4 Edit the objects.

You can copy and paste from one editor to another. Data must be of the same type.

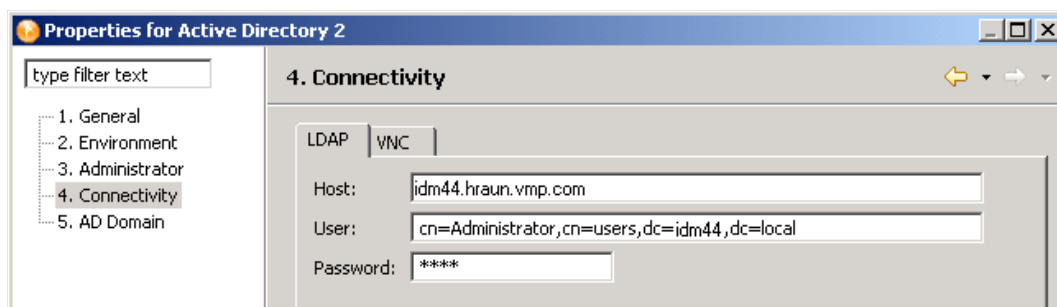
7.13 Modeling Active Directory Domain Controllers

- [Section 7.13.1, “Configuring a Connection,” on page 95](#)
- [Section 7.13.2, “Discovering Controllers,” on page 95](#)
- [Section 7.13.3, “Information about Domain Controllers,” on page 96](#)

7.13.1 Configuring a Connection

You can configure an LDAP connection to an Active Directory* system so that you can discover its Domain Controllers.

- 1 Right-click the AD application, then select *Properties > Connectivity*.



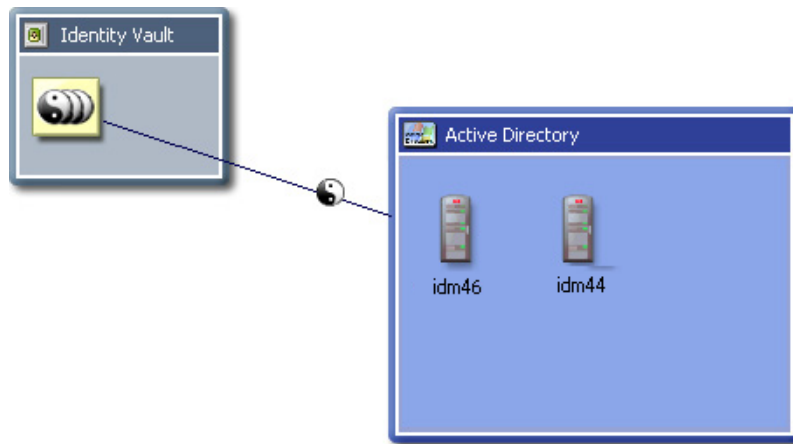
- 2 Complete the LDAP authentication information.

As you tab from the *Host* field to the *User* field, Designer automatically builds a full user context. You can modify this context.

7.13.2 Discovering Controllers

- 1 Right-click the Active Directory application.

- 2 Select *Discover Domain Controllers*.

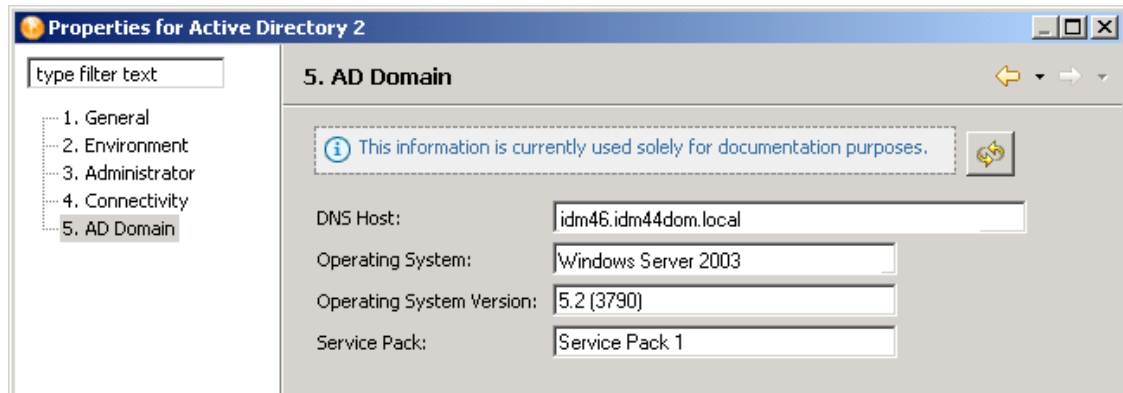


If Designer finds any controllers, it lays them out and expands the Active Directory application as a container.

7.13.3 Information about Domain Controllers

Information about each controller is loaded into the Modeler. To view this information, edit the Domain Controller object and select the AD Domain page.

Figure 7-34 The AD Domain Page



If the LDAP connection information is filled out, you can reread the information from that system by clicking the *Refresh* button on the AD Domain page.

7.14 Saving Your Model

To save your model, do one of the following:

- From the main menu, select *File > Save* (or *Save All*).
- From the main menu, select *File > Close > Yes*.
- Click the *X* in the Modeler's tab, then select *Yes*.

For more information, see “Saving” in [Section 8.4.2, “Tasks You Can Perform from the Project View,”](#) on page 114.

Workspaces, Perspectives, and Views

8

This section assumes that you have created a project. See [Chapter 6, “Creating a Project,”](#) on page 49.

Designer has many views to help you manage your projects. Many of these views help you manage the different elements that appear in the Modeler view and give you a different perspective on the projects you are creating. For instance, the Project view shows all of the projects you have presently created, while the Outline view shows all of the objects that have been created and added to the project. The Navigator view shows all of the files that have been created for the project and their respective file types and locations.

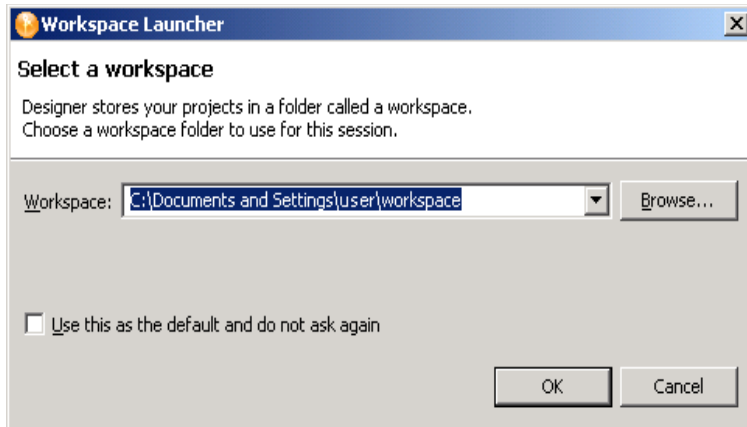
For information on the Modeler view and its functions, see [Chapter 7, “Building a Model,”](#) on page 53. For information on policies, see [Understanding Policies for Identity Manager 3.5.1](#) and [Policies in Designer 2.1](#).

- ◆ [Section 8.1, “Specifying the Workspace,”](#) on page 99
- ◆ [Section 8.2, “Managing Perspectives,”](#) on page 105
- ◆ [Section 8.3, “Managing Views,”](#) on page 107
- ◆ [Section 8.4, “The Project View,”](#) on page 113
- ◆ [Section 8.5, “The Outline View,”](#) on page 121
- ◆ [Section 8.6, “The Navigator View,”](#) on page 135
- ◆ [Section 8.7, “The Policy Set View,”](#) on page 138
- ◆ [Section 8.8, “The Properties View,”](#) on page 139
- ◆ [Section 8.9, “The Search Results View,”](#) on page 142
- ◆ [Section 8.10, “The Tasks View,”](#) on page 143
- ◆ [Section 8.11, “The Trace and Error Log Views,”](#) on page 144
- ◆ [Section 8.12, “The Dataflow View,”](#) on page 144

8.1 Specifying the Workspace

A workspace is a directory on your local hard drive where your projects and settings are stored. You specify your workspace directory each time you launch Designer, unless you select the *Use this as the default and do not ask again* option to skip the Workspace Launcher window.

Figure 8-1 Selecting a Workspace Directory



A typical default workspace directory for a Windows* workstation is `Documents and Settings\username\workspace`.

You can switch to another workspace by selecting *File > Switch Workspace*. Designer quickly restarts, loads the new workspace, and loads any projects and settings you have in that workspace. If the workspace doesn't exist, Designer creates it.

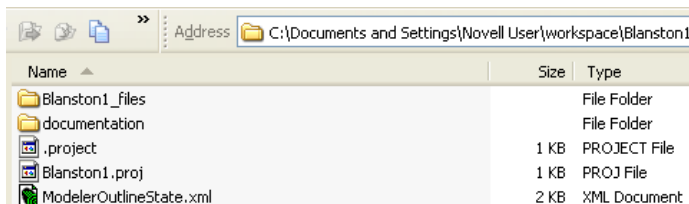
You can have multiple workspaces.

- ◆ [Section 8.1.1, “Project File Format,” on page 100](#)
- ◆ [Section 8.1.2, “Editing by Using a Related Editor,” on page 101](#)

8.1.1 Project File Format

All projects are stored on the file system in a directory. By default on Windows workstations, this is in the `C:\Documents and Settings\Username\workspace` directory on a Windows workstation. However, you can specify a different directory when you create the project. (See [Chapter 6, “Creating a Project,” on page 49.](#)) All of your data and graphical layout information is saved in a directory corresponding to the project name. For example, the following files are stored under the `\workspace\Blanston3` directory:

Figure 8-2 Local Directory Structure



The main file in the project's directory is the `.proj` file. This file uses an industry-standard XML format called XMI (XML Metadata Interchange Format). You never need to directly edit this file.

What is XMI?

XMI enables easy interchange of metadata between modeling tools and between tools and metadata repositories in distributed heterogeneous environments. XMI integrates three key industry standards:

- ◆ XML (Extensible Markup Language, a W3C standard)
- ◆ UML (Unified Modeling Language, an OMG modeling standard)
- ◆ MOF (Meta Object Facility, an OMG modeling and metadata repository standard)

The integration of these three standards into XMI combines the best of OMG and W3C metadata and modeling technologies. The integration enables developers of distributed systems to share object models and other metadata. XMI, together with MOF and UML, forms the core of the OMG repository architecture that integrates object-oriented modeling and design tools with each other and with a MOF-based extensible repository framework.

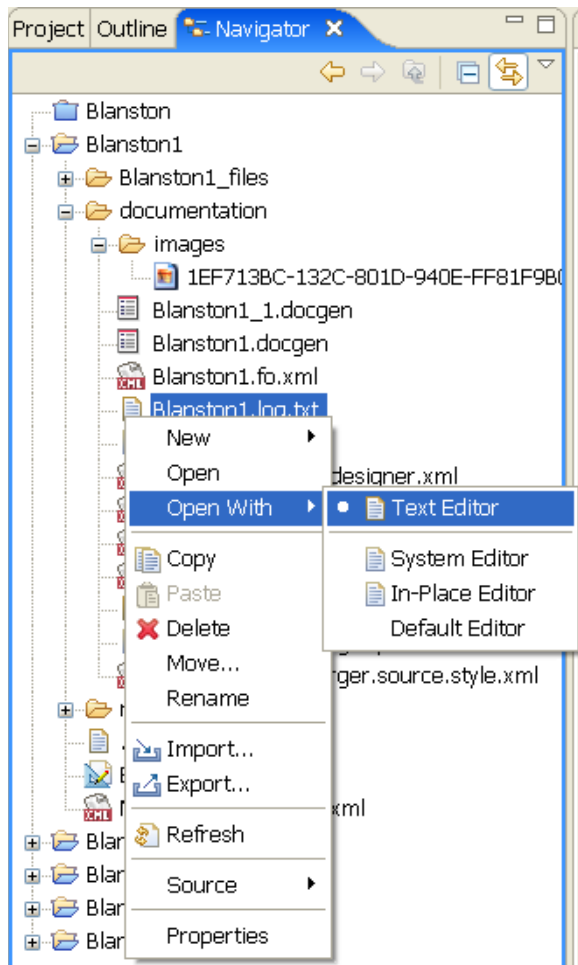
For more information about XMI, go to [XML Metadata Interchange \(XMI\) \(http://www.oasis-open.org/cover/xmi.html\)](http://www.oasis-open.org/cover/xmi.html).

8.1.2 Editing by Using a Related Editor

The editors enable you to edit projects or other items listed in the Navigator view. Designer comes with a number of editors that you can use for editing, with a default editor designated for most file types. To select an editor other than the default editor:

- 1 Expand a project in the Navigator view.
- 2 Select the `documentation\images` or the `resources` folder.

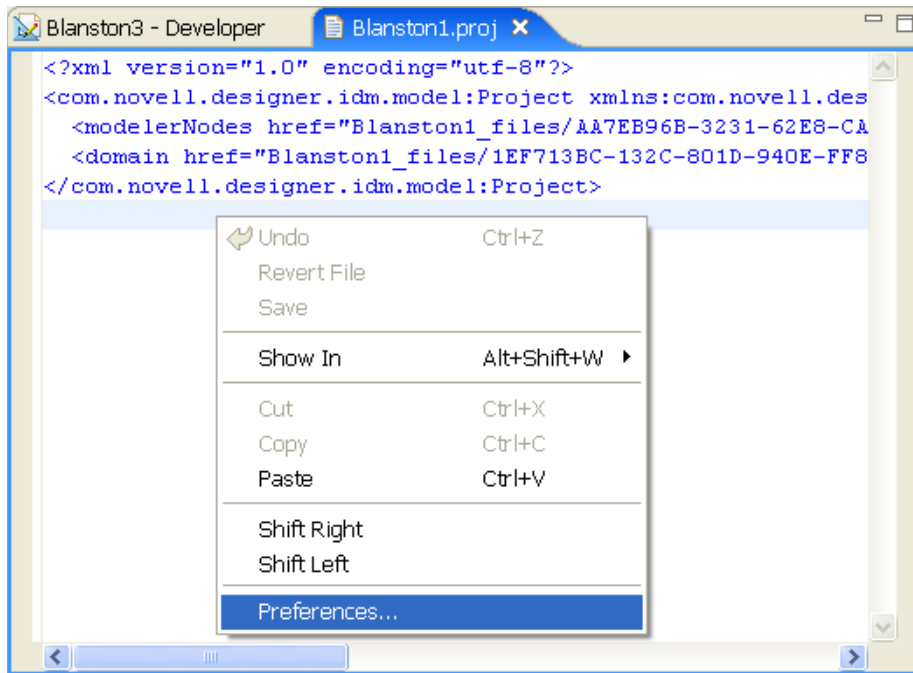
- 3 Right-click on any of the files, such as .docgen, .xml, .pdf, or .txt, and select *Open With > Editor Name*. The example below shows the file opening with the Text Editor from the Navigator view.



The editor you choose depends on the type of file you click on.

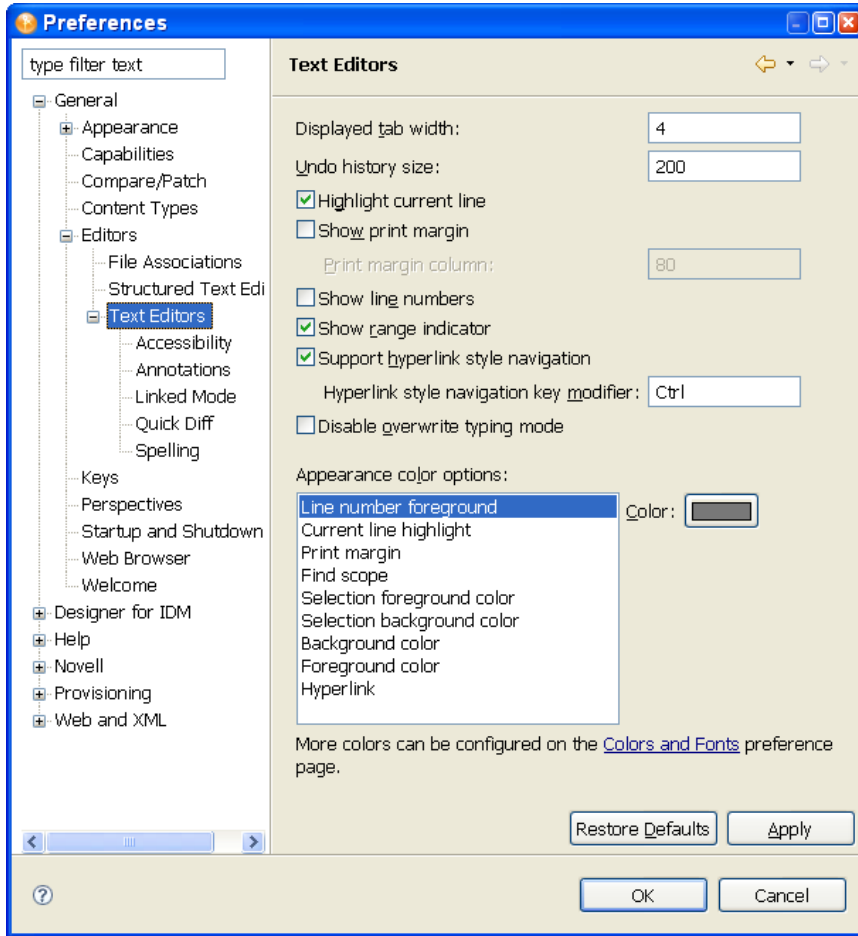
The following figure illustrates a text editor. When you right-click the editor, a menu is available:

Figure 8-3 *The Text Editor's Menu*



The Preferences window lets you set a text editor's preferences. You get to this screen by selecting *Window > Preferences > General > Editors > Text Editors*.

Figure 8-4 Preferences: Text Editors



For information on the settings, see “Text Editors” on page 446.

Each file type you right-click brings up a list of editors.

- ◆ .docgen extension:
 - ◆ Style Editor (default)
 - ◆ Text Editor
 - ◆ System Editor
 - ◆ In-Place Editor
 - ◆ Default Editor
- ◆ .xml extension:
 - ◆ Novell XML Editor (default)
 - ◆ Text Editor
 - ◆ System Editor
 - ◆ In-Place Editor
 - ◆ Default Editor

- ◆ `.txt` extension:
 - ◆ Text Editor (default)
 - ◆ System Editor
 - ◆ In-Place Editor
 - ◆ Default Editor
- ◆ `.pdf` extension:
 - ◆ Text Editor
 - ◆ System Editor (default)
 - ◆ In-Place Editor
 - ◆ Default Editor
- ◆ `.project` extension:
 - ◆ Text Editor (default)
 - ◆ System Editor
 - ◆ In-Place Editor
 - ◆ Default Editor
- ◆ `.proj` extension:
 - ◆ Modeler (default)
 - ◆ Text Editor
 - ◆ System Editor
 - ◆ In-Place Editor
 - ◆ Default Editor
- ◆ `.log` extension:
 - ◆ DS Trace Editor
 - ◆ Text Editor
 - ◆ System Editor
 - ◆ In-Place Editor
 - ◆ Default Editor

8.2 Managing Perspectives

A perspective is a collection of related views and editors. To get to the perspective choices, select *Window > Open Perspective > Other*. The following perspectives are available:

- ◆ **Designer:** Provides views (for example, Project, Outline, Property, and Tasks) and editors (for example, the Modeler or the Policy Editor) to help you design and deploy Identity Management solutions.
- ◆ **Resource:** A generic perspective. You can use this perspective for projects other than Designer.
- ◆ **exteNd Perspective:** For XML development to the exteNd platform.
- ◆ **Debug:** Used for XML development.
- ◆ [Section 8.2.1, “Opening a Perspective,” on page 106](#)

- ♦ [Section 8.2.2, “Resetting a Perspective,” on page 107](#)
- ♦ [Section 8.2.3, “Closing a Perspective,” on page 107](#)

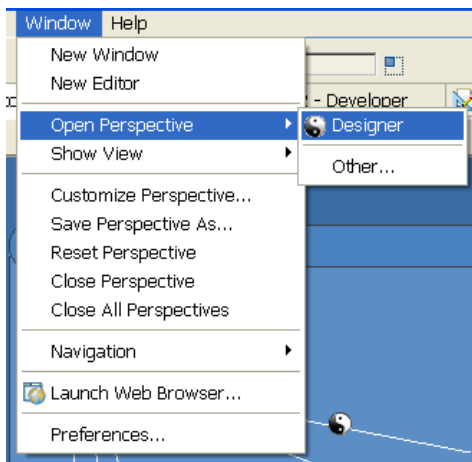
8.2.1 Opening a Perspective

The Designer perspective should be open and active the first time you run the application. If you close the perspective and want to reopen it, use menus or the Perspective Bar.

- ♦ [“Using Menus to Open a Perspective” on page 106](#)
- ♦ [“Using the Perspective Bar to Open a Perspective” on page 106](#)

Using Menus to Open a Perspective

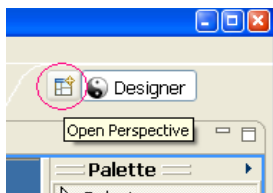
- 1 On the main menu, click *Window > Open Perspective*.



- 2 Select *Designer*.
- 3 Click *OK*.

Using the Perspective Bar to Open a Perspective

- 1 On the main Perspective Bar (found on the right side, on the same level as the toolbar), click the *Perspective Switcher* icon.



- 2 From the list, select *Designer*, then click *OK*.
You can also double-click the Designer label.

8.2.2 Resetting a Perspective

After moving or resizing items in the Modeler workspace, you might want to return your perspective to its original layout. To quickly reset items in the workspace:

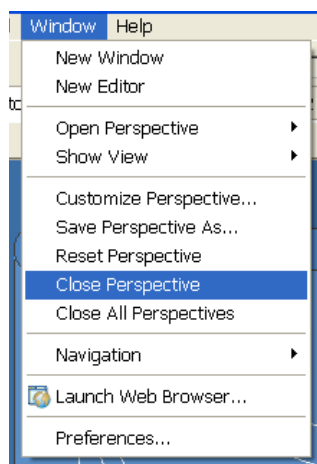
- 1 From the main menu, click *Window*.
- 2 Click *Reset Perspective*.

8.2.3 Closing a Perspective

Do one of the following:

- ◆ On the main menu, select *Window > Close Perspective* (or *Close All Perspectives*).

Figure 8-5 Options to Close a Perspective



- ◆ On the Perspective Bar, right-click the perspective, then click *Close*.

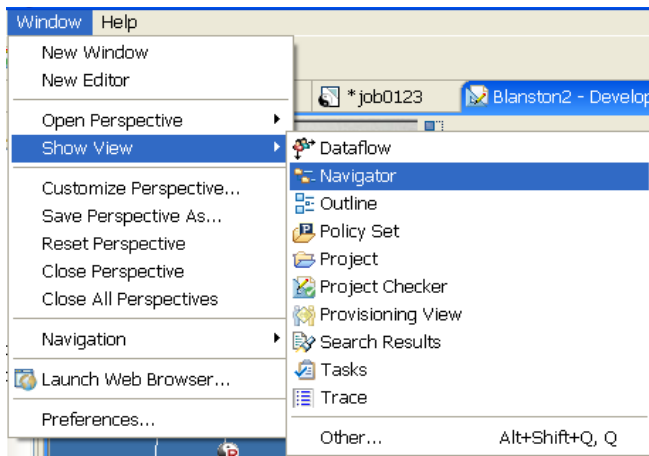
8.3 Managing Views

Views enable you to view or edit projects.

- ◆ [Section 8.3.1, “Opening a View,” on page 108](#)
- ◆ [Section 8.3.2, “Moving a View,” on page 108](#)
- ◆ [Section 8.3.3, “Minimizing and Maximizing Views,” on page 109](#)
- ◆ [Section 8.3.4, “Closing a View,” on page 109](#)
- ◆ [Section 8.3.5, “Reopening a View Window,” on page 110](#)
- ◆ [Section 8.3.6, “Using Fast Views,” on page 111](#)

8.3.1 Opening a View

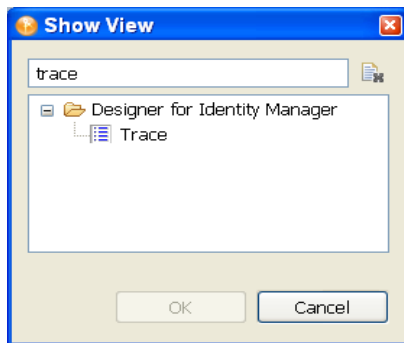
Figure 8-6 Selecting a View to Open



To open a view, select *Windows > Show View*, then select an option.

To narrow the list of views, use the view filter:

- 1 Select *Window > Show View > Other*.
- 2 In the edit box, type the name of the view that you want (for example, Trace).



You can open, maximize, and close these views, arrange them however you want, or make Fast Views from them. See [Section 8.3.6, “Using Fast Views,” on page 111](#).

To quickly expand a view, double-click the view’s tab.

8.3.2 Moving a View


To move a view, click the view’s tab, then drag it to one of the following locations:

- ♦ The left, bottom, top, or right of Designer
- ♦ Stacked with other views
- ♦ Outside of Designer, where it floats on your desktop

If two or more views are open, you can switch their order.


8.3.3 Minimizing and Maximizing Views

To minimize a view, do one of the following:

- ◆ Click the *Minimize* button in the view's title bar .
- ◆ Press Ctrl+M.

To restore the view to its original size, click the *Restore* button in the view's title bar.

To maximize a view or editor, do one of the following

- ◆ Double-click the view's tab.
- ◆ Click the *Maximize* button .
- ◆ Press Ctrl+M.

When expanded, the view or editor fills the entire application space. This feature provides maximum working space.

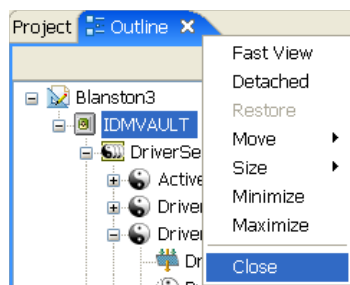
To restore the editor or view to its original dimensions, double-click the view's tab.

8.3.4 Closing a View

To close a view or editor, do one of the following:

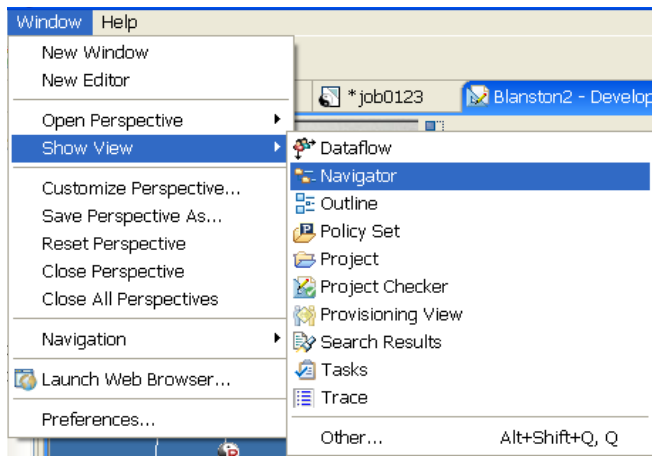
- ◆ Click the *X* on the title bar.
- ◆ Right-click the title bar, then select *Close*.

Figure 8-7 Options to Close a View



8.3.5 Reopening a View Window

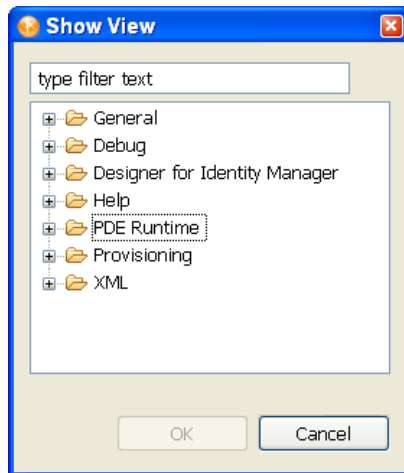
- 1 From the main menu, select *Window > Show View*.



- 2 Select one of the views (for example, Navigator).

You can also select *Window > Show View > Other*, then select one of the views under Designer for Identity Manager (for example, Projects).

Figure 8-8 Other View Choices




The views you can select are as follows:

- ◆ General
 - ◆ Bookmarks
 - ◆ Console
 - ◆ Internal Web Browser
 - ◆ Navigator
 - ◆ Outline
 - ◆ Problems

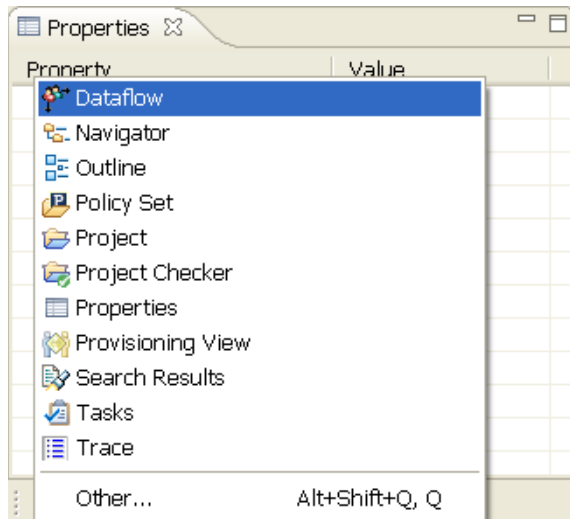
- ◆ Project Explorer
- ◆ Properties
- ◆ Tasks
- ◆ Debug
 - ◆ Breakpoints
 - ◆ Debug
 - ◆ Expressions
 - ◆ Memory
 - ◆ Registers
 - ◆ Variables
- ◆ Designer for Identity Manager
 - ◆ Dataflow
 - ◆ eDirectory Browser
 - ◆ Policy Set
 - ◆ Project
 - ◆ Project Checker
 - ◆ Search Results
 - ◆ Trace
- ◆ Help
 - ◆ Help
- ◆ PDE Runtime
 - ◆ Error Log
 - ◆ Plug-In Registry
- ◆ Provisioning
 - ◆ Data Item Mapping
 - ◆ E-mail Notification
 - ◆ Provisioning View
- ◆ XML
 - ◆ XPath Navigator

8.3.6 Using Fast Views

Fast Views provide an efficient way to manage certain types of views that you don't want to clutter the desktop, but still want to have readily accessible. To create a Fast View perspective:

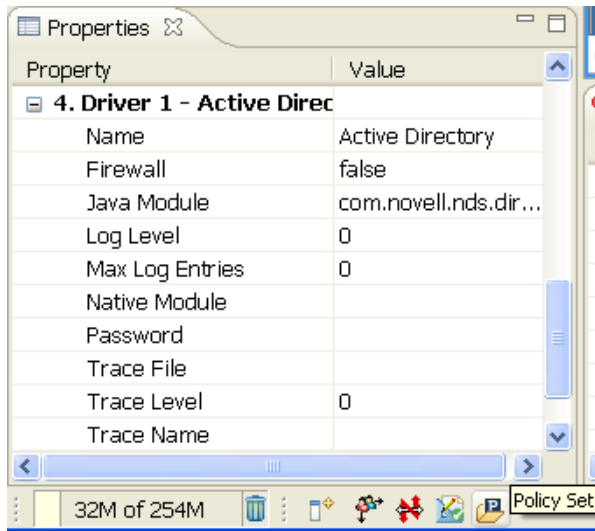
- 1 Click the *Show View as a Fast View Perspective* icon  in the bottom left corner of the Designer window (under the Properties view).

2 Select a View.



3 The view appears to the left of the Modeler view. Right-click the view's tab and deselect *Fast View* to see the view's default placement in Designer.

4 Reselect *Fast View* to have the view with its icon appear next to the *Show View as a Fast View Perspective* icon.



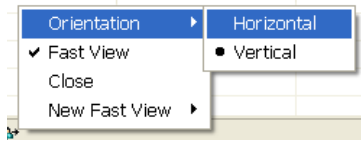
After the icon appears next to the *Show View as a Fast View Perspective* icon, do the following to restore a Fast View to a normal view:

- 1** Right-click the *Fast View* icon.
- 2** Select *Fast View*.
- 3** Reselect *Fast View* to have the view with its icon appear next to the *Show View as a Fast View Perspective* icon.

You can also set the following:

- ◆ Whether you want the Fast View to appear horizontally or vertically (the *Orientation* option)

Figure 8-9 Options for Fast Views



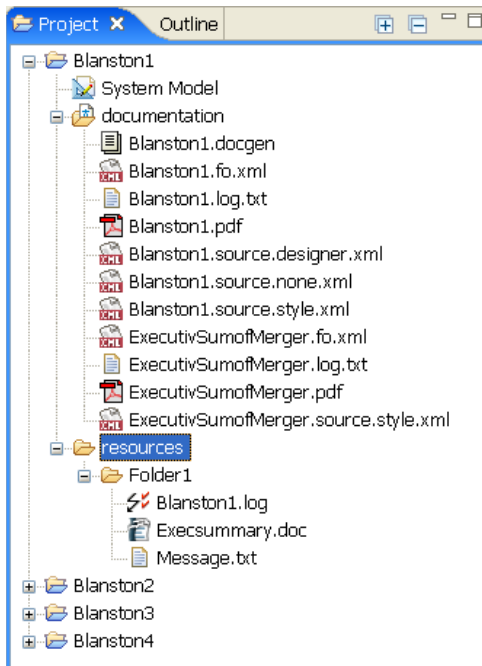
- ◆ Where you want to dock the Fast View

Views and selected items that appear by default in the Modeler view do not have the Fast View option.

8.4 The Project View

The Designer Project view lists all of your projects so that you can quickly find, open, view, navigate, manage, close, or delete any project.

Figure 8-10 The Project view



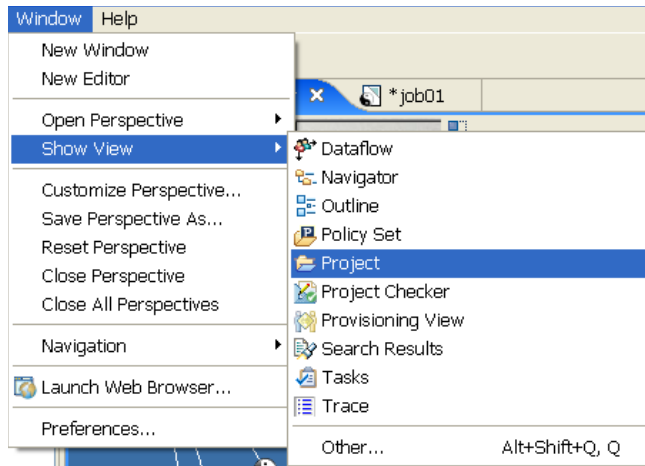
- ◆ [Section 8.4.1, “Viewing and Editing Projects: The Project View,”](#) on page 113
- ◆ [Section 8.4.2, “Tasks You Can Perform from the Project View,”](#) on page 114
- ◆ [Section 8.4.3, “What You See in the Project View,”](#) on page 116

8.4.1 Viewing and Editing Projects: The Project View

To open the Project view:

- 1 Click the *Project* tab in the upper left view of Designer.

2 If the view is not present, click *Window > Show View* from the main menu, then click *Project*.



8.4.2 Tasks You Can Perform from the Project View

You can perform the following operations in the Project view.

Table 8-1 *Project View Operations*

Operation	Description
Creating	You can create projects with an easy-to-use wizard. You can also import projects from the file system or an Identity Vault. Right-clicking a folder allows you to create a new folder or a new file within that folder.
Importing	You can import projects from the Identity Vault, from an iManager export file, or from the file system.
Opening	Double-clicking the <i>System Model</i> icon opens the project into the Modeler view. If more than one editor is registered for the item you are opening, you can select which editor you want to open it with. Double-clicking the <i>Project Name</i> tab in the Modeler expands the Modeler to full screen and double-clicking the <i>Project Name</i> tab again brings the Modeler back to its default view size. All tabs work this way, including the <i>Project</i> tab.
Deleting	To delete a project (or multiple projects), right-click the project, then select <i>Delete</i> . Or select a project, then press the <i>Delete</i> key.
Copying and Moving a Project	To copy a project, right-click the project, then select <i>Copy Project</i> . Give the project a name, select where to place the project contents, then click <i>OK</i> . To move a project, right-click the project, then select <i>Move</i> . In the Move Project window, select the location for the project, then click <i>OK</i> .
Copy and Paste	You can copy files from one project to another by using <i>Copy</i> and <i>Paste</i> . Dragging and dropping a file from one documentation folder to another works like a copy and paste menu selection.
Drag and Drop	You can drag and drop files from one project to another and from one documentation or resource folder to another.

Operation	Description
Viewing Properties	You can view the properties of a project by right-clicking a project and selecting <i>Properties</i> . For convenience, the Properties view is located just below the Project view and allows you to edit the properties.
Saving	<p>When a project is first created, it doesn't need to be explicitly saved because it's already created in the file system. Whenever a project changes (through modifications made in an editor) and you are exiting that project, you see a <i>Save Resources</i> dialog box, which allows you to save your work.</p> <p>To manually save after opening the project in an editor, do one of the following:</p> <ul style="list-style-type: none"> ◆ From the Main menu, click <i>File > Save</i> (or <i>Save All</i>). ◆ On the main menu, click <i>File > Close</i> (or <i>Close All</i>) or click the <i>X</i> in the <i>Modeler's</i> tab. If something was changed in the editor, you are prompted to save. ◆ Click the <i>Save</i> icon. <p>For information on the file format, see Section , "What is XMI?," on page 101.</p>

Right-click any of the view tabs in the Project Group view to perform the following functions.

Figure 8-11 Options in the Project Functions

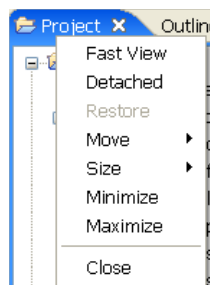


Table 8-2 Right-Click Operations

Operation	Description
Fast View	Places the Project view in the lower left corner of the Designer screen.
Detached	<p>Detaches the Project pane from the upper right location. You can then move the pane to another location in the Designer window. The pane size adjusts to the area where you place it. This is a feature of Eclipse that allows you to design your own look and feel for the Designer utility.</p> <p>To reattach the pane to the upper left location, drag the detached pane to just below the Outline/Navigator panes, then position the pane just below the colored Project bar. When you see the folders icon appear next to the Outline/Navigator tabs, release the mouse button.</p>
Restore	Restores a collapsed pane to its normal size.
Move	<i>Move View</i> allows you to move the selected tab to another location in the Designer window. <i>Move Tab Group</i> allows you to move all of the tabs connected to that group to another location in the Designer window.
Size	Allows you to adjust the size of the tab group's pane.

Operation	Description
Minimize	Allows you to collapse the selected tab's pane. You can also click the <i>Minimize/Maximize</i> icons in the right corner of each tab.
Maximize	Restores a collapsed pane to its normal size. You can also click the <i>Minimize/Maximize</i> icons in the right corner of each tab.
Close	Closes the tab. To re-open a closed tab, select <i>Window > Show View > Project</i> (or whatever tab you want to open).

Icons to the right of the Project view tab change, depending on the tab you have open. With the Project tab selected, you see the following icons:

Figure 8-12 Project Tab Icons

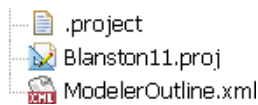





Table 8-3 Project Tab Icons

Name	Description
<i>Expand Selected Elements (+)</i>	Allows you to see anything under the item that you have selected. For example, in the <i>Project</i> tab, selecting this icon shows all items in all of the projects. In the <i>Outline</i> tab, select a driver and click this icon to expand all of the items associated with that driver.
	
<i>Collapse Selected Elements (-)</i>	Allows you to collapse all of the items that you have selected. For example, in the <i>Project</i> tab, selecting this icon collapses all items to show only the project titles. In the <i>Outline</i> tab, select a driver and click this icon to collapse all of the items associated with that driver.
	
<i>Minimize/Maximize</i>	<i>Minimize</i> allows you to collapse the selected tab's view. <i>Maximize</i> restores the collapsed pane to its normal size.
	

8.4.3 What You See in the Project View

What you see when you click the Project tab depends on what is open. Below is an open Project tab showing most of the items you can see for a project.

Figure 8-13 Project View Icons

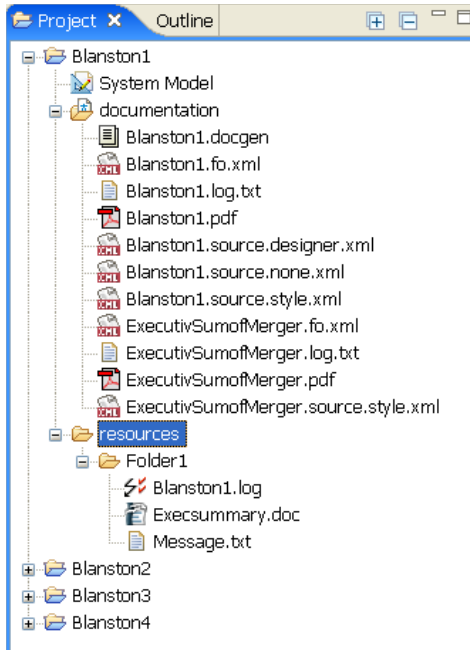









Table 8-4 Project View Icons

Icon Name	Description
	The name of the created project. It is the name of the project stored in the selected workspace directory. Double-clicking the Project name expands and collapses the entries under the project.
	<p>Right-clicking a project name allows you to do the following:</p> <ul style="list-style-type: none"> ◆ Create a new Identity Manager Project. ◆ Import an Identity Manager Project from: <ul style="list-style-type: none"> ◆ An Identity Vault ◆ An iManager Export file ◆ A file system ◆ Refresh the project. ◆ Make a copy of the project. ◆ Move the project to another workspace. ◆ Delete the project. ◆ View the project's properties window.

Icon Name	Description
<i>System Model</i>	Double-clicking the <i>System Model</i> entry puts the project into the Modeler view. You can create or import driver sets, drivers, channels, or policies into the Modeler to modify the project as needed. Then you can deploy the driver set, drivers, channels, or policies into an Identity Manager system.
 System Model	Right-clicking the System Model allows you to: <ul style="list-style-type: none">◆ Create a new Identity Manager Project.◆ Import an Identity Manager Project from:<ul style="list-style-type: none">◆ An Identity Vault◆ An iManager Export file◆ A file system◆ Open the project in the Modeler.

Icon Name	Description
<p><i>Documentation</i></p>  Documentation	<p>Shows all of the files that have been saved under the <code>documentation</code> folder. For example, under <code>Blanston1</code> you see XML Source files (icon), PDF documents (icon), and the log file. In addition, <code>Blanston2</code> contains a Document Generator style (<code>.docgen</code> with the * icon) that can be used as a template for all documents generated, as well as PDF, RTF, and text files.</p> <p>Right-clicking the <code>documentation</code> folder allows you to:</p> <ul style="list-style-type: none"> ◆ Create a new Identity Manager Project. ◆ Create a new file. ◆ Open a wizard to create a document export template wizard. The wizard generates a style (template with a <code>.docgen</code> extension) that displays under the <code>documentation</code> folder. (See Chapter 16, “Documenting Projects,” on page 299 for more information.) ◆ Import an Identity Manager Project from: <ul style="list-style-type: none"> ◆ An Identity Vault ◆ An iManager Export file ◆ A file system ◆ Import a file or a folder. ◆ Copy the <code>documentation</code> folder. ◆ Paste items into the <code>documentation</code> folder. ◆ Refresh the folder’s view. <p>Right-clicking an entry under Documentation allows you to open the file in a text or a system editor, where you can do the following:</p> <ul style="list-style-type: none"> ◆ Create a new Identity Manager Project. ◆ Import an Identity Manager Project from: <ul style="list-style-type: none"> ◆ An Identity Vault ◆ An iManager Export file ◆ A file system ◆ Use <i>Copy</i> and <i>Paste</i> to copy documentation files to different projects or folders. ◆ Refresh the folder’s contents. ◆ On Windows machines, <code>.txt</code> files are opened by default in a text editor in the Modeler view. ◆ On Windows machines, <code>.pdf</code> files are opened in Adobe* Reader if it is available on the machine. ◆ Open the file with the Text or System Editor. You can also double-click the file to use the default editor assigned to the file type. ◆ Rename a file. ◆ Delete a file. ◆ Right-click a <code>.docgen</code> file to run the Document Generator for this style. See Editing by Using a Related Editor for more options.

Icon Name	Description
<p><i>Resources</i></p> 	<p>The new <code>resources</code> folder under each project allows you to place any files that you want included with your Designer projects. For example, you can place DTrace, HTML, XML, OpenOffice, etc. files in this folder and then double-click the file to use the default editors within Eclipse or on the file system to edit those files. You can drag and drop files from the desktop or from your favorite file browser.</p> <p>Right-clicking the <code>resources</code> folder allows you to:</p> <ul style="list-style-type: none"> ◆ Create a new Identity Manager Project. ◆ Add a new folder. ◆ Add a new file. ◆ Import an Identity Manager Project from: <ul style="list-style-type: none"> ◆ An Identity Vault ◆ An iManager Export file ◆ A file system ◆ Import a file ◆ Import a folder ◆ Copy, paste, open, rename, and delete subfolders. ◆ Refresh the folder's view.
<p><i>.docgen File</i></p> 	<p>Right-clicking the <code>.docgen</code> file entry allows you to:</p> <ul style="list-style-type: none"> ◆ Create a new Identity Manager Project. ◆ Import an Identity Manager Project from: <ul style="list-style-type: none"> ◆ An Identity Vault ◆ An iManager Export file ◆ A file system ◆ Copy and delete the <code>.docgen</code> file. Use <i>Copy</i> and <i>Paste</i> to copy documentation files to different projects or folders. ◆ Refresh and rename the file. ◆ Open the <code>.docgen</code> file with the Style Editor. You can also do the same by double-clicking the <code>.docgen</code> file. ◆ Generate a document for this style.

Icon Name	Description
Additional File Types	<p>Because the <code>resources</code> folder can contain many different files, you can see different icons displayed, depending on the file extension.</p> <p>Right-clicking these any of these files allows you to:</p> <ul style="list-style-type: none"> ◆ Create a new Identity Manager Project. ◆ Import an Identity Manager Project from: <ul style="list-style-type: none"> ◆ An Identity Vault ◆ An iManager Export file ◆ A file system ◆ Copy and delete the file. Use Copy and Paste to copy documentation files to different projects or folders. ◆ Refresh and rename the file. ◆ Open the file or open the file with a text or system Editor. You can also double-click the file and have Designer choose an editor for the file. ◆ Delete the file.
 resources	

8.5 The Outline View

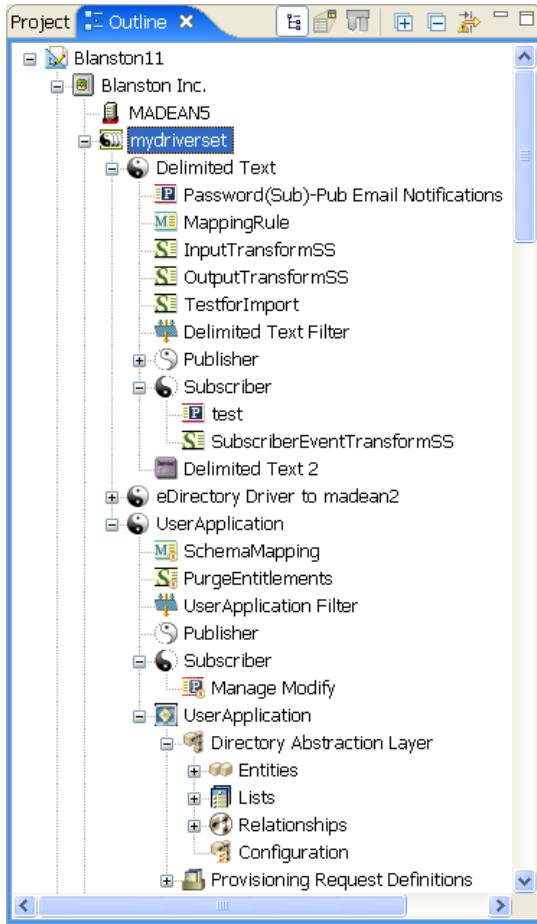
The Outline view provides an outline of the active editor. The contents vary, depending on the editor.

- ◆ For the standard Designer editors, you see a tree view of the information, a thumbnail graphical view, or a Policy Flow view, depending on which mode you are in.
- ◆ If you open an XML editor on an XML file in the Navigator view (not the embedded XML editor that's part of the Designer editors), you see a tree view of the XML document.
- ◆ [Section 8.5.1, “Viewing Contents of the Active Editor: The Outline View,” on page 121](#)
- ◆ [Section 8.5.2, “Tasks You Can Perform in the Outline View,” on page 122](#)
- ◆ [Section 8.5.3, “The Outline View Tab,” on page 123](#)

8.5.1 Viewing Contents of the Active Editor: The Outline View

Typically, the *Outline* tab displays next to the Project tab or the Navigator tab. If it isn't visible, click *Window > Show View > Outline*.

Figure 8-14 An Outline Tab and View




8.5.2 Tasks You Can Perform in the Outline View







Right-click any of the tabs in the Project Group View to perform the functions explained in [Section 8.4.2, “Tasks You Can Perform from the Project View,”](#) on page 114. There are also some icons to the right of the Outline tab. With the *Outline* tab selected, you see the following icons:

Figure 8-15 Outline Tab Icons



Table 8-5 Outline Tab Icons (from Left to Right)

Icon Name	Description
Show Model Outline	The default. Shows the project’s outline in the <i>Outline</i> tab.
	

Icon Name	Description
 <i>Show Thumbnail of Modeler</i>	Shows a thumbnail of what you see in the Modeler view.
 <i>Show Policy Flow</i>	With a driver selected in the Outline view, click this icon to see a working model of that driver's policy flow. By clicking any item in the Policy Flow modeler, you can see the policies that have been created at that level. For more information about policies, see Policies in Designer 2.1 .
 <i>Expand Selected Elements (+)</i>	Allows you to see anything under the item that you have selected. For example, in the Outline view, selecting the project name and clicking this icon shows all items in the project. Select a driver and click this icon to expand all of the items associated with that driver.
 <i>Collapse Selected Elements (-)</i>	Allows you to collapse all of the items that you have selected. For example, in the Outline view, selecting the project name and clicking this icon collapses all items to show only the project name. Select a driver and click this icon to collapse all of the items associated with that driver.
 <i>Filter</i>	<p>By enabling the Filter icon, you can select which items you want to see in the <i>Outline</i> tab. The filters include the Identity Vault, Server, Driver Set, Driver, Publisher, Subscriber, Policy, Entitlement, Application, Email Notification Template, and Design Elements.</p> <p>For example, you can filter the entitlements out of the outline view by clicking the filter and deselecting the <i>Entitlement</i> check box. This hides the entitlement objects from the Outline view. You can still add and edit entitlements in the Modeler view if this filter is enabled.</p> <p>For the E-mail Notification Template selections, Designer reads the machine's locale settings to determine which e-mail templates to display. If you want to see e-mail templates in other languages, click the <i>Filter</i> icon and select the languages.</p>
 <i>Minimize/Maximize</i>	<i>Minimize</i> allows you to collapse the selected tab's view, while <i>Maximize</i> restores the collapsed pane to its normal size.

8.5.3 The Outline View Tab

What you see when you click the *Outline* tab depends on what is open. Below is an open *Outline* tab showing most of the items you can see under a project. As you select an item (or items) in the Outline view, the selection is also made in the Modeler. If the item in the Modeler isn't visible, it automatically scrolls into view.

As items are added or removed in the Modeler, they are automatically added to the Outline view. To undo that operation from this view, right-click and select *Undo* or *Redo*.

Figure 8-16 Items You Can See under a Project

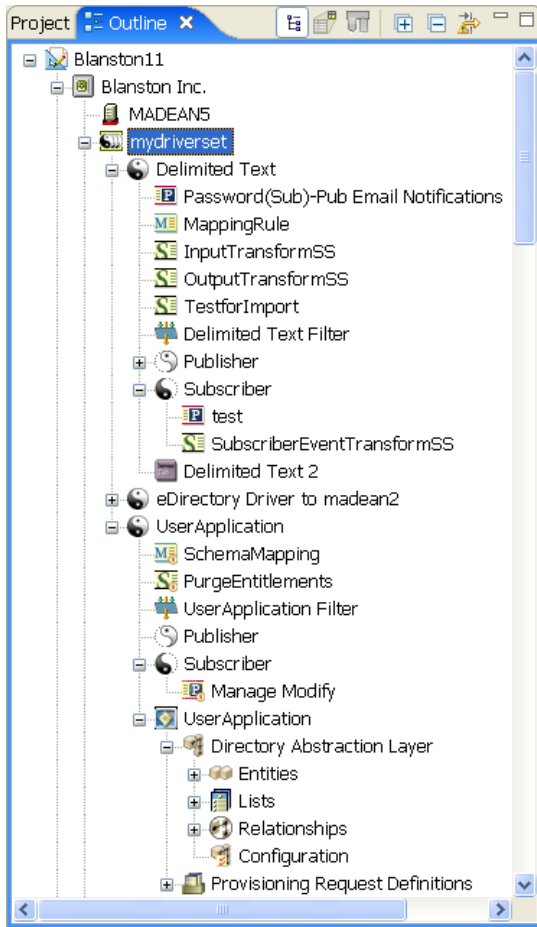









Table 8-6 Items You Can See under a Project


Icon Name	Description
	<p>Project Name The name of the project when you create it. This is the name of the project stored in the <code>\workspace\project name</code> directory. Right-click the project name to do the following:</p> <ul style="list-style-type: none"> ◆ Under <i>New</i>, you can add an application, a Domain Group, or an Identity Vault. ◆ Document the project. ◆ Through the <i>Live</i> option, you can import a driver set from the Identity Vault. ◆ Delete the project. ◆ See the Properties window associated with this object.





Icon Name	Description
	<p>An Identity Vault represents the eDirectory™ tree that has Identity Manager installed on it. Right-click the Identity Vault to perform the following:</p> <ul style="list-style-type: none"> ◆ Under <i>New</i>, you can add a driver set, a library, or a server object to the project. ◆ Manage Identity Vault schema. ◆ Generate documentation on the Identity Vault. ◆ Import schemas from a file. ◆ Export the Identity Vault configuration and schema to a file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Import a driver set from eDirectory ◆ Deploy the Identity Vault ◆ Compare all driver sets ◆ Import and deploy schema ◆ Open iManager ◆ Browse the Identity Vault ◆ Start, stop, and restart drivers, as well as show their status ◆ Delete the Identity Vault. ◆ See the Properties window associated with this object.
	<p>Domain groups allow you to group projects into granular portions. Right-click the Domain object to perform the following:</p> <ul style="list-style-type: none"> ◆ Under <i>New</i>, you can add an application, a Domain Group, or an Identity Vault. ◆ Generate documentation on the Domain. ◆ Export driver sets to a configuration file. ◆ Under <i>Live</i>, you can Import from the Identity Vault. ◆ Delete the Domain Group. ◆ See the Properties window associated with this object. <p>If you select <i>Document this Group</i>, your generated document filters items that are outside the Domain Group. Drivers bordering the domain are included in the document.</p>




Icon Name	Description
<p><i>Library</i></p> 	<p>The Library object is a repository of commonly used policies that can be referenced from multiple locations. You can place a policy in the library that every driver in the driver set can reference. Right-click the Library object to perform the following:</p> <ul style="list-style-type: none"> ◆ Under <i>New</i>, you can add the following policy types: <ul style="list-style-type: none"> ◆ Credential application ◆ Credential repository ◆ DirXML[®] Script ◆ ECMAScript ◆ Mapping Table ◆ Resource ◆ Schema Map ◆ XSLT ◆ Add a policy from a copy ◆ Export the library to a configuration file. ◆ Under <i>Live</i>, you can: <ul style="list-style-type: none"> ◆ Import a policy ◆ Delete a policy ◆ Compare a policy ◆ Delete the Library object. ◆ See the Properties window associated with this object.
<p><i>Server Object</i></p> 	<p>The Server object is the server in the tree where you have Identity Manager installed. Right-click the Server object to perform the following:</p> <ul style="list-style-type: none"> ◆ Delete the Server object. ◆ See the Properties window associated with this object.




Icon Name	Description
<i>Driver Set</i>	The driver set is a collection of Identity Manager drivers that are stored in a single location. Right-click the driver set to perform the following:
	<ul style="list-style-type: none"> ◆ Under <i>New</i>, you can create a new driver, a job, or a library. ◆ Copy the driver set settings and global configuration values. ◆ Generate documentation on the driver set. ◆ Import a driver configuration to a file. ◆ Export the driver set to a configuration file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Import a driver ◆ Deploy a driver set ◆ Compare the driver set in Designer to a deployed driver set in the Identity Vault ◆ Import and deploy driver set configuration attributes ◆ Start, stop, and restart drivers, as well as show their status ◆ Delete the driver set. ◆ See the Properties window associated with this object. <p>If you select <i>Document Selection</i>, your generated document filters most items that are outside the driver set. Child objects are also included and parent objects are included in the document for perspective purposes.</p>



Icon Name	Description
<i>Driver Object</i> 	<p>A Driver object is a collection of channels, policies, rules, and filters that connect an application to an eDirectory tree running Identity Manager. Each driver performs different tasks. Policies, rules, and filters tell the driver how to manipulate the data to perform those tasks. Right-click the Driver object to do the following:</p> <ul style="list-style-type: none"> ◆ Under <i>New</i>, you can add the following policy types: <ul style="list-style-type: none"> ◆ Credential application ◆ Credential repository ◆ DirXML Script ◆ ECMAScript ◆ Entitlement ◆ Job ◆ Mapping Table ◆ Resource ◆ Schema Map ◆ XSLT ◆ Add a policy from a copy ◆ Copy driver settings or server-specific driver settings. ◆ Show the dataflow view. ◆ Simulate policy transformation. ◆ Show the driver's policy sets. ◆ Run the Driver Configuration Wizard. ◆ Synchronize passwords to an associated server. ◆ Manage application schema. ◆ Generate documentation for this driver. ◆ Export the driver to a configuration file. ◆ Copy server-specific data between servers. To do this, you must have more than one server associated with the driver set. You can add a server to a driver set by going into the Driver Set Properties page. The replica data that you can copy includes: <ul style="list-style-type: none"> ◆ Global configuration values ◆ Named passwords ◆ Driver authentication information ◆ Driver startup option ◆ Driver parameters



Icon Name	Description
Driver Object, Continued.	<p>Right-click the Driver object to perform the following:</p> <ul style="list-style-type: none"> ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Import a driver ◆ Deploy a driver ◆ Compare the driver in Designer to a deployed driver in the Identity Vault ◆ Import and deploy driver configuration attributes ◆ Start, stop, and restart drivers, as well as show their status ◆ Set up driver security ◆ Delete the Driver object. ◆ See the Properties window associated with this object. <p>If you select <i>Document Selection</i>, your generated document will filter most items that are outside the driver. Child objects are also included and parent objects are included in the document for perspective purposes.</p>
 <i>Publisher Channel</i>	<p>A channel is a combination of rules and policies, and Designer allows you to import a channel instead of the entire driver. The Subscriber and Publisher channels describe the direction in which the information flows.</p> <p>The Publisher channel takes the event from the application, database, csv file, etc., and sends that event to the Identity Vault. Channel events do not need to be corresponding.</p> <p>Right-click the Publisher or Subscriber channel to perform the following:</p> <ul style="list-style-type: none"> ◆ Under <i>New</i>, you can add the following policy types: <ul style="list-style-type: none"> ◆ Credential application ◆ Credential repository ◆ DirXML Script ◆ ECMAScript ◆ Mapping Table ◆ Resource ◆ Schema Map ◆ XSLT ◆ Add a policy from a copy ◆ Run the Simulate Policy Transformation program on the channel to ensure that it is working correctly. ◆ Export the Channel configuration to a file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Import objects into a driver ◆ Import a policy ◆ Deploy a channel ◆ Compare the channel in Designer to a deployed driver channel in an Identity Vault ◆ See the Properties window associated with this object.


Icon Name	Description
 Subscriber Channel	<p>A channel is a combination of rules and policies, and Designer allows you to import a channel instead of the entire driver. The Subscriber channel takes the event from the Identity Vault and sends that event to whatever the receiving system is (application, database, CSV file, etc.). (See <i>Publisher Channel</i> for right-click actions.)</p>
 Policy	<p>A policy is a collection of rules and arguments that allow you to configure an application to send or receive events to or from eDirectory. You use policies to manipulate the data you receive from eDirectory or from the application. Each driver performs different tasks, and policies tell the driver how to manipulate the data to perform those tasks.</p> <p>Right-click the Policy object to perform the following:</p> <ul style="list-style-type: none"> ◆ Edit, duplicate, save as, and delete the Policy object. ◆ Run the Simulate Policy Transformation program on the policy to ensure that it is working correctly. ◆ Export a policy configuration to a file. ◆ Under Live: <ul style="list-style-type: none"> ◆ Deploy a policy ◆ Compare the policy in Designer to a deployed driver or channel policy in an Identity Vault ◆ Delete a policy. ◆ See the Properties window associated with this object.
 Overlay Icons	<p>Overlay icons have a symbol overlay in one of the corners to signify that the object is in a certain state. For example, the policy icon to the left signifies that the policy does not belong to a policy set. An application icon with an overlay signifies that there are more than one driver connected to the application. A driver set icon with a server overlay signifies that the driver set is connected to an Identity Manager server.</p> <p>Overlay icons can be informational, warnings, or they can signify errors. In the Compare window, overlay icons tell you whether the object is on the server or is in Designer.</p>
 Schema Mapping	<p>Schema Mapping allows Identity Manager to synchronize information to the connected system without changing the connected system. Schema translates all classes and attributes for the connected system.</p> <p>Right-click the Schema Mapping object to perform the following:</p> <ul style="list-style-type: none"> ◆ Edit, copy, save as, and delete the Schema Mapping object. ◆ Run the Simulate Policy Transformation program on the object to ensure that it is working correctly. ◆ Export a Schema Mapping policy configuration to an XML file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Deploy a policy ◆ Compare the policy in Designer to a deployed driver or channel policy in an Identity Vault ◆ Delete the object. ◆ See the Properties window associated with this object.

Icon Name	Description
Resource Objects	<p>Resource objects include Application SSO objects and Repository SSO objects. Double-clicking a resource object brings up the object in an editor.</p> <p>Right-click the Resource object to perform the following:</p>  <ul style="list-style-type: none"> ◆ Edit, duplicate, save as, and delete the Resource object. ◆ Export a Resource policy configuration to a file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Deploy a Resource policy ◆ Compare the policy in Designer to a deployed driver or channel policy in an Identity Vault ◆ Delete the object. ◆ See the Properties window associated with this object.
XSLT	<p>XSLT represents XSLT style sheets, which are used instead of the DirXML Script.</p> <p>Right-click the XSLT object to perform the following:</p>  <ul style="list-style-type: none"> ◆ Edit, duplicate, save as, and delete the object. ◆ Run the Simulate Policy Transformation program on the style sheet to ensure that it is working correctly. ◆ Export an XSLT style sheet configuration to an XML file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Deploy the style sheet ◆ Compare the style sheet in Designer to a deployed driver or channel style sheet in an Identity Vault ◆ Delete the object. ◆ See the Properties window associated with this object.
Filter	<p>Filter allows the objects to be sent. It's the gatekeeper that restricts the objects (groups, users, and the attributes associated with them) that can be sent to and from an Identity Vault.</p> <p>Right-click the Filter object to perform the following:</p>  <ul style="list-style-type: none"> ◆ Edit, and save the object. ◆ Run the Simulate Policy Transformation program on the filter to ensure that it is working correctly. ◆ Export a filter configuration to an XML file. ◆ Under <i>Live</i>, you can deploy the filter into an existing eDirectory tree. ◆ Clear the Filter object.

Icon Name	Description
<p><i>Application Object</i></p> 	<p>The Application object represents a connected system, whether that system is a database, an application, or CSV files.</p> <p>Right-click the Application object to do the following:</p> <ul style="list-style-type: none"> ◆ Show the dataflow view. ◆ Manage application schema ◆ Set up remote control through VNC connectivity. ◆ Generate documentation on the application object. ◆ Under <i>Driver</i>, you can manage application schemas, deploy the driver, and set up driver security. ◆ Delete the object. ◆ Under <i>Distribute</i>, you can select <i>Horizontal</i> or <i>Vertical</i>. ◆ If this is an eDir-to-eDir connection, under <i>Live</i>, you can create eDir-to-eDir certificates if this is an eDir-to-eDir connection. ◆ If this is an eDir-to-eDir connection, you can configure the connections for SSL/TLS activation. ◆ See the Properties window associated with this object. <p>If you select <i>Document Selection</i>, your generated document filters most items that are outside the driver. The driver object is also included and parent objects above the driver object are included in the document for perspective purposes.</p>
<p><i>Entitlement Object</i></p> 	<p>The Entitlement object represents an entitlement assigned to the Subscriber channel.</p> <p>Right-click the Entitlement object to perform the following:</p> <ul style="list-style-type: none"> ◆ Edit, copy, and save the object. ◆ Export an entitlement configuration to an XML file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Deploy the entitlement ◆ Compare the entitlement in Designer to a deployed entitlement in an Identity Vault ◆ Delete the object.
<p><i>Job Object</i></p> 	<p>The Job object represents a job that you can schedule to run either immediately or at a later date. Designer presently ships with three job types: Random Password Generator, Schedule Driver, and Subscriber Channel Trigger. You can also have custom jobs.</p> <p>Right-click the Job object to perform the following:</p> <ul style="list-style-type: none"> ◆ Edit the job object. ◆ Export a job configuration to an XML file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Deploy the job ◆ Compare the job in Designer to a deployed job in an Identity Vault ◆ Delete the object. ◆ See the Properties window associated with this object.

Icon Name	Description
<i>ECMAScript Object</i>	The ECMAScript object represents an ECMAScript assigned to the Subscriber Channel.
	<p>Right-click the ECMAScript object to perform the following:</p> <ul style="list-style-type: none"> ◆ Edit, copy, and save the object. ◆ Export an ECMAScript configuration to an XML file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Deploy the ECMAScript ◆ Compare the ECMAScript in Designer to a deployed ECMAScript in an Identity Vault ◆ Delete the object. ◆ See the Properties window associated with this object.
<i>Mapping Table Object</i>	The Mapping Table object represents a Mapping Table assigned to the driver.
	<p>Right-click the Mapping Table object to perform the following:</p> <ul style="list-style-type: none"> ◆ Edit, copy, and save the object. ◆ Export a Mapping Table configuration to an XML file. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Deploy the Mapping Table ◆ Compare the Mapping Table in Designer to a deployed Mapping Table in an Identity Vault ◆ Delete the object. ◆ See the Properties window associated with this object.

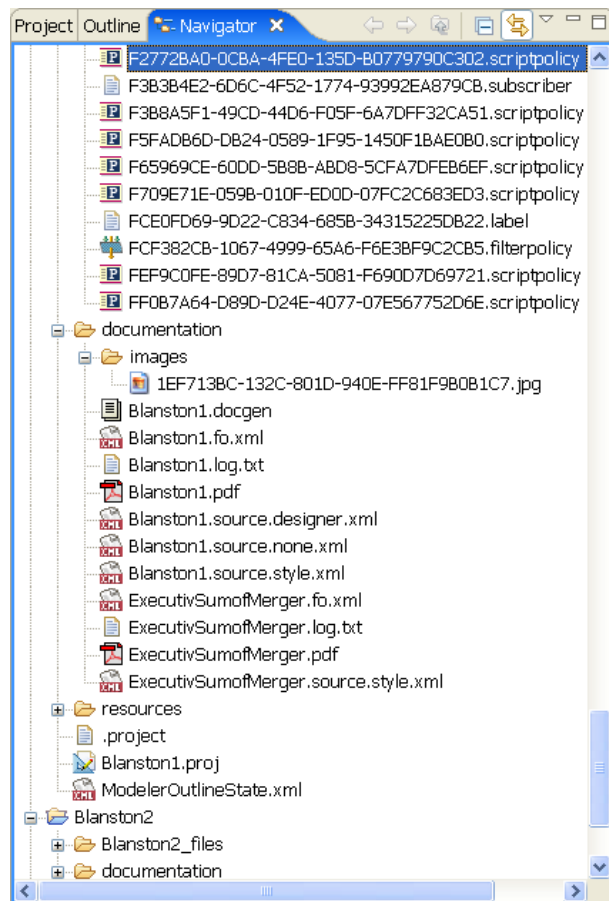
Icon Name	Description
User Application Object 	<p>The User Application object allows you to manage workflow provisioning. Right-click the User Application object to perform the following:</p> <ul style="list-style-type: none"> ◆ Show the Dataflow view. ◆ Manage application schemas. ◆ Set up remote control through VNC connectivity. ◆ Generate documentation for the User Application. ◆ Under <i>Driver</i>: <ul style="list-style-type: none"> ◆ Manage application schemas ◆ Deploy the User Application object ◆ Set up driver security ◆ Delete the User Application object. ◆ Under <i>Localize</i>: <ul style="list-style-type: none"> ◆ Export localization data ◆ Import localization files ◆ Under <i>Distribute</i>, you can select <i>Horizontal</i> or <i>Vertical</i>. ◆ Migrate the User Application object. ◆ Import the User Application object from a file. ◆ Export the User Application object to a file. ◆ Deploy the User Application object to an Identity Vault. ◆ Compare the User Application object in Designer to a deployed User Application object in an Identity Vault. ◆ Import the User Application object from an Identity Vault. ◆ Validate the User Application object. ◆ Show the Provisioning view. ◆ See the Properties window associated with this object.
Default Notification Collection 	<p>Placeholder for all notification templates.</p> <p>Right-click the Notification Collection object to perform the following:</p> <ul style="list-style-type: none"> ◆ Create a new notification template. ◆ Edit Filter. (If you want to see e-mail templates in other languages, click this option and select the languages.) ◆ Import a notification template from a file. ◆ Update templates. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Import an e-mail template from an Identity Vault ◆ Deploy e-mail templates ◆ Compare the Notification Collection in Designer to a deployed Notification Collection in an Identity Vault ◆ See the Properties window associated with this object.

Icon Name	Description
<i>Notification Object</i>	A notification template.
	<p>Right-click the Notification object to perform the following:</p> <ul style="list-style-type: none"> ◆ Edit, copy, or save a template. ◆ Under <i>Live</i>: <ul style="list-style-type: none"> ◆ Deploy e-mail templates ◆ Compare the Notification object in Designer to a deployed Notification object in an Identity Vault ◆ Delete a template. ◆ See the Properties window associated with this object.

8.6 The Navigator View



The Navigator view enables you to manage and view the file resources that are in your workspace. You can see your projects and files as they actually exist in the file system. You can add files and folders, and append them to your projects or workspace. These extra items won't change or affect the behavior of your Identity solution. However, they enable you to conveniently append other pieces of information that you want to associate with your projects and workspace.

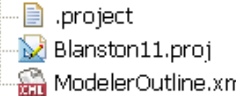
Figure 8-17 *The Navigator View*



The Navigator view enables you to do the following:

Table 8-7 Navigator View Icons

Icon Name	Description
	<p>The name of the created project. This is the name of the project stored in the selected workspace directory. Double-clicking the Project name expands and collapses the entries under the project.</p> <p>Right-clicking a project name allows you to do the following:</p> <ul style="list-style-type: none">◆ Under <i>New</i>:<ul style="list-style-type: none">◆ Create a new project◆ Create a new Identity Manager Project◆ Create a new provisioning artifact◆ Create other projects◆ See just this project's workspace (the <i>Go Into</i> option).◆ Open in a new window.◆ Copy, paste, delete, move, and rename the project.◆ Import and export the project.◆ Refresh the project.◆ Close the project or close unrelated projects.◆ Through the <i>Source</i> option, format the document.◆ View the project's Properties window.
	<p>The <i>files</i> folder contains the files of the created project. Beneath the <i>files</i> directory are the documentation and resources folders and any other folders you have created in this project.</p> <p>Right-clicking any of the project folder names allows you to do the following:</p> <ul style="list-style-type: none">◆ Under <i>New</i>:<ul style="list-style-type: none">◆ Create a new project◆ Create a new Identity Manager Project◆ Create a new provisioning artifact◆ Create other projects◆ See only this project's workspace (the <i>Go Into</i> option).◆ Open the directories in a new window.◆ Copy, paste, delete, move, and rename the folder.◆ Import and export the project.◆ Refresh the directory.◆ Through the <i>Source</i> option, format the document.◆ View the project's Properties window.

Icon Name	Description
All File Types	<p>Because the project folders can contain many different files, you can see different icons displayed, depending on the file extension. Right-clicking any of these files allows you to:</p>
	<ul style="list-style-type: none"> ◆ Under <i>New</i>: <ul style="list-style-type: none"> ◆ Create a new project ◆ Create a new Identity Manager Project ◆ Create a new provisioning artifact ◆ Create other projects ◆ Open the file in a Designer-chosen editor. ◆ Open the file with the following choices: <ul style="list-style-type: none"> ◆ Text editor ◆ System editor ◆ In-place editor ◆ Default editor ◆ XML files can choose an XML editor ◆ Proj files can choose the Modeler ◆ Filter files can choose the Filter editor ◆ Policy files can choose the Policy Builder editor ◆ Copy, paste, delete, move, and rename the files. ◆ Import and export the files. ◆ Refresh the file or directory. ◆ Through the <i>Source</i> option, format the document. ◆ View the file's Properties window.

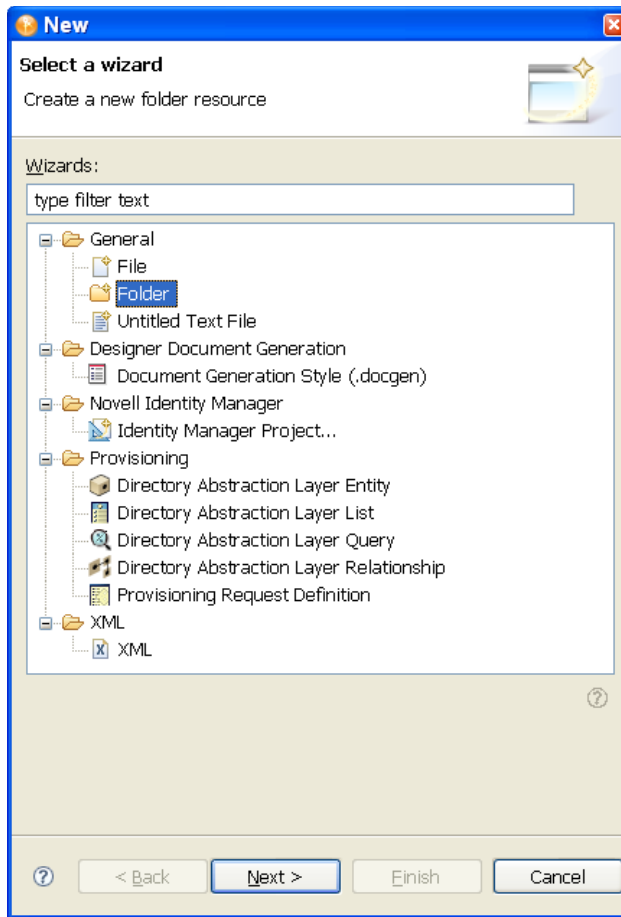
In the Navigator view, you can move any document into your project and edit it with a selected editor. For example, you can add a Word* or PowerPoint* document, then double-click the document. The native application loads inside Designer, as any other editor would, where you can view and edit the document. (This feature only works on Windows.)

You can create linked resources at any depth within projects. This allows you to put resources from different places in the file system anywhere you like in the project structure, including below other linked resources. The same resource can be linked to the same project.

To create a linked resource:

- 1 Select *Window > Show View > Navigator*.
- 2 In the Navigator view, right-click a project, then select *New > More Projects*.

3 Expand the *General* folder.



4 Select *Folder*, then click *Next*.

5 Click *Advanced*, then select *Link a folder in the filesystem*.

6 Browse to and select a folder, click *OK*, then click *Finish*.

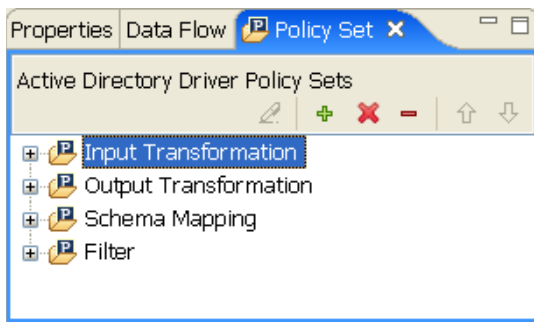
Designer creates a link to that folder directly under the project. You can then easily access the folder.

You can use the same procedure to link to a file.

8.7 The Policy Set View

The Policy Set view lists policies and displays a toolbar.

Figure 8-18 The Policy Set View



The list displays all policies contained in the selected policy set. During a transformation, the policies within the list are executed from top to bottom.

The toolbar contains buttons and a drop-down menu so that you can manage policies displayed in the list. To manage policies, you edit, add, delete, remove, or change the processing order of the policies.

The Policy Set view renders differently, depending on what mode the Outline view is in. If you are in Tree mode, the Policy Set view shows policies in a hierarchical containment. However, if you are in Policy Flow mode, the view shows a flat list of policies in the selected policy set.

To bring up a policy in the Policy Builder view, double-click the policy. For more on policies, see [Understanding Policies for Identity Manager 3.5.1](#) and [Policies in Designer 2.1](#).

8.8 The Properties View

The Properties view displays information on key-value pairs (for example, Max Log Entries 2). The information displayed depends on what is selected in the active view (for example, Modeler or Outline).

Often, the Properties view lets you edit the information. For example, you can select a driver in the Modeler and then edit the Trace Level value that is displayed in the Properties view. You can also view and edit the Administrator settings in an Identity Vault.

Figure 8-19 The Properties View of an Active Directory Driver

Property	Value
Name	Active Directory
Firewall	false
Java Module	
Log Level	0
Max Log Entries	0
Native Module	addrdriver.dll
Password	*****
Trace File	
Trace Level	0
Trace Name	
Trace Size Limit (KB)	
Use DS Log Settings	false

- ◆ [Section 8.8.1, “The Spinner Control,”](#) on page 140

- ◆ Section 8.8.2, “The DN Cell Editor,” on page 140
- ◆ Section 8.8.3, “Editing Boolean Values,” on page 140
- ◆ Section 8.8.4, “Keyboard Tips,” on page 141
- ◆ Section 8.8.5, “Resizing Column Widths,” on page 141
- ◆ Section 8.8.6, “Summary Table,” on page 141

8.8.1 The Spinner Control

Figure 8-20 *The Spinner Control*

Trace File		
Trace Level	2	↑ ↓
Trace Name		
Trace Size Limit (10		

When you edit numeric values in the Properties view, Designer displays a spinner control so that you can increase or decrease the number.

The maximum value on the spinner is set appropriately, based on the attribute type. For example, the maximum for log entries is 500.

8.8.2 The DN Cell Editor

Figure 8-21 *DN Cell Editor*

Deploy Context	domain.blanston	🔍
----------------	-----------------	---

To edit attributes that are of type DN syntax (for example, Driver Set Deploy Context):

- 1 Click *Deploy Context*.
- 2 Type the distinguished name of the context.

You can also click the *Browse* button, then browse to and select an object with the eDirectory browser. In this case, you are selecting the Distinguished Name of the container to deploy the driver set.

8.8.3 Editing Boolean Values

Figure 8-22 *Drop-Down List for Boolean Values*

Trace Size Limit (10		
Use DS Log Settin	true	▼
	false	
	true	

To edit Boolean values, select *true* or *false* from the drop-down list.

8.8.4 Keyboard Tips

Table 8-8 *Keyboard Tips*

Task	Description
Save the value of the current cell editor and set the focus on the next editor	Press Tab, Enter, the Up-arrow, or the Down-arrow.
Navigate backwards	Press Shift+Tab. Navigation loops when you reach the top or bottom of a value.
Exit the Properties view	Press Ctrl+Tab for forward navigation. Press Ctrl+Shift+Tab for backward navigation.

8.8.5 Resizing Column Widths

If you resize the columns in the Properties view, those column widths persist in user preferences. When you open the Properties view again or restart Designer, the resized widths are displayed.

8.8.6 Summary Table

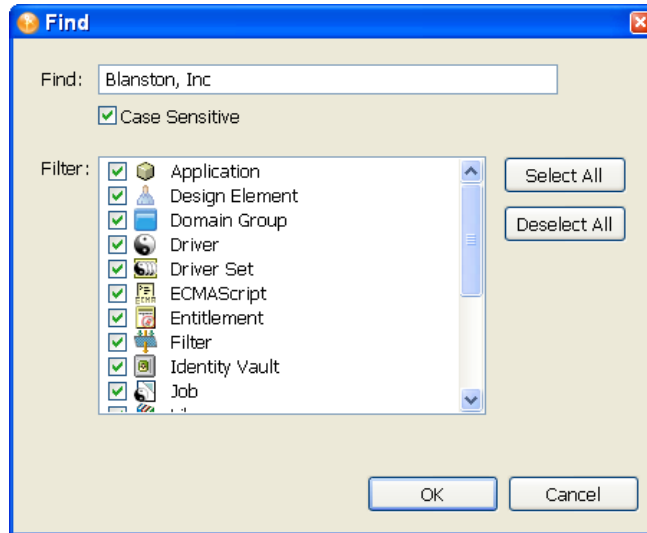
Below is a summary table of what you can do in the Properties view.

Function	Description
Opening	If the view doesn't already display, select <i>Window > Show View > Properties</i> .
Edit	You can edit the settings of any item selected in the Modeler or Outline View.
Multi-edit	If you have selected more than one item, you can edit any settings that both items have in common. For example, if they both have a Name setting, you can select both and edit once. The change is made to both items. This function can boost productivity on certain edit operations.
Saving	When you click <i>Save</i> , all changes are saved to the project file. When you reopen the project, all settings are restored.
Boolean Editor	You can edit Boolean settings (<i>true/false</i>) by clicking the value, which toggles to the other value.

8.9 The Search Results View

To search across an entire project:

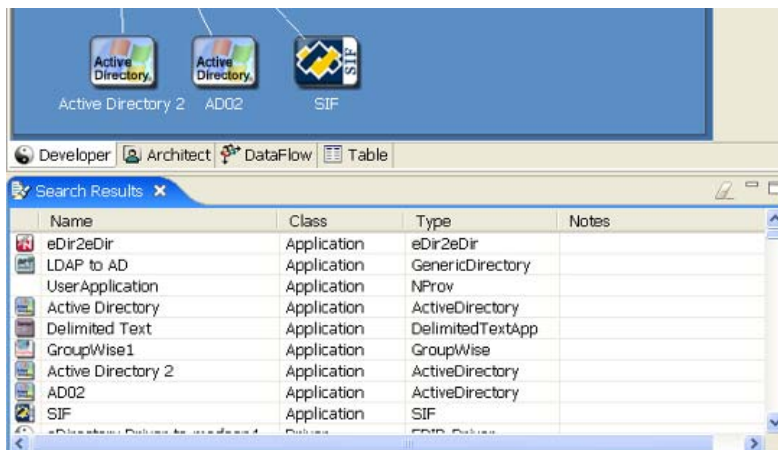
- 1 With the Modeler view active, press Ctrl+F.



- 2 Select whether the search is case-sensitive.
- 3 Filter object types you want to search on. You can select or deselect all items through the *Select All* or *Deselect All* buttons.
- 4 Click *OK*.

The results appear in the Search Results view, sorted by name, class, type, and notes. To sort on a different column, click the column headers.

Figure 8-23 The Search Results View



If the Search Results view isn't already open, the search opens it.

As you select an item (for example, a policy) in the Search view, the corresponding item (for example, the driver) is selected in the Modeler.

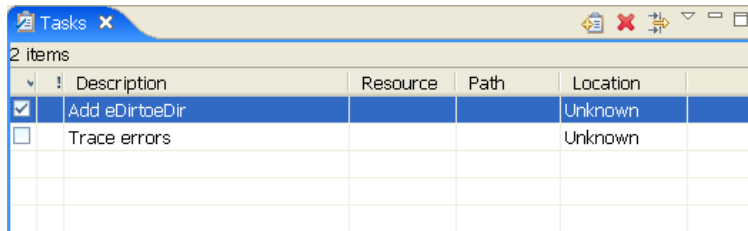
If you double-click an item, its appropriate dialog box or editor launches. To open an item with another editor, right-click the item and select *Open With*.

To clear the Search view, click the *Clear* icon in the Search view's toolbar.

When you subsequently invoke the Search dialog box, Designer remembers your latest search settings.

8.10 The Tasks View

Figure 8-24 *The Tasks View*



The Tasks view enables you to make and prioritize notes while you manage identities. This view is for your convenience. It can contain whatever you want. It won't affect the functionality of your solution.


This view enables you to do the following:

- ◆ Add tasks
- ◆ Delete tasks
- ◆ Filter on completion status
- ◆ Filter on priority
- ◆ Filter on text in the description
- ◆ Mark if completed
- ◆ Provide a description
- ◆ Set a priority
- ◆ Sort on all fields

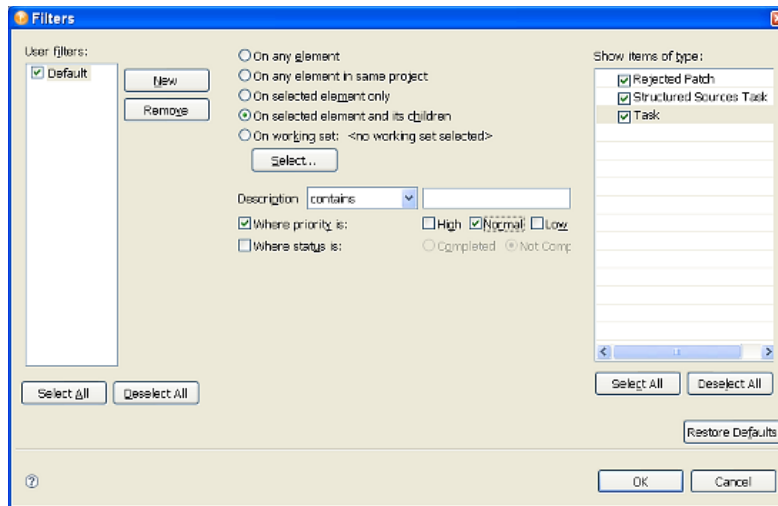
To add a task:

- 1 Open the Tasks view by selecting *Window > Show View > Tasks*.
- 2 Right-click in the Tasks view, then click *Add Task*.
- 3 Type a name in the *Description* field, select a priority, then click *OK*.

To filter tasks:

- 1 Click the filter icon .

2 Select filters, then click *OK*.



8.11 The Trace and Error Log Views

The Trace and Error Log views help you troubleshoot Designer. For more information, see the following:

- ♦ [“Turning on Trace Messages” on page 492](#)
- ♦ [“Viewing the Error Log” on page 491](#)
- ♦ [“Checking Loaded Plug-Ins” on page 494](#)

8.12 The Dataflow View

Designer allows you to manage how the data flows between the Identity Vault and the connected systems. You can see how the data flows between all of the connected systems, make changes as needed, create reports about the data, and view the flow of passwords between the systems.

The Dataflow view and the Dataflow editor manage the data. The Dataflow view displays the flow of data in the modeler per driver. The Dataflow editor displays a more granular view. For more information, see [Chapter 12, “Managing the Flow of Data,” on page 221](#).

Configuring Objects in Designer

9

Designer allows you to easily view, configure, and modify settings for domain groups, Identity Vaults, servers, driver sets, drivers, and connected systems.

- ◆ [Section 9.1, “Viewing Object Properties,” on page 145](#)
- ◆ [Section 9.2, “Configuring a Domain Group,” on page 148](#)
- ◆ [Section 9.3, “Configuring Identity Vaults,” on page 148](#)
- ◆ [Section 9.4, “Configuring Servers,” on page 150](#)
- ◆ [Section 9.5, “Configuring Driver Sets,” on page 152](#)
- ◆ [Section 9.6, “Configuring Drivers,” on page 158](#)
- ◆ [Section 9.7, “Configuring Jobs,” on page 169](#)
- ◆ [Section 9.8, “Configuring Application Properties,” on page 170](#)
- ◆ [Section 9.9, “Configuring Policies,” on page 174](#)
- ◆ [Section 9.10, “Adding Prompts to a Driver Configuration File,” on page 175](#)
- ◆ [Section 9.11, “Editing a Notification Template Name,” on page 177](#)
- ◆ [Section 9.12, “Synchronizing Passwords,” on page 177](#)

9.1 Viewing Object Properties

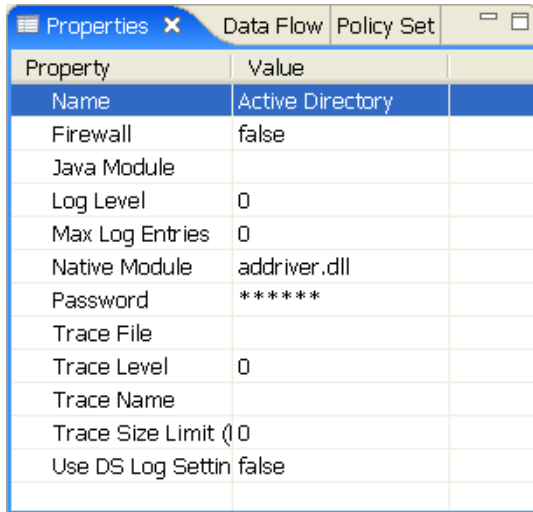
To quickly view or edit properties of items (for example, an Identity Vault or a driver), you can use the Properties view or a Properties dialog box.

- ◆ [Section 9.1.1, “Properties View,” on page 145](#)
- ◆ [Section 9.1.2, “Properties Dialog Box,” on page 146](#)
- ◆ [Section 9.1.3, “Operations Relating to Properties,” on page 147](#)

9.1.1 Properties View

If the Properties view is open when you select an item in the Modeler, information about that item displays in the Properties view. You can then quickly view or edit information.

Figure 9-1 *The Properties View*



Property	Value
Name	Active Directory
Firewall	false
Java Module	
Log Level	0
Max Log Entries	0
Native Module	addriver.dll
Password	*****
Trace File	
Trace Level	0
Trace Name	
Trace Size Limit (0	
Use DS Log Settin	false

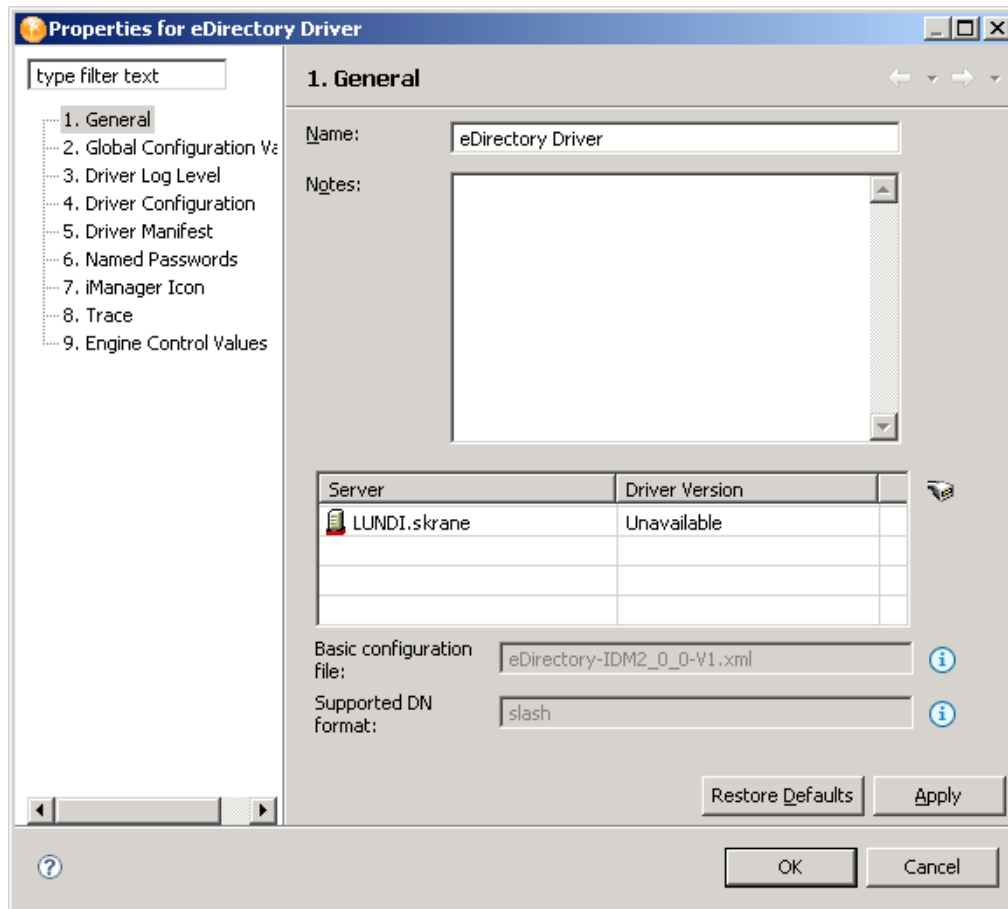
To open the Properties view, click *Window > Show View > Other > General > Properties*. For additional information, see [Section 8.8, “The Properties View,”](#) on page 139.

9.1.2 Properties Dialog Box

To view or edit properties of items:

- 1 Open the Properties dialog box by doing one of the following:
 - ◆ Double-click an item in the Modeler or in the Outline view.
 - ◆ Right-click an item (for example, an Identity Vault) in the Modeler or Outline view, then select *Properties*.
 - ◆ Select an item, then press *Enter*.
 - ◆ Select an item, then select *File > Properties*.
 - ◆ Select an item, then select *Model > [object] > Properties*.

The following figure illustrates a driver's properties page:



2 Edit settings, then click *OK* to save.

9.1.3 Operations Relating to Properties

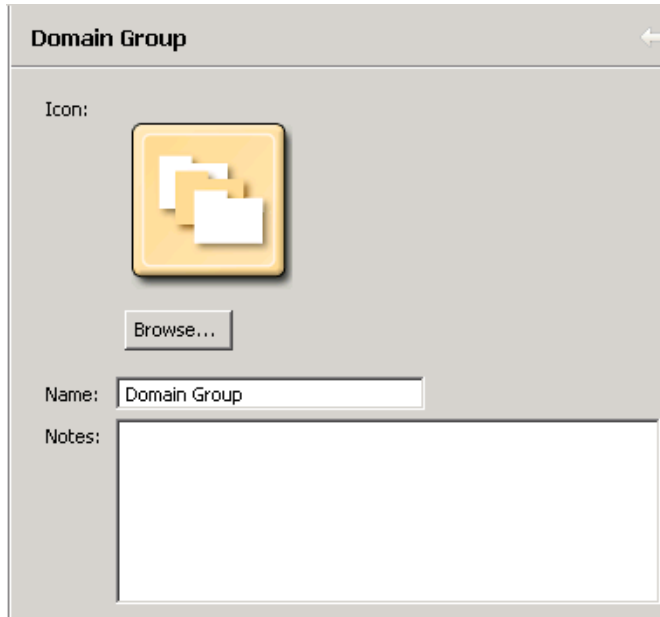
Table 9-1 *Operations Relating to Properties*

Operation	Description
Open the Properties view	Click <i>Window > Show View > Other > General > Properties</i> .
Open the Properties dialog box	Double-click an item, or right-click, then select <i>Properties</i> .
Edit settings	You can edit the settings of any item selected in the Modeler or Outline view.
View a server's properties	In the Outline view, right-click the server icon, then select <i>Properties</i> .
Save to memory or disk	When you click <i>Apply</i> or <i>OK</i> in a properties dialog box, changes are committed to memory. However, changes are not saved to disk unless you select <i>File > Save</i> .

9.2 Configuring a Domain Group

To view or change a domain group's settings, double-click the domain group.

Figure 9-2 *The Domain Group Property Page*



- 1 To change the domain group's icon, click *Browse*, then navigate to and select an image file.

By default, the *Browse* button opens the `icons/group` folder in the `com.novell.designer.core` plug-in. The default image selected is `administrative.png`. To select a different image, double-click it.

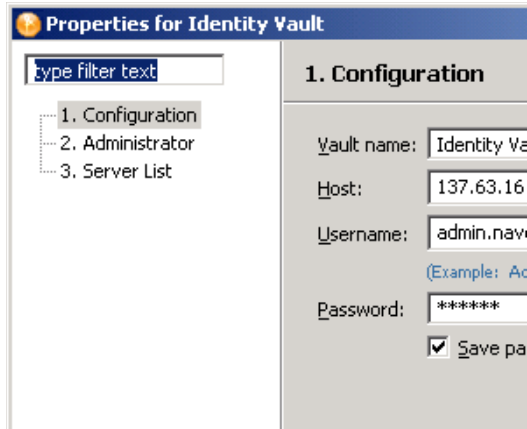
- 2 Click *Apply*.
- 3 To change the name of the domain group, edit the *Name* field.
- 4 Add details in the *Notes* pane.
- 5 Click *OK*.

The image (for example, `administrative.png`) appears to the left of the domain name in the Modeler.

9.3 Configuring Identity Vaults

To view or change an Identity Vault's settings, double-click the vault. The Identity Vault properties page has three options: *Configuration*, *Administrator*, and *Server List*.

Figure 9-3 Identity Vault Properties



Configuration Settings

Table 9-2 Configuration Settings for an Identity Vault

Field	Description
<i>Vault name</i>	The name of the Identity Vault object. The default is Identity Vault.
<i>Host</i>	The eDirectory™ host where you plan to log in and deploy.
<i>Username</i>	The eDirectory username that has sufficient rights to make changes to objects associated with this deployment.
<i>Password</i>	The password for the above user.
<i>Save Password</i>	Saves the password permanently, so that you can be authenticated into this Identity Vault without having to prompt for a password each Designer session. If you use Save Password, the password is saved locally in Designer's file system and is not secure. If you do not select this option, the password is remembered only until you close Designer.
<i>Test Credentials</i>	Attempts a connection to the eDirectory server to validate your login credentials.
<i>Deploy context</i>	The default DN container assigned to all driver sets that are associated with this Identity Vault. If you specify a DN container on the Driver Set object, that setting takes precedence over the default setting.

Administrator

The *Administrator* option is divided into three sections. Entering information in these sections is optional.

- ◆ **Personal Information:** Lets you enter information specific to the Identity Vault, such as Name, Title, Department, and Location,

- ♦ **Contact Information:** Lets you enter information such as Email, Phone, Cell phone, Pager, and Fax.
- ♦ **Notes:** Allows you to type any reminders you might need for future reference.

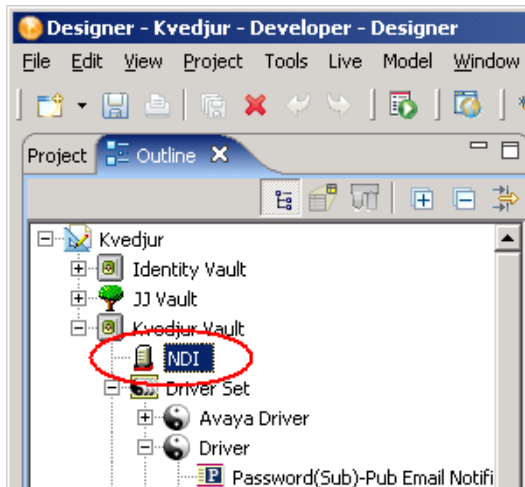
Server List

The *Server List* option displays the servers that are associated with the selected Identity Vault. You can add, edit, or remove the server entries.

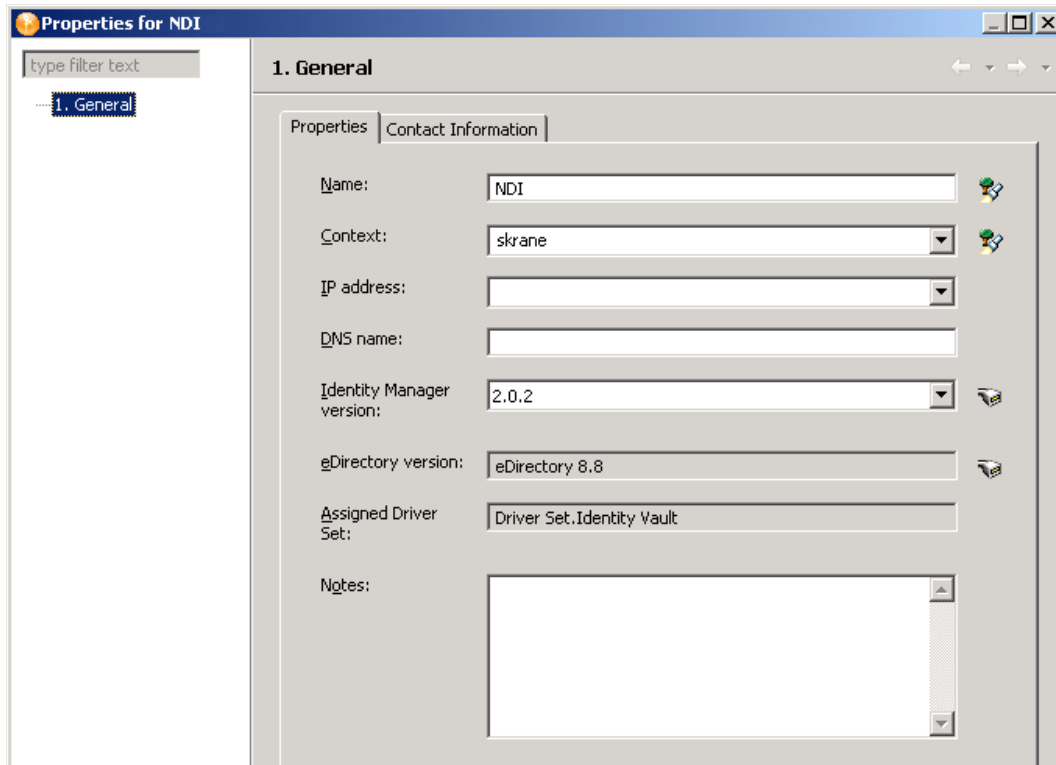
NOTE: If you select the option to allow a default server to be created, that server shows up as Default Server.default_container in the list. You cannot deploy a driver set into an existing eDirectory tree if you have Default Server.default_container in the Server List. You must first remove this reference and add a Metadirectory server in an eDirectory tree.

9.4 Configuring Servers

- 1 Right-click the server icon in the Outline view.



2 Select Properties.



The following tables lists settings for the Properties page:

Field	Description
<i>Name</i>	The name of the Server object. The Identity Vault lists the server. You can browse to and select the server.
<i>Context</i>	The server's context. The Identity Vault assigns the context. You can browse to and select the context.
<i>IP address</i>	The server's IP address.
<i>DNS name</i>	The domain name, or complete directory context name.
<i>Identity Manager version</i>	The version of Identity Manager that is running on the server. The default is Identity Manager 3.5. You can change the version by using the drop-down list. See Section 10.2, "Changing the Identity Manager Version," on page 180.
<i>eDirectory version</i>	The version of eDirectory that the server is using.
<i>Assigned driver set</i>	The driver set that the server is assigned to.
<i>Notes</i>	Information that you want to specify, to help you maintain the server.

The following figure illustrates the Contact Information page for the server:

The screenshot shows a Windows-style dialog box titled "Properties for LUNDI". On the left, there is a tree view with a search box containing "type filter text" and a single item "1. General". The main area is titled "1. General" and contains two tabs: "Properties" and "Contact Information". The "Contact Information" tab is active and contains several text input fields: Name, Title, Phone, Cell, Email, Location, Department, Pager, and Fax. Below these is a larger text area for Notes with a vertical scrollbar. At the bottom right are buttons for "Restore Defaults", "Apply", "OK", and "Cancel".

Use this page to provide information on the person to contact and other items of interest concerning the server.

9.5 Configuring Driver Sets

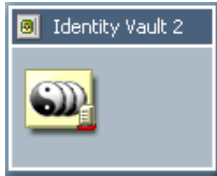
A driver set is a container that holds Identity Manager drivers. Only one driver set can be active on a server at a time. As a result, all active drivers must be grouped into the same driver set. To view or change settings, double-click a driver set in the Modeler.

- ◆ [Section 9.5.1, “Driver Set General Options,” on page 153](#)
- ◆ [Section 9.5.2, “Driver Set Global Configuration Values,” on page 154](#)
- ◆ [Section 9.5.3, “Driver Set Log Levels,” on page 154](#)
- ◆ [Section 9.5.4, “Driver Set Server List,” on page 155](#)
- ◆ [Section 9.5.5, “Driver Set Trace,” on page 156](#)

9.5.1 Driver Set General Options

When you create an Identity Vault, a driver set is added to the vault by default.

Figure 9-4 A Driver Set in an Identity Vault



You can add other driver sets by dragging the Driver Set object from the palette to the Modeler.

From the *General* page, you can specify or change driver set values.

Figure 9-5 Driver Set Properties

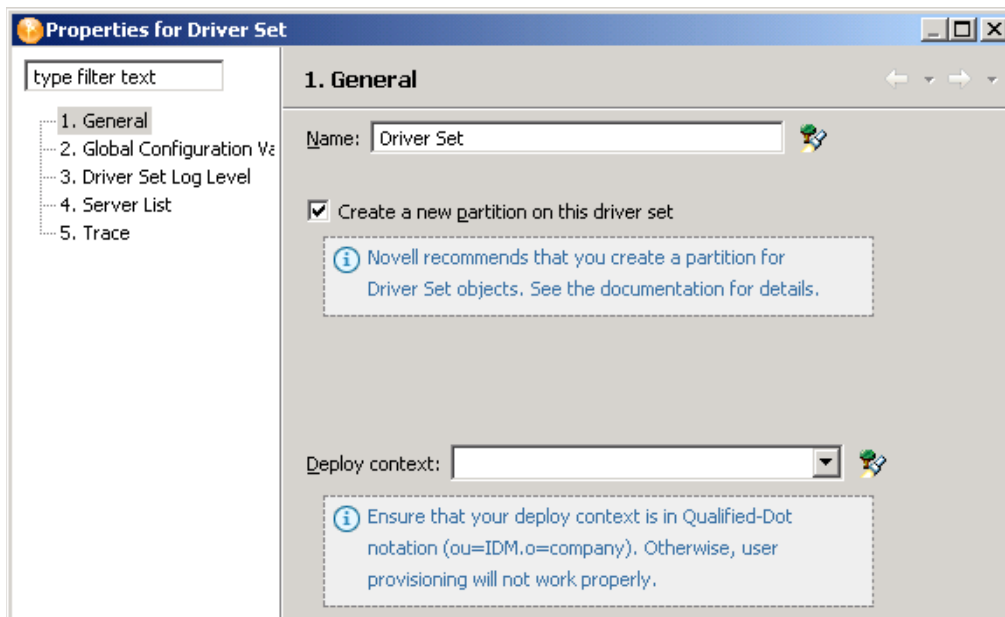


Table 9-3 Driver Set Settings

Field	Description
Name	The name of the Driver Set object (for example, DriverSet1.)
<i>Create a new partition on this driver set</i>	We recommend that you select this option. For details, see Planning the Technical Details of your Identity Manager Implementation (http://www.novell.com/documentation/idm/install/data/alorukq.html#alorukq)

Field	Description
<i>Deploy context</i>	The Identity Vault assigns the default DN container value to all driver sets. If you specify a DN container here on the Driver Set object, that setting takes precedence over the Identity Vault setting. You can manually enter this value or browse for it.

9.5.2 Driver Set Global Configuration Values

Global configuration values (GCVs) are settings that are similar to driver parameters. Global configuration values can be specified for a driver set as well as an individual driver. If a driver does not have a GCV, the driver inherits the value for that GCV from the driver set.

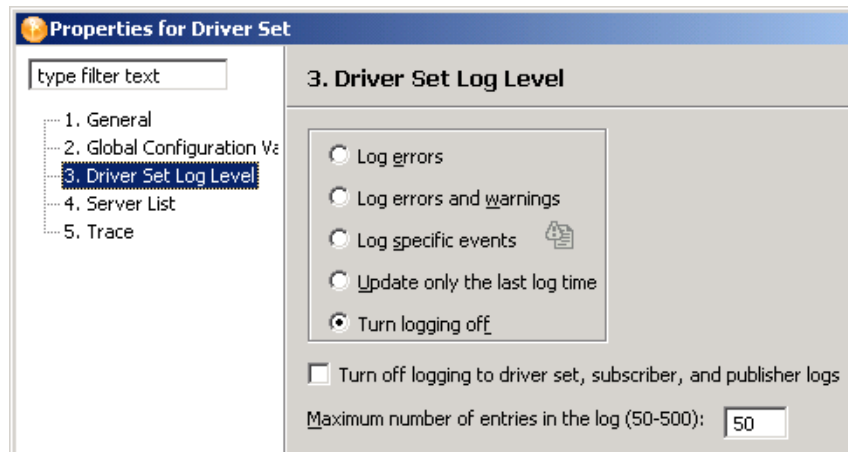
GCVs allow you to specify settings for Identity Manager features such as password synchronization and driver heartbeat, as well as settings that are specific to the function of an individual driver configuration. Some GCVs are provided with the drivers, but you can also add your own. You can refer to these values in a policy to help you customize your driver configuration.

To view or change the driver set's GCV settings, double-click the driver set. From the Global Configuration Values page, you can add, edit, or remove values, or edit the XML file for the driver set.

9.5.3 Driver Set Log Levels

The *Driver Set Log Level* options enable you to view high-level information. For lower-level information, use [“Driver Set Trace” on page 156](#).

Figure 9-6 *Driver Set Options for Logging*



By default, logging is turned off. To track errors, messages, or events, change the default.

- 1 Double-click the driver set.
- 2 Select *Driver Set Log Level*.
- 3 Select a logging option.

The log option that you select determines which messages are available in the log.

To configure audit instrumentation, select *Log specific events*, click the event selector button, select events, then click *OK*.

The *Update only the last log time* option updates the time stamp to indicate the last activity of the driver.

- 4 Specify the number of entries in the log.

The default is 50 entries (lines) in the log. If you want a longer history, increase the number.

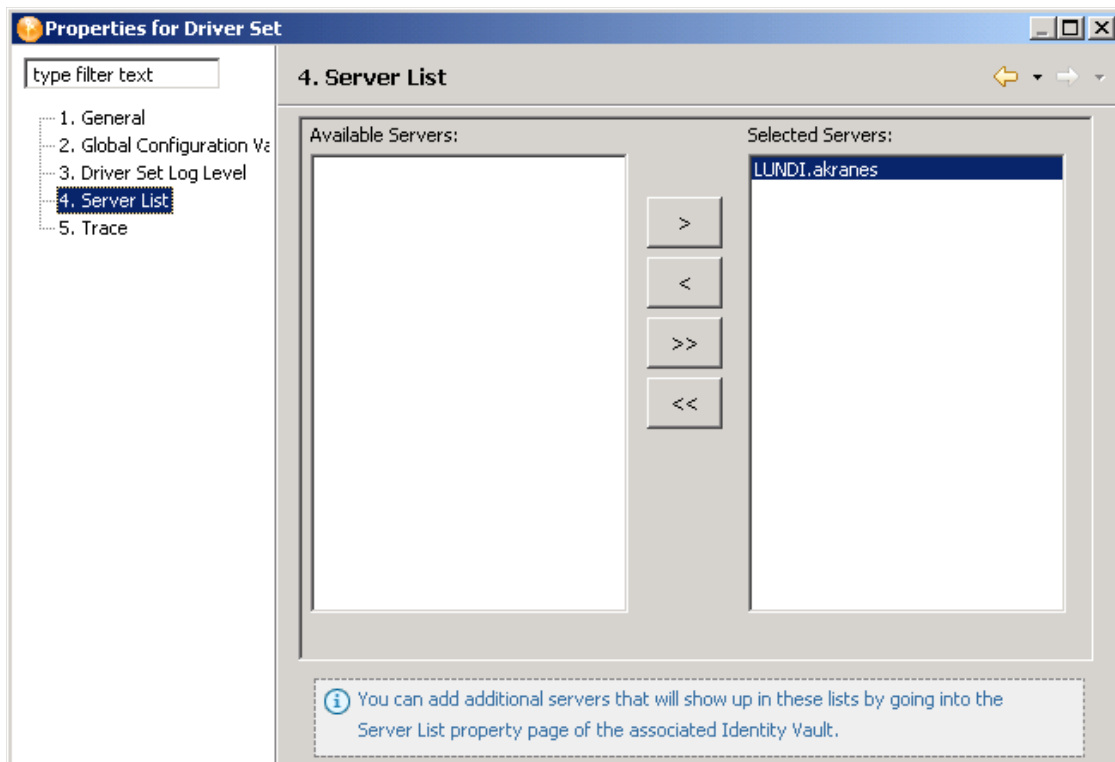
- 5 Save changes by clicking *OK*.

The driver set log contains messages from the engine when it tries to start or stop drivers. To view the log, use iManager. Select the *Status Log* icon above the Identity Vault in the *Identity Manager Overview*.

9.5.4 Driver Set Server List

After adding one or more servers to the Identity Vault, you can view or change the driver set's server association.

Figure 9-7 *Server List*



Select a server in the *Available Servers* list, then use the arrows to move the server to the *Selected Server* list. If a server is not in the Available Server list, you must first add it by editing the Identity Vault properties. See [Section 9.3, "Configuring Identity Vaults,"](#) on page 148.

9.5.5 Driver Set Trace

Although a driver set has nothing to trace, you can add a trace level to your driver. The *Trace* setting specifies the trace level for all drivers in the driver set that use the driver set trace settings.

With the trace level set, DSTRACE displays the Identity Manager (DirXML[®]) events as the engine processes the events. The trace level affects each driver in the driver set. Use the trace level for troubleshooting issues with the drivers when they are deployed. DSTRACE displays the output of the specified trace level.

Table 9-4 *Driver Set Trace Settings*

Setting	Description
<i>Driver trace level</i>	<p>As the driver trace level increases, the amount of information displayed in DSTRACE increases.</p> <p>Trace level 1 shows errors, but not the cause of the errors. If you want to see password synchronization information, set the trace level to 5.</p>
<i>XSL trace level</i>	<p>DSTRACE displays XSL events. Set this trace level only when troubleshooting XSL style sheets. If you do not want to see XSL information, set the level to 0.</p>
<i>Java debug port</i>	<p>Allows developers to attach a Java debugger.</p>
<i>Java trace file</i>	<p>When a value is set in this field, all Java information for the driver is written to file. The value for this field is the path for that file.</p> <p>As long as the file is specified, Java information is written to this file. If you do not need to debug Java, leave this field blank.</p>
<i>Trace file size limit</i>	<p>Sets a limit for the Java trace file. If you set the file size to Unlimited, the file grows in size until no space is available on the disk.</p> <p>NOTE: The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.</p>

The following methods help you capture and save Identity Manager trace information.

NetWare

Use `dstrace.nlm` to display trace messages on the system console or trace messages to a file (`sys:\system\dstrace.log`). `Dstrace.nlm` displays the trace messages to a screen labeled DSTrace Console.

Use the following commands at the server console:

DSTRACE SCREEN ON: Allows trace messages to appear on the DSTRACE Console.

DSTRACE FILE ON: Captures trace messages sent to the DSTRACE Console to the `dstrace.log` file.

DSTRACE FILE OFF: Stops capture trace messages to the log file.

DSTRACE -ALL: Turns off all Trace flags.

DSTRACE +DXML DSTRACE +DVRS: Displays the Identity Manager (DirXML) events.

DSTRACE +TAGS DSTRACE +TIME: Displays the message tags and times tamps.

Windows

Open the Control Panel, select *NDS Services*, then click *DSTRACE.DLM > Start*. A window named NDS Server Trace Utility opens.

To set the filters to capture the DirXML trace information:

- 1 Click *Edit > Options > Clear All*.
- 2 Click the boxes next to *DirXML* and *DirXML Drivers*, then click *OK*.

To save the information to a file:

- 1 Click *File > New*.
A dialog box prompts for a filename.
- 2 Enter a filename with the extension of `.log`.
- 3 To stop capturing information, click *File > Close*.
The file is saved.

UNIX

Use the `ndstrace` command at the console to display the Identity Manager events. The `exit` command quits the trace utility.

Table 9-5 *ndstrace* Commands

Command	Description
<code>Set ndstrace=nodebug</code>	Turns off all trace flags.
<code>Set ndstrace on</code>	Displays trace messages to the console.
<code>Set ndstrace file on</code>	Captures trace message to the <code>ndstrace.log</code> file in the <code>/var/nds</code> directory.
<code>Set ndstrace file off</code>	Stops capturing trace messages to the file.
<code>Set ndstrace=+dxml</code>	Displays the Identity Manager events
<code>Set ndstrace=+dvrs</code>	Displays the Identity Manager driver events.

iMonitor

Use iMonitor to get DSTRACE information from a Web browser.

Table 9-6 *Platforms and Commands for Web Browsers*

Platform	Command
NetWare®	<code>ndsimon.nlm</code>
Windows	<code>ndsimon.dlm</code>
Linux/Solaris/AIX/HP-UX	<code>ndsmonitor</code>

- 1 Access iMonitor from `http://server_ip:8008/nds` (the default port).
- 2 Click *Trace Configuration*.
- 3 Click *Clear All*.
- 4 Click *DirXML* and *DirXML Drivers*.
- 5 Click *Trace On*, then click *Trace History*.
- 6 Click the *Current* document icon to view the live trace.

9.6 Configuring Drivers

A driver provides the connection between an application and the Identity Vault. The driver is the connector that enables data synchronization and sharing between systems. To view or change settings, double-click a driver or driver line in the Modeler.

- ◆ [Section 9.6.1, “Driver General Settings,” on page 159](#)
- ◆ [Section 9.6.2, “Driver Global Configuration Values,” on page 160](#)
- ◆ [Section 9.6.3, “Driver Log Level,” on page 160](#)
- ◆ [Section 9.6.4, “Driver Configuration,” on page 161](#)
- ◆ [Section 9.6.5, “Driver Manifest,” on page 163](#)
- ◆ [Section 9.6.6, “Driver Named Passwords,” on page 163](#)
- ◆ [Section 9.6.7, “Editing or Creating a Driver iManager Icon,” on page 164](#)
- ◆ [Section 9.6.8, “Driver Trace Levels,” on page 165](#)
- ◆ [Section 9.6.9, “Engine Control Values,” on page 167](#)

9.6.1 Driver General Settings

Figure 9-8 The Driver's General Property Page

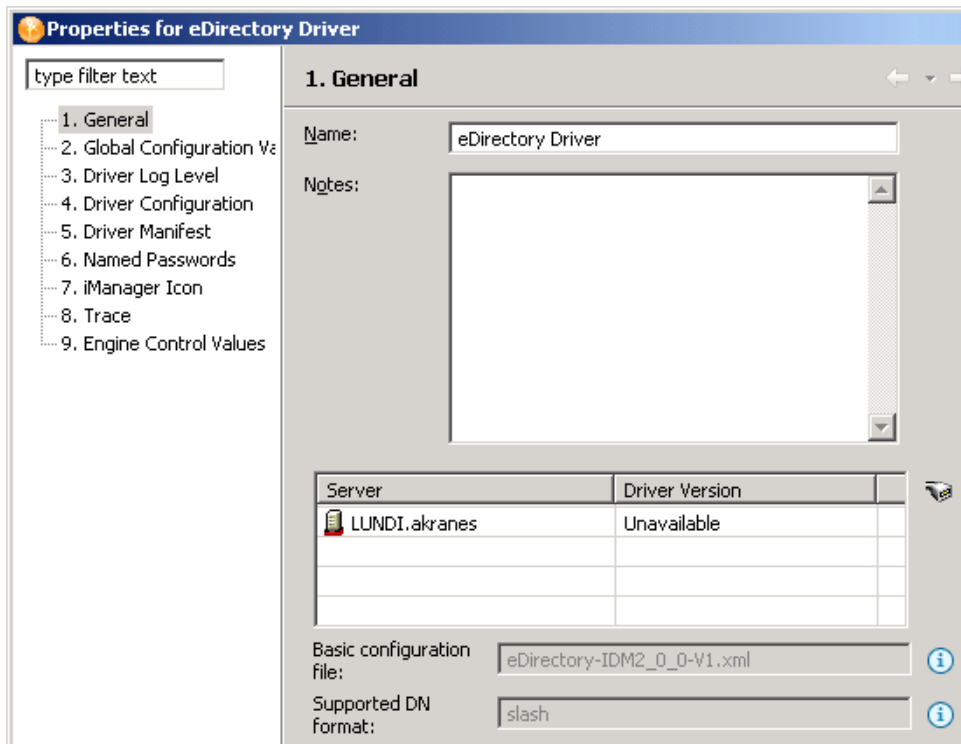


Table 9-7 The Driver's General Settings

Setting	Description
<i>Name</i>	Displays the driver name, which you can change.
<i>Notes</i>	Enables you to type notes about your driver implementation.
<i>Basic configuration file</i>	<p>Displays the configuration filename that this driver uses. Contains the filename of the configuration file that was used during import.</p> <p>To view the path to this file, click the information icon next to the filename. You might want to view the file to find out version information.</p> <p>If you haven't yet run the import wizard, this field is set to <i>None</i>.</p>
<i>Supported DN format</i>	<p>Displays the format (for example, LDAP) that is supported for each driver. This DN information is important for policy building and simulation.</p> <p>For additional details, click the information icon next to the format field.</p>

The driver property page is dynamic. Labels and descriptions are dynamically read from the driver configuration file for the Authentication ID and connection information.

9.6.2 Driver Global Configuration Values

Global configuration values (GCVs) are settings that are similar to driver parameters. Global configuration values can be specified for an individual driver as well as a driver set. If a driver does not have a GCV, the driver inherits the value for that GCV from the driver set.

GCVs allow you to specify settings for Identity Manager features such as password synchronization and driver heartbeat, as well as settings that are specific to the function of an individual driver configuration. Some GCVs are provided with the drivers, but you can also add your own. You can refer to these values in a policy to help you customize your driver configuration.

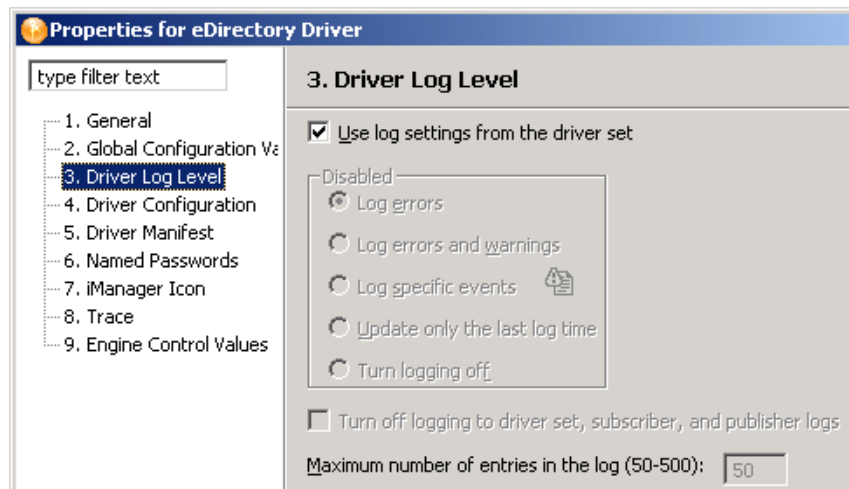
To edit the driver set's GCV settings, double-click the Driver object. From the Global Configuration Values page, you can add, edit, remove, or edit the XML for GCVs.

To view or change the driver's GCV settings, double-click the driver. From the Global Configuration Values page, you can add, edit, or remove values, or edit the XML file for the driver.

9.6.3 Driver Log Level

The Driver Log Level options enable you to view high-level information. For lower-level information, use the Trace option. See [Section 9.6.8, "Driver Trace Levels," on page 165](#).

Figure 9-9 Settings for the Driver Log Level



By default, logging inherits the setting from the driver set. To change the default:

- 1 Double-click the driver.
- 2 Select *Driver Log Level*.
- 3 Select a logging option.

The option that you select determines which information is available in the log.

To configure the audit instrumentation, select *Log specific events*, click the event selector button, select events, then click *OK*.

- 4 Specify the number of entries in the log.

The default is 50 entries (lines) in the log. If you want a longer history, increase the number.

- 5 Save changes by clicking *OK*.

The driver log contains messages from the driver. The messages are related to operations that the driver performed or tried to perform. To view the log, use iManager. Select the log icon on the Driver object in the Identity Manager Overview.

9.6.4 Driver Configuration

- ◆ [“Driver Module” on page 161](#)
- ◆ [“Authentication” on page 161](#)
- ◆ [“Startup Option” on page 163](#)
- ◆ [“Driver Parameters” on page 163](#)

Driver Module

Table 9-8 *Driver Module Settings*

Setting	Description
<i>Java: Name of the Java class</i>	Specify the name of the Java class that will be instantiated for the shim component of the driver. This class can be located in the <code>classes</code> directory as a class file, or in the <code>lib</code> directory as a <code>.jar</code> file.
<i>Native: Name of the DLL</i>	Specify the name of the <code>.dll</code> file that will be instantiated for the application shim component of the driver.
<i>Connect to Remote Loader</i>	Select this option if you want to connect the driver to the Metadirectory engine that uses the Remote Loader.
<i>Driver object password: Set Password</i>	Set a password for the Driver object. If you are using the Remote Loader, you must enter a password on this page or the remote driver cannot run. The Remote Loader uses this password to authenticate itself to the remote driver.
<i>Remote Loader client configuration for documentation: Include in documentation</i>	Enables you to document your Remote Loader configuration for the driver. From the drop-down list, select a name that you specified on the driver's documentation property page. To use this option, see Section 9.6.9, “Engine Control Values,” on page 167 .

Authentication

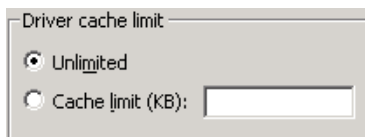
Table 9-9 *Authentication Settings*

Setting	Description
<i>Authentication information for server</i>	The server that the driver is associated with.

Setting	Description
<i>Authentication ID</i>	Enter the application user ID. This ID is used to pass Identity Vault subscription information to the application. If you have enabled SSL/TLS for eDirectory drivers, this option is dimmed.
<i>Remote tree address and port</i>	Enter the address or name and port of the server that the application shim should communicate with.
<i>Set Password</i>	Enables you to set or change an application password (for example, Active Directory).
<i>Remove Password</i>	Deletes the password to the application.
<i>Host name</i>	Specifies the address or name of the machine where the Remote Loader runs. For example, enter <code>hostname=192.168.0.1</code> . If you don't enter this communication parameter, this value defaults to <i>localhost</i> .
<i>Port</i>	Specifies the port that the Remote Loader uses to accept connections from the remote interface shim. For example, enter <code>port=8090</code> . If you don't enter this communication parameter, this value defaults to 8090.
<i>KMO</i>	Specifies the Key Name of the Key Material Object containing the keys and certificate used for SSL. For example, enter <code>kmo=remote driver cert</code> . If you don't enter this communication parameter, no value is stored for this parameter. SSL won't be available.
<i>Other parameters</i>	Provides reference information. It is included when you document your entire project.

Driver Cache Limit

Figure 9-10 Options for the Driver Cache



The driver cache is a file that holds Identity Vault events until a driver can process them.

This file can become very large in the following situations:

- ◆ Events occur at a steady rate that is faster than Identity Manager can process them over a long period of time.
- ◆ The driver is shut down for long period of time but is not disabled.

By default, the driver cache (file) size is limited only by available disk space. This is the recommended setting.

The only reason to set some other limit is to protect against accidentally filling up the disk. The number that you use depends on the difference between projected amount of available disk space without anything in the cache and the amount of free disk space that you want to ensure will always be left available, divided by the number of drivers on the server.

The primary reason that the cache file becomes very large is if the driver is left not running over a long period of time. In this case, the recommendation is to disable the driver rather than set a cache limit. After the limit is reached, all the cached events are discarded.

Startup Option

Table 9-10 *Startup Settings*

Setting	Description
<i>Auto start</i>	The driver starts automatically when the Metadirectory engine loads.
<i>Manual</i>	You must start the driver manually from the driver state location.
<i>Disabled</i>	Disables the driver.
<i>Do not automatically synchronize the driver</i>	If you don't select this option, a driver that has been deployed but disabled resynchronizes on startup. If you select this option, a driver that has been deployed but disabled does not resynchronize.

Driver Parameters

From this tab, you can enter common driver options, Subscriber and Publisher channel options, as well as edit XML. Because the options here differ between drivers, refer to the [Identity Manager Drivers \(http://www.novell.com/documentation/idm35/index.html\)](http://www.novell.com/documentation/idm35/index.html) for configuration information.

9.6.5 Driver Manifest

The driver manifest is like a resume for the driver. The driver manifest states what the driver supports, and includes a few configuration settings. The driver developer should provide the driver manifest. Usually a network administrator does not need to edit the driver manifest.

For more information, see the developer documentation for Identity Manager drivers.

9.6.6 Driver Named Passwords

You can store multiple passwords securely for each of your drivers. This functionality is referred to as Named Passwords because each different password is accessed by a key, or name.

Named Passwords are useful because you can refer to the password by name in a driver policy. When the driver needs to use the password, it requests the password from the Metadirectory engine. This method lets you avoid revealing the actual password in the code for a driver policy.

The following example shows how a Named Password can be referenced in a driver policy on the Subscriber channel in XSLT:

```
<xsl:value-of  
select="query:getNamedPassword($srcQueryProcessor,'mynamedpassword
```

```
' ) "  
xmlns:query="http://www.novell.com/java/  
com.novell.nds.dirxml.driver.XdsQueryProcessor/>
```

You can store and retrieve Named Passwords for any driver without making changes to the driver shim.

As a security measure, in addition to using Named Passwords, you should control access to all Identity Manager objects in eDirectory.

NOTE: A driver developer can also customize a driver to use Named Passwords in other ways, such as retrieving Named Passwords when the driver starts up, instead of requesting them from the Metadirectory engine each time they are needed.

For example, the Identity Manager Driver for Lotus Notes* driver has been customized to support additional ways of using Named Passwords, and examples of those methods are included in the sample driver configurations. For more information, see the [Identity Manager driver guides \(http://www.novell.com/documentation/idm35/index.html\)](http://www.novell.com/documentation/idm35/index.html).

9.6.7 Editing or Creating a Driver iManager Icon

You can see and edit the iManager icons that each driver uses. This is important because iManager renders driver icons in a particular way. However, those icons don't appear anywhere in Designer. Conversely, Designer's application icons don't appear anywhere in iManager's user interface.

To help bridge that gap, you can view the iManager icon to be used in Designer:

Figure 9-11 The Option to View the iManager Icon



- 1 Right-click a driver (for example, eDirectory) in the Modeler, then select *Properties*.
- 2 Select *iManager Icon*.

Designer displays an icon. It is associated with the driver in Designer, unless a different one was imported and stored on the driver.

- 3 To change the icon, click *Browse* or *New*, browse to and select an image file, then click *OK* twice.

The next time that the driver deploys, Designer sets the icon in eDirectory, and iManager uses it.

Also see [Chapter 22, “Editing Icons for Drivers and Applications,”](#) on page 437.

9.6.8 Driver Trace Levels

Through the Modeler, you can add a trace level to your driver or driver set. With the trace level set, DSTRACE displays the Identity Manager events as the engine processes the events. The trace level only affects the driver or driver set it is set on.

Use the trace level for testing or for troubleshooting issues with the drivers when they are deployed. If you run DSTRACE while the drivers are in production, the Identity Manager server might process events slowly.

DSTRACE displays the output of the specified trace level.

The following tables describes each trace field:

Table 9-11 *Driver Trace Settings*

Field	Description
<i>Trace level</i>	As the driver trace level increases, the amount of information displayed in DSTRACE increases. Trace level 1 shows errors, but not the cause of the errors. To see password synchronization information, set the trace level to 5.
<i>Trace level: Use setting from the driver set</i>	If you select this option, all trace levels set at the driver set level take precedence over any driver settings. Otherwise, settings at the driver level are effective.
<i>Trace file</i>	Specify a filename and location where the Identity Manager information is written for the selected driver. When a value is set in this field, all Java information for the driver is written to file. As long as the file is specified, Java information is written to this file. If you do not need to debug Java, leave this field blank.
<i>Trace file: Use setting from the driver set</i>	If you select this option, all trace levels set at the driver set level take precedence over any driver settings. Otherwise, settings at the driver level are effective.
<i>Trace file size limit</i>	Allows you to set a limit for the Java trace file. If you set the file size to Unlimited, the file grows in size until no disk space is available. NOTE: The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.
<i>Trace name</i>	Helps you track trace messages. The name that you specify here appears with the driver trace messages. Use a trace name if the driver name is very long.

NetWare

Use `dstrace.nlm` to display trace messages on the system console or trace messages to a file (`sys:\system\dstrace.log`). `Dstrace.nlm` displays the trace messages to a screen labeled DSTrace Console.

Use the following commands at the server console:

DSTRACE SCREEN ON: Allows trace messages to appear on the DSTRACE Console.

DSTRACE FILE ON: Captures trace messages sent to the DSTRACE Console to the `dstrace.log` file.

DSTRACE FILE OFF: Stops capture trace messages to the log file.

DSTRACE -ALL: Turns off all Trace flags.

DSTRACE +DXML DSTRACE +DVRS: Displays the Identity Manager (DirXML) events.

DSTRACE +TAGS DSTRACE +TIME: Displays the message tags and times tamps.

Windows

Open the Control Panel, select *NDS Services*, then click *DSTRACE.DLM > Start*. A window named NDS Server Trace Utility opens.

To set the filters to capture the DirXML trace information:

- 1 Click *Edit > Options > Clear All*.
- 2 Click the boxes next to *DirXML* and *DirXML Drivers*, then click *OK*.

To save the information to a file:

- 1 Click *File > New*.
A dialog box prompts for a filename.
- 2 Enter a filename with the extension of `.log`.
- 3 To stop capturing information, click *File > Close*.
The file is saved.

UNIX

Use the `ndstrace` command at the console to display the Identity Manager events. The `exit` command quits the trace utility.

Table 9-12 *ndstrace* Commands

Command	Description
<code>Set ndstrace=nodebug</code>	Turns off all trace flags.
<code>Set ndstrace on</code>	Displays trace messages to the console.
<code>Set ndstrace file on</code>	Captures trace message to the <code>ndstrace.log</code> file in the <code>/var/nds</code> directory.

Command	Description
Set ndstrace file off	Stops capturing trace messages to the file.
Set ndstrace=+dxml	Displays the Identity Manager events
Set ndstrace=+dvrs	Displays the Identity Manager driver events.

iMonitor

Use iMonitor to get DSTRACE information from a Web browser.

Table 9-13 Platforms and Commands for Web Browsers

Platform	Command
NetWare®	ndsimon.nlm
Windows	ndsimon.dlm
Linux/Solaris/AIX/HP-UX	ndsimonitor

- 1 Access iMonitor from `http://server_ip:8008/nds` (the default port).
- 2 Click *Trace Configuration*.
- 3 Click *Clear All*.
- 4 Click *DirXML* and *DirXML Drivers*.
- 5 Click *Trace On*, then click *Trace History*.
- 6 Click the *Current* document icon to view the live trace.

9.6.9 Engine Control Values

The engine control values enable you to change certain default behaviors of the Metadirectory engine can be changed. You can access the values only if a server is associated with the Driver Set object.

In Designer:

- 1 In the Modeler, right-click the driver line.
- 2 Select *Properties > Engine Control Values*.
- 3 Click the tooltip icon to the right of the *Engine Controls for Server* field. If a server is associated with the Identity Vault, and if you are authenticated, the engine control values display in the large pane.

Table 9-14 Engine Control Values

Option	Description
<i>Subscriber channel retry interval in seconds</i>	The Subscriber channel retry interval controls how frequently the Metadirectory engine retries the processing of a cached transaction after the application shim's Subscriber object returns a retry status.

Option	Description
<i>Qualified form for DN-syntax attribute values</i>	The qualified specification for DN-syntax attribute values controls whether values for DN-syntax attribute values are presented in unqualified slash form or qualified slash form. A True setting means the values are presented in qualified form.
<i>Qualified form from rename events</i>	The qualified form for rename events controls whether the new-name portion of rename events coming from the Identity Vault is presented to the Subscriber channel with type qualifiers. For example, CN=. A True setting means the names are presented in qualified form.
<i>Maximum eDirectory replication wait time in seconds</i>	The maximum eDirectory™ replication wait time controls the maximum time that the Metadirectory engine waits for a particular change to replicate between the local replica and a remote replica. This only affects operations where the Metadirectory engine is required to contact a remote eDirectory server in the same tree to perform an operation and might need to wait until some change has replicated to or from the remote server before the operation can be completed (for example, object moves when the Identity Manager server does not hold the master replica of the moved object; file system rights operations for Users created from a template.)
<i>Use non-compliant backwards-compatible mode for XSLT</i>	This control sets the XSLT processor used by the Metadirectory engine to a backward-compatible mode. The backward-compatible mode causes the XSLT processor to use one or more behaviors that are not XPath 1.0 and XSLT 1.0 standards-compliant. This is done in the interest of backward-compatibility with existing DirXML® style sheets that depend on the non-standard behaviors. For example, the behavior of the XPath “!=” operator when one operand is a node set and the other operand is other than a node set is incorrect in DirXML releases up to and including Identity Manager 2.0. This behavior has been corrected; however, the corrected behavior is disabled by default through this control in favor of backward-compatibility with existing DirXML style sheets.
<i>Maximum application objects to migrate at once</i>	This control is used to limit the number of application objects that the Metadirectory engine requests from an application during a single query that is performed as part of a Migrate Objects from Application operation. If java.lang.OutOfMemoryError errors are encountered during a Migrate from Application operation, this number should be set lower than the default. The default is 50.
<i>Set creatorsName on objects created in Identity Vault</i>	NOTE: This control does not limit the number of application objects that can be migrated; it merely limits the batch size. This control is used by the Identity Manager engine to determine if the creatorsName attribute should be set to the DN of this driver on all objects created in the Identity Vault by this driver. Setting the creatorsName attribute allows for easily identifying objects created by this driver, but also carries a performance penalty. If not set, the creatorsName attribute defaults to the DN of the NCP™ Server object that is hosting the driver.

Option	Description
<i>Write pending associations</i>	<p>This control determines whether the Identity Manager engine writes a pending association on an object during Subscriber channel processing.</p> <p>Writing a pending association confers little or no benefit but does incur a performance penalty. Nevertheless, the option exists to turn it on for backward compatibility.</p>
<i>Use password event values</i>	<p>This control determines the source of the value reported for the nspmDistributionPassword attribute for Subscriber channel Add and Modify events.</p> <p>Setting the control to False means that the current value of the nspmDistributionPassword is obtained and reported as the value of the attribute event. This means that only the current password value is available. This is the default behavior.</p> <p>Setting the control to True means that the value recorded with the eDirectory event is decrypted and is reported as the value of the attribute event. This means that both the old password value (if it exists) and the replacement password value at the time of the event are available. This is useful for synchronizing passwords to certain applications that require the old password to enable setting a new password.</p>
<i>Enable password synchronization status reporting</i>	<p>This control determines whether the Identity Manager engine reports the status of Subscriber channel password change events.</p> <p>Reporting the status of Subscriber channel password change events allows applications such as the Identity Manager User Application to monitor the synchronization progress of a password change that should be synchronized to the connected application.</p>

9.7 Configuring Jobs

Designer has a job scheduling utility to schedule events. Through this utility, the system can be set to disable an account on a specific day, or to initiate a workflow to request an extension for a person's access to a corporate resource. Designer's job scheduler contains the same functionality as the job scheduler found in iManager. For information on creating jobs, see [Section 18.1, "Creating a Job," on page 361](#).

In the Outline view, right-click the Job icon, then select Properties.

- ◆ [Section 9.7.1, "General," on page 169](#)
- ◆ [Section 9.7.2, "Trace," on page 170](#)

9.7.1 General

You have one selection under the General heading: *Policy Name*. You can change the job's name by modifying the name that appears in the *Policy Name* entry. Then click *OK*.

9.7.2 Trace

Through the Modeler, you can add a trace level to your jobs. With the trace level set, DSTRACE displays the Identity Manager events as the engine processes the events. The trace level only affects the driver it is set on.

Use the trace level for testing or for troubleshooting issues with the designated job. If you run DSTRACE while the drivers are in production, the Identity Manager server might process events slowly.

DSTRACE displays the output of the specified trace level.

The following table describes each trace field:

Table 9-15 Job Trace Settings

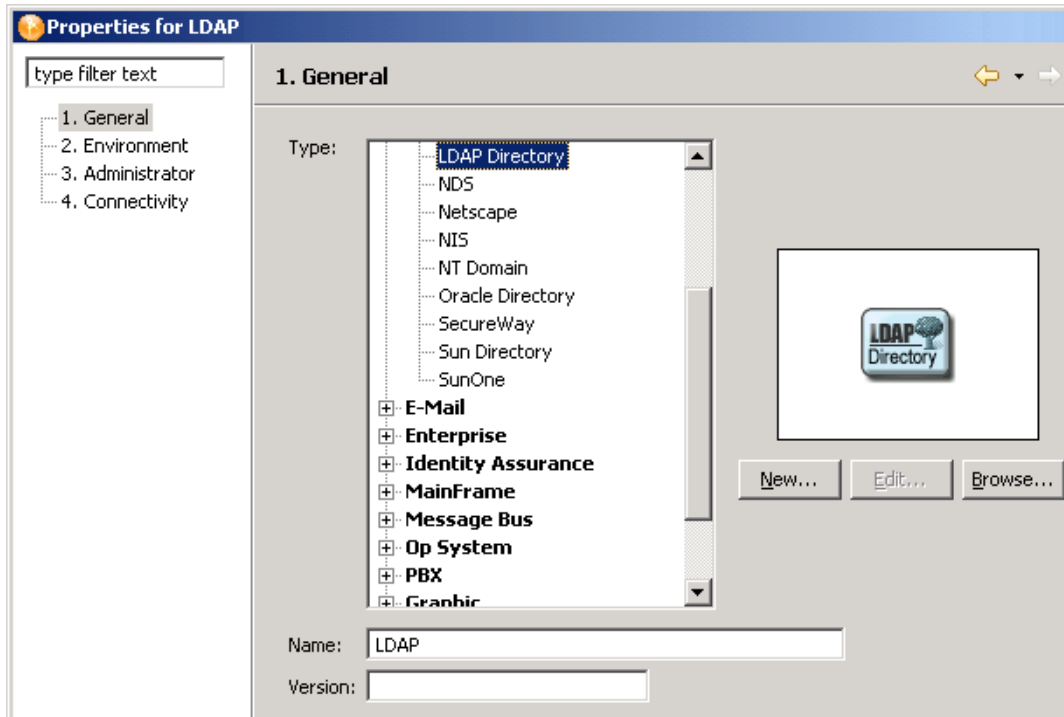
Field	Description
<i>Trace level</i>	<p>As the job trace level increases, the amount of information displayed in DSTRACE increases.</p> <p>Trace level 1 shows errors, but not the cause of the errors. To see password synchronization information, set the trace level to 5.</p>
<i>Trace file</i>	<p>Specify a filename and location where the Identity Manager information is written for the selected driver. When a value is set in this field, all Java information for the job is written to file.</p> <p>As long as the file is specified, Java information is written to this file. If you do not need to debug Java, leave this field blank.</p>
<i>Trace file size limit</i>	<p>Allows you to set a limit for the Java trace file. If you set the file size to Unlimited, the file grows in size until no disk space is available.</p> <p>NOTE: The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.</p>
<i>Trace name</i>	<p>Helps you track job trace messages. The name that you specify here appears with the job trace messages.</p>

For more information on DSTRACE, see [Section 9.6.8, “Driver Trace Levels,”](#) on page 165.

9.8 Configuring Application Properties

To view or change an application’s settings, double-click the application (for example, LDAP Directory) in the Modeler.

Figure 9-12 The General Properties Page for an Application



- ◆ Section 9.8.1, “General,” on page 171
- ◆ Section 9.8.2, “Environment,” on page 172
- ◆ Section 9.8.3, “Administrator,” on page 172
- ◆ Section 9.8.4, “Connectivity,” on page 172
- ◆ Section 9.8.5, “AD Domain,” on page 174

9.8.1 General

Table 9-16 An Application’s General Settings

Field	Description
Type	Changes the type of application your driver connects to. For example, if you configure a JDBC driver to connect to a MySQL database, but then need to change to an Oracle database, you can scroll to <i>Database</i> , select <i>Oracle</i> , then click <i>Apply</i> .
New	.Enables you to edit a driver’s icon. See Section 9.6.7, “Editing or Creating a Driver iManager Icon,” on page 164.
Edit	Enables you to use the Icon Editor to customize the application’s icon. Is available after you click <i>New</i> , edit an icon, and click <i>Update</i> .
Browse	.Enables you to navigate to and select an image file.
Name	.Enabled you to customize the application’s name or label.

Field	Description
<i>Version</i>	.Enables you to document the application's version.

9.8.2 Environment

You can enter notes about the application's environment.

9.8.3 Administrator

The *Administrator* option is divided into three sections. Entering information in these sections is optional.

- ◆ **Personal Information:** Enables you to enter information specific to the Identity Vault, such as Name, Title, Department, and Location,
- ◆ **Contact Information:** Enables you to enter information such as Email, Phone, Cell phone, Pager, and Fax.
- ◆ **Notes:** Allows you to type any reminders you might need for future reference.

9.8.4 Connectivity

- ◆ [“LDAP” on page 172](#)
- ◆ [“VNC” on page 172](#)
- ◆ [“Configuring a Remote Connection” on page 173](#)
- ◆ [“Customizing the Viewer” on page 174](#)

LDAP

You can configure some applications (for example, Active Directory, eDirectory, and LDAP) for an LDAP connection. If the application doesn't support an LDAP connection, the LDAP tab doesn't display.

Host: The server's IP address or DN (in LDAP format).

User: The user's name.

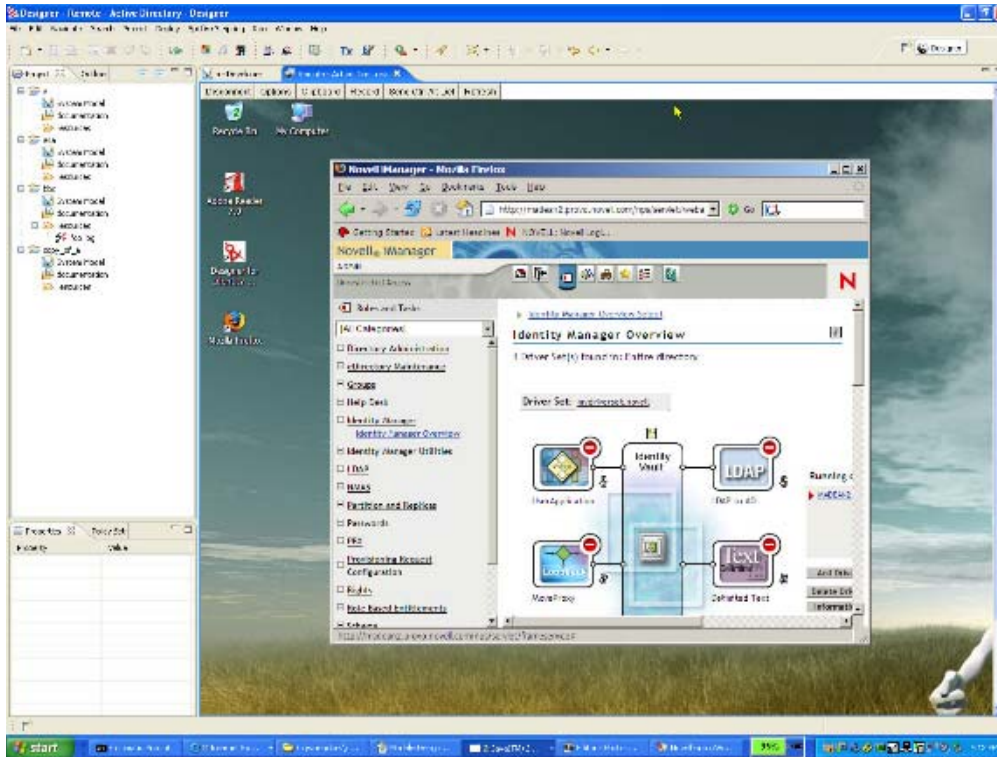
Password: The user's password.

VNC

From within Designer, you can view the desktop of the machine that is running your applications, and remotely control that desktop by interacting with it. This feature enables you to administer users or your applications with the native tools of that system, from one location.

This functionality is hosted in an embedded editor inside Designer. You can have multiple remote control sessions with different systems, all open at the same time.

Figure 9-13 A Remote Desktop

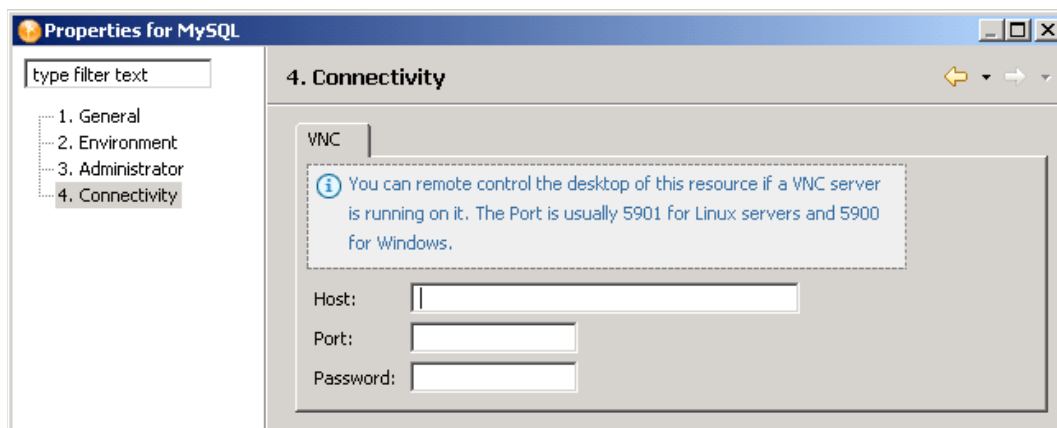


Configuring a Remote Connection

To remotely control a desktop, the machine that is running your application needs to have a VNC (virtual network computing) server installed and running. You can usually download a VNC server for free from the Internet.

You can easily configure any system or design element in Designer for this feature by editing any application or design element:

- 1 Right-click an application or design element.
- 2 Select *Properties > Connectivity*.



- 3 On the VNC tab, type the authentication information.

Host: The DN (for example, server33.houston.company.com) of the server where the VNC is running.

Port: Typically 5901 for Linux servers or 5900 for Windows.

Password: The password to the VNC server.

4 Click *OK*.

Customizing the Viewer

A toolbar at the top of the desktop viewer enables you configure the following:

- ◆ Encoding type (RAW, RRE, CoRRE, Hextile, Zlib, Tight).
The default is Tight.
- ◆ Compression level.
- ◆ JPEG Image Quality (0 - 9).
The default is 6.
- ◆ Cursor shape updates.
The default is Enable.
- ◆ Use CopyRect.
The default is Yes.
- ◆ Mouse buttons 2 and 3.
The default is Normal.
- ◆ View only.
The default is No, so that you can interact with the desktop.
- ◆ Clipboard.
- ◆ Record session and save to file.
- ◆ Send Ctrl+Alt+Delete.
- ◆ Refresh.

For more information, see TightVNC documentation.

9.8.5 AD Domain

You can capture information about an Active Directory application. This information is useful if you want Document Generator to include this information when you document the project.

If you provided information in the **LDAP** settings, Designer populates the AD Domain fields.

9.9 Configuring Policies

- ◆ [Section 9.9.1, “Editing a Policy Name,” on page 175](#)
- ◆ [Section 9.9.2, “Viewing References,” on page 175](#)
- ◆ [Section 9.9.3, “Editing a Credential Provisioning Policy,” on page 175](#)

9.9.1 Editing a Policy Name

To edit a policy name:

- 1 In the Outline view, right-click a policy or rule.
- 2 Select *Properties*.
The *General* setting displays by default.
- 3 Edit the name in the *Policy Name* field, then click *OK*.

9.9.2 Viewing References

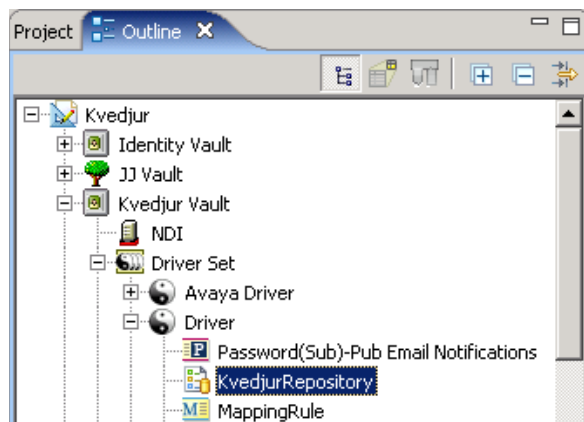
The References page lists policy sets and policies that reference the policy listed in the General page. To view the references to this policy:

- 1 In the Outline view, right-click a policy or rule.
- 2 Select *Properties* > *References*.

Linkage is how the policies reference each other. In Identity Manager versions earlier than 3.5, linkage determined the order that policies were executed. To change the linkage, use the Policy Builder.

9.9.3 Editing a Credential Provisioning Policy

- 1 In the Outline view, right-click a credential provisioning policy (for example, a repository).



- 2 Select *Properties*.
- 3 Edit the policy name, then click *OK*.

9.10 Adding Prompts to a Driver Configuration File

Several node types are defined for driver configuration files. These extensions were made to support the following:

- ◆ Prompting once for a value that is used repeatedly throughout a single driver configuration file.

- ◆ Prompting once for a value that is used across multiple driver configuration files, as part of the Import Drivers wizard.
- ◆ Allowing the user to select a value from a drop-down list of values.
- ◆ Global modification of the driver configuration file as per a contained XSL stylesheet.
- ◆ Built-in variables that can be referenced without declaring them to access information about the driver and its environment (a tree name, driver set name, driver set DN, server name, server DN, driver name and driver DN).
- ◆ The ability to “layer” prompts. It is possible to ask the user multiple sets of questions, with the 2nd and later sets being controlled by the user's responses to prior sets. For more information, refer to the FlexiblePrompts document.

The primary new node types are variable-decl, variable-ref, and xsl-modify.

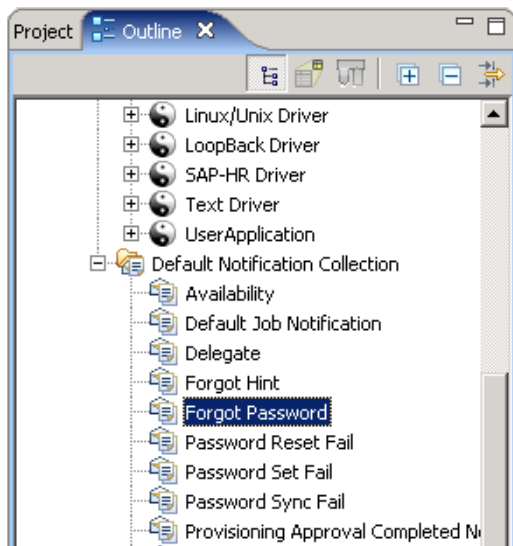
Table 9-17 *New Node Types*

New Node Type	Description
variable-decl	Allows you to define driver configuration variables that are prompted for (optionally) and replaced into a driver configuration file during its import. Multiple variable-decl blocks can be used to define a “layered” set of prompts. Refer to the FlexiblePrompts document for more information.
variable-ref	Used to reference a variable defined in a variable-decl within your driver configuration files.
xsl-modify	Used to globally modify the driver configuration file after all variables (and prompting) have been resolved. The contents of this Node are extracted and used as an XSL stylesheet that is applied to the patched driver configuration file.

For information on adding prompts to a sample configuration file, see “Editing Driver Configuration Files,” an appendix in the [Novell Identity Manager Administration Guide \(http://www.novell.com/documentation/idm35/index.html\)](http://www.novell.com/documentation/idm35/index.html).

9.11 Editing a Notification Template Name

- 1 In the Outline view, expand *Default Notification Collection*.



- 2 Right-click a notification template (for example, Forgot Password), then select *Properties*.
- 3 Edit the name, then click *OK*.

For additional information, see [Chapter 14, “Setting Up E-Mail Notification Templates,”](#) on [page 249](#).

9.12 Synchronizing Passwords

To view or edit password synchronization, use the Dataflow editor. See [Section 12.2.1, “Filtering Views,”](#) on [page 228](#) and [“Synchronizing Passwords”](#) on [page 228](#).

Managing Versions

10

Your environment might have versions of Identity Manager earlier than version 3.5, or you might have a mixture of different versions of Identity Manager.

Before Designer 2.1, if you configured and wrote policies for an earlier version of Identity Manager in your environment, you might have encountered the following issues:

- ◆ You could easily build a solution that would not deploy.
- ◆ You did not know which features worked in one environment versus another environment.

To solve these issues, Designer 2.1 tracks versions of the following objects:

- ◆ Identity Manager engines
- ◆ Identity Vaults (trees)
- ◆ Drivers

As you use Designer 2.1, you see only the UI of features that apply to the version that you are working on. Project Checker and Deploy ensure that what you have configured is supported in the target environment.

- ◆ [Section 10.1, “Key Differences in Identity Manager Versions,” on page 180](#)
- ◆ [Section 10.2, “Changing the Identity Manager Version,” on page 180](#)
- ◆ [Section 10.3, “Tracking Versions of Identity Manager,” on page 181](#)
- ◆ [Section 10.4, “Information in the Driver Configuration Wizard,” on page 183](#)
- ◆ [Section 10.5, “Checking Projects for Version Issues,” on page 185](#)
- ◆ [Section 10.6, “Adjusting the UI Based on the Version Number,” on page 186](#)

10.1 Key Differences in Identity Manager Versions

Table 10-1 Differences in Identity Manager Versions

Identity Manager 3.0	Identity Manager 3.0.1	Identity Manager 3.5
<ul style="list-style-type: none"> ◆ New DirXML[®] Script actions ◆ Entitlements were added 	<ul style="list-style-type: none"> ◆ New DirXML Script actions ◆ New Log Events: <ul style="list-style-type: none"> ◆ Add Value (on add) ◆ Get Named Password ◆ Reset Attributes ◆ Add Value (on modify) ◆ Resource Objects were added: <ul style="list-style-type: none"> ◆ Credential Provisioning Applications ◆ Credential Provisioning Repository Objects ◆ Generic Resource Objects 	<ul style="list-style-type: none"> ◆ New object types were added: <ul style="list-style-type: none"> ◆ ECMAScript Objects ◆ Jobs ◆ Mapping Table Resource Objects ◆ Resource Libraries ◆ New Policy Linking capabilities where a policy can be in multiple lists ◆ Many new DirXML Script actions, conditions, tokens, and verbs ◆ Ability for DirXML Script to nest conditions ◆ Driver-scoped local variables in DirXML Script that let you refer to variable outside of the policy

10.2 Changing the Identity Manager Version

You can both import and deploy to all versions of Identity Manager that shipped since Identity Manager 2.0, up to and including Identity Manager 3.5.1. You can also import from DirXML 1.x environments.

Because versions earlier than Designer 2.0 M5 did not track Identity Manager versions, those earlier projects do not have version information.. When you convert an earlier project, Designer 2.1 defaults the Identity Manager version numbers to 3.0.1. During conversion, Designer informs you that this default is being applied.

You can change this version number by doing either of the following:

- ◆ In the Outline view, right-click the Server object, select *Properties*, then select from the *Identity Manager Version* drop-down list.
- ◆ In the Modeler, select an Identity Vault, click *Window > Preferences > Designer for IDM > Versions*, then select a version from the drop-down list.

You can also find information on upgrades, information on downgrades, and a link to a help topic. This information explains the key differences between versions of Identity Manager.

When you import into a new server (or create a server based on a server that you have browsed to in the directory), the new server inherits the imported version of Identity Manager.

If you do a live update in the server properties page, Designer updates the server to the current version of Identity Manager that is in the target environment.

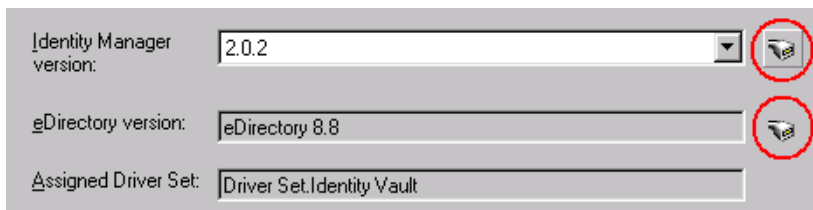
10.3 Tracking Versions of Identity Manager

Designer 2.1 tracks the Identity Manager version. Filtering functionality is based on this version information. When multiple servers are associated to a driver set, Designer calculates an “effective engine version.” This version is the earliest Identity Manager version in the driver set.

If you want to use the latest Identity Manager 3.5.1 features, it is important that all servers belonging to the driver set are upgraded to 3.5.1. This version can be manually upgraded or downgraded from the server properties page.

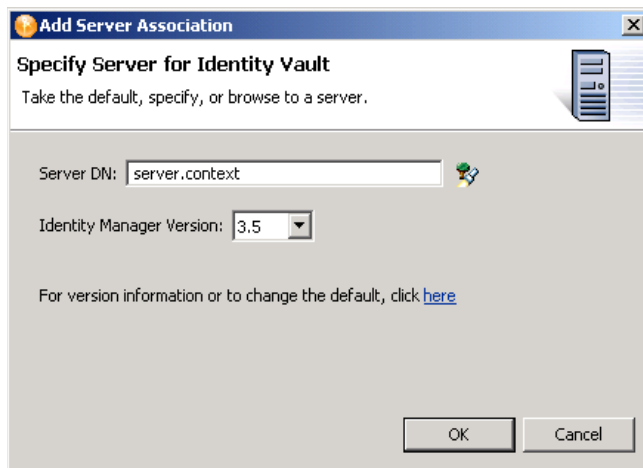
Additionally, live update buttons retrieve current Identity Manager and eDirectory version information on the server properties page.

Figure 10-1 Live Update Buttons



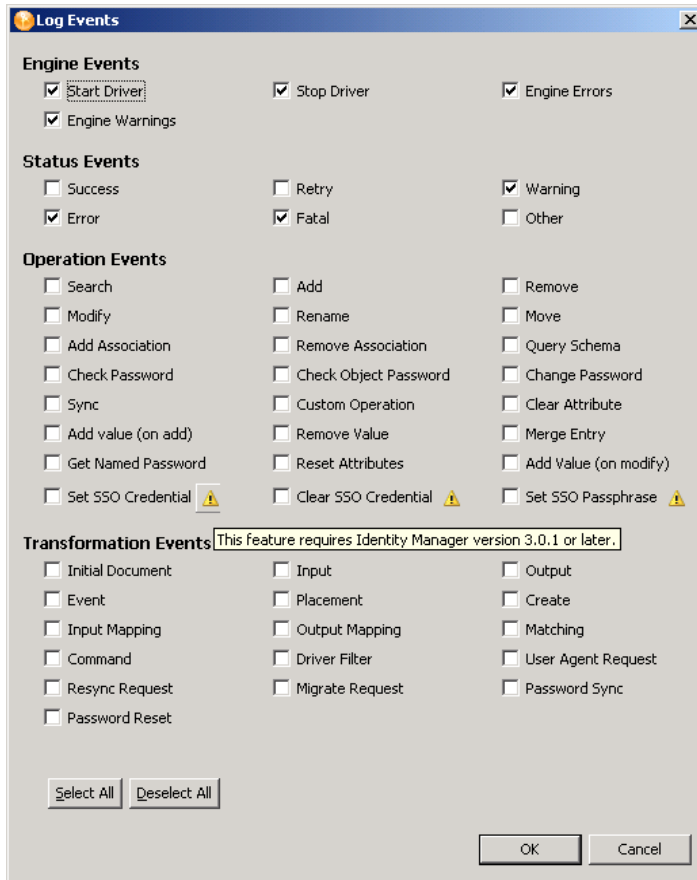
The *Add Server* dialog box allows you to specify version information when an Identity Vault is created.

Figure 10-2 The Add Server Dialog Box



The Driver Set Log Level and Driver Log Level property pages have dynamic version widgets next to any log event that is not supported by your effective Identity Manager version. Designer 2.0 M5 added support for several Identity Manager log events: Set SSO Credential, Clear SSO Credential and Set SSO Passphrase.

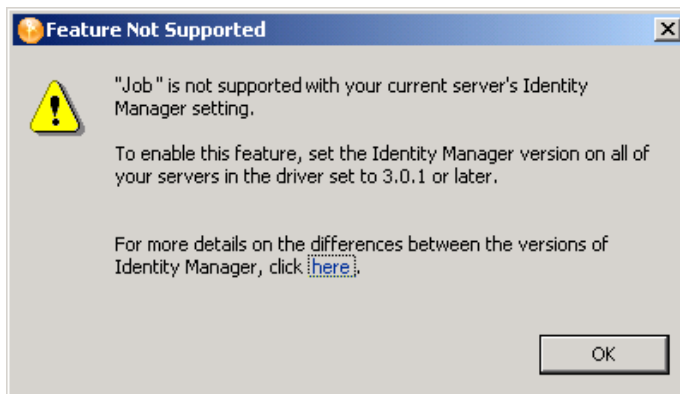
Figure 10-3 Log Events



For more information about these events, see the [Credential Provisioning Policies \(http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy_credprov/data/bookinfo.html\)](http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy_credprov/data/bookinfo.html) documentation.

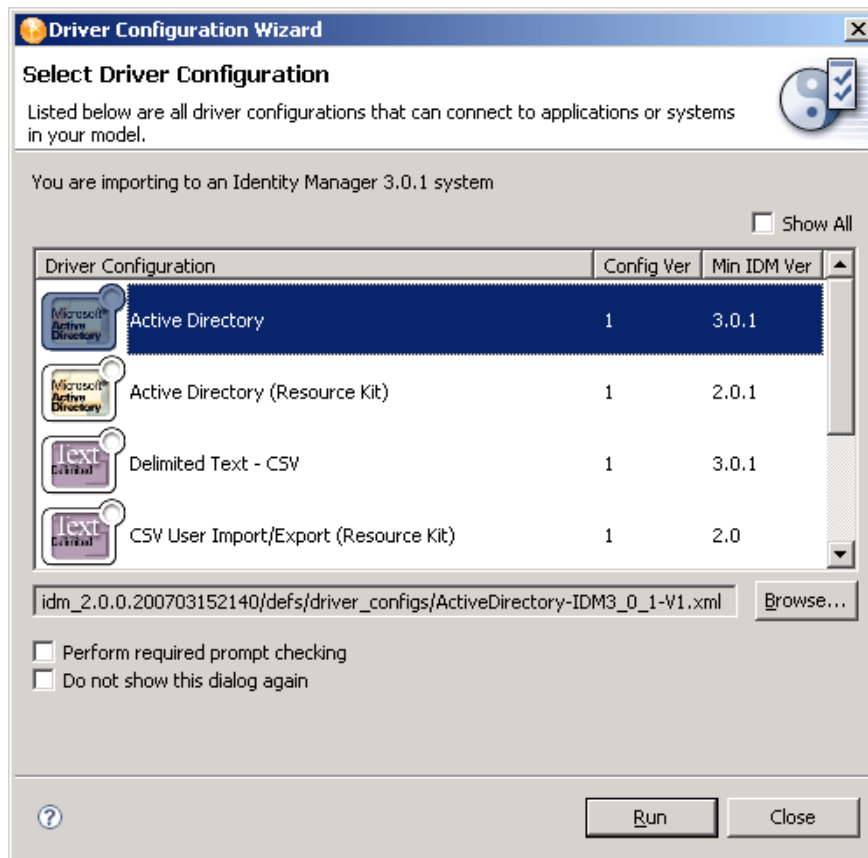
The following figure illustrates an unsupported log event:

Figure 10-4 Message Concerning an Identity Manager Version



10.4 Information in the Driver Configuration Wizard

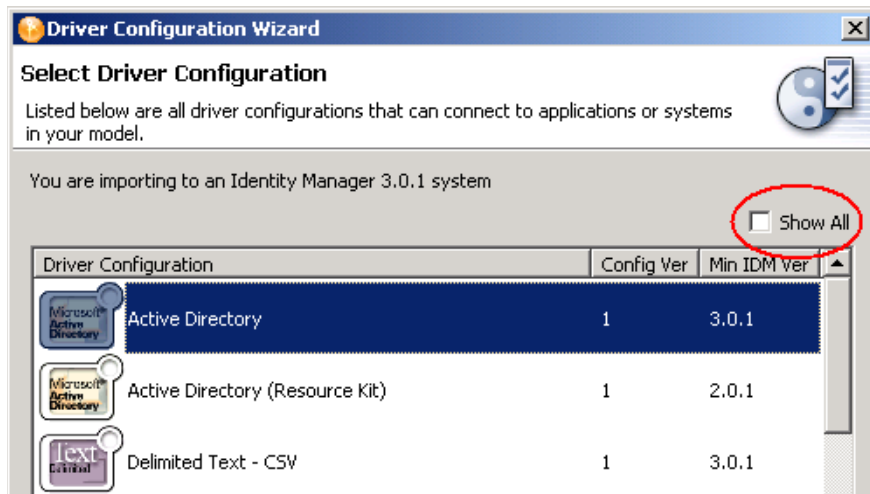
Figure 10-5 The Driver Configuration Wizard



In Designer 2.1, the Driver Configuration Wizard provides more information than previous milestone releases provided:

- ◆ The engine version that you are importing into. This information is taken from the current project. You control the version number.
- ◆ A descriptive name of the driver configuration (no longer just the filename).
- ◆ The version of the configuration as a single (undelimited) version number.
- ◆ The minimum required engine version for this configuration to run.
- ◆ The full filename of the selected list item. This name is below the list. It is displayed there for transparency.
- ◆ A check box that indicates possible unrecommended or incompatible configuration files.

Figure 10-6 A Deselected Show All Check Box



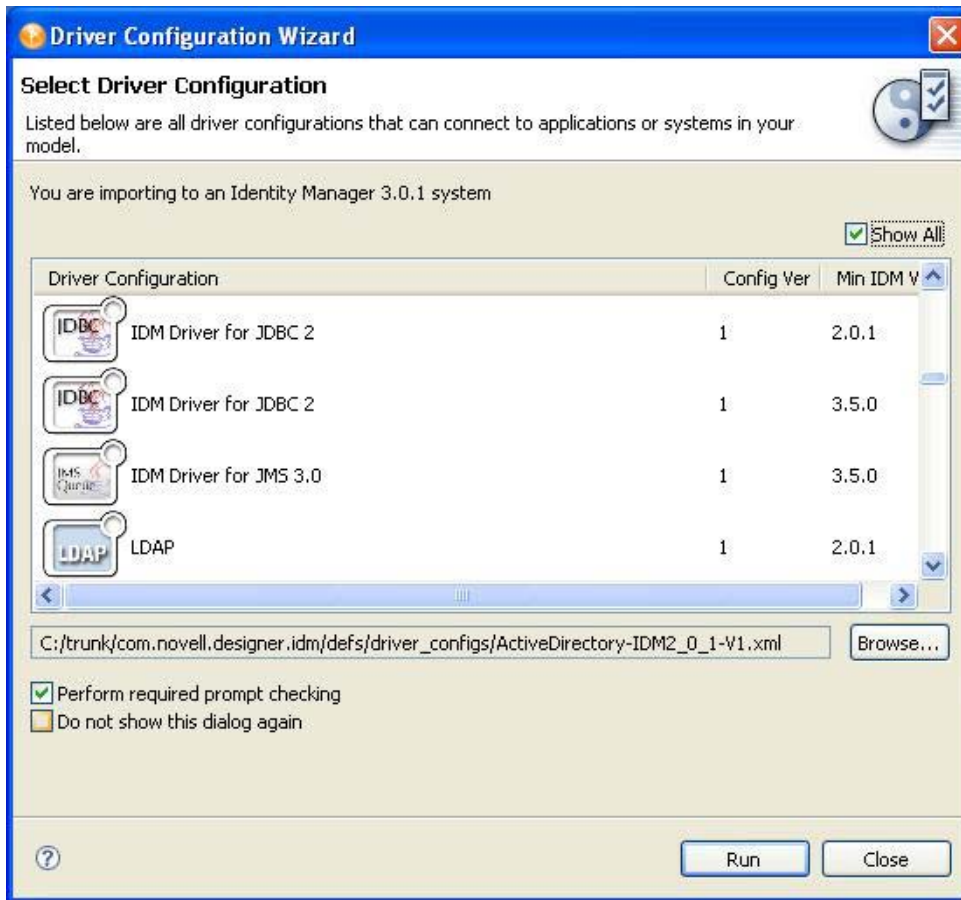
By default, the *Show All* check box is deselected if unrecommended or possibly incompatible configuration files are available. If all available driver configuration files are recommended and guaranteed compatible, the check box is dimmed and selected, demonstrating that all available options are displayed.

A deselected *Show All* check box implies the following:

- ◆ Additional driver configuration files are available but they are not recommended.
- ◆ The additional driver configuration files are probably incompatible with the engine version that you are importing to.

In the following figure, the *Show All* check box is selected.

Figure 10-7 A Selected Check Box in the Wizard



The list now contains many more items than were displayed when the check box was deselected. These new items were previously hidden because the minimum required engine version for them is 3.5. Because the user is importing to 3.0.1, the configuration might be incompatible.

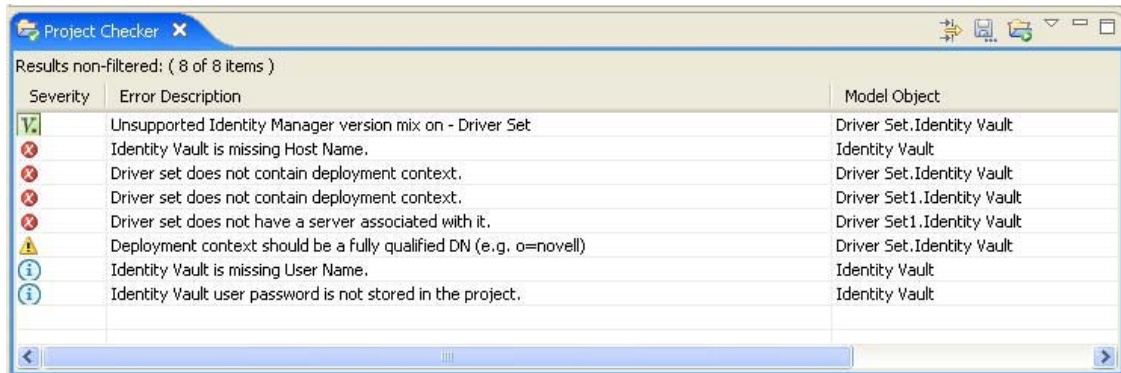
10.5 Checking Projects for Version Issues

A full suite of project checks makes sure that what you have configured makes sense for your target environment and can be successfully deployed. Designer's UI blocks the creation of unsupported objects and hides features based on the version number. Nevertheless, unsupported actions might still occur through a few “back-door” methods, such as copying and pasting, importing, and downgrading your server after you have configured for a newer environment.

In all of these methods, Project Checker catches the problems.

For example, for policy libraries to work, all of the servers on a given driver set need to be at the same Identity Manager 3.5 version. Project Checker catches problems like this where you might have an unsupported mix of servers. In this case, the project check results would look like the following figure:

Figure 10-8 Project Checker



Version problems are sorted to the top and have a version icon. If you double-click the item, you get more details about the problem and how to resolve it.

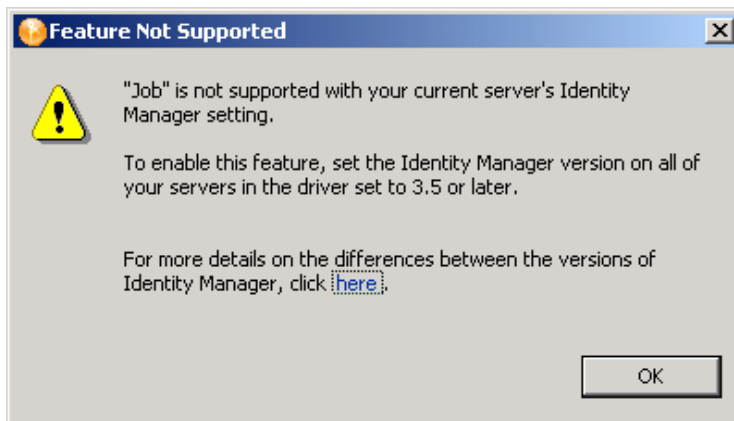
10.6 Adjusting the UI Based on the Version Number

Designer hides or shows and enables or disables capabilities based on the version of Identity Manager that is associated with the Identity Manager engine. For example, if you edit a policy that is associated with a server that uses Identity Manager 3.5, Policy Builder shows you all of the new actions, conditions, verbs, and tokens that ship with that release. This feature lets you try out the next version of Identity Manager before it is even released.

Also, if you set the server to Identity Manager 3.0.1 (or earlier), you get the previous version of Policy Builder that Designer has shipped with in the past.

If you try to create an object that is not supported by your server version, a prompt tells you that this action isn't supported. For example, Identity Manager 3.5 introduces the concept of Jobs, Mapping Tables, and Policy Libraries. If you try to create one of these objects on a 3.0.1 server, you see the following message:

Figure 10-9 Prompt: Feature Not Supported



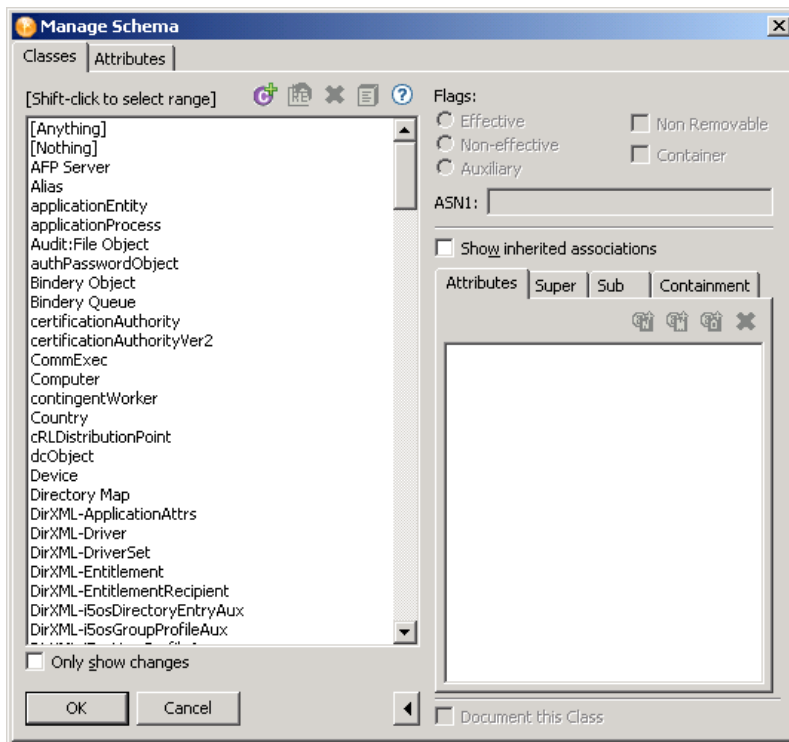
Future milestones of Designer will continue to evolve the UI to better handle version differences.

Managing the Schema

Designer includes a copy of the base eDirectory™ schema, which is stored in the BaseIVSchema.xml file. This file is located in designer\eclipse\plugins\com.novell.designer_x.x.x\defs\schema.

Do not directly modify BaseIVSchema.xml. Instead, use Designer to add the schema information from this file into your project. The Manage Schema tool allows you to change the schema as part of the project. The original BaseIVSchema.xml file is not changed.

Figure 11-1 The Manage Schema Tool



You can add, delete, rename, and modify classes and attributes in the eDirectory schema. You can import the eDirectory schema from the production environment, or use the default schema. After modifying the schema, you can deploy it into the production eDirectory tree.

WARNING: If you do not have a good understanding of how the eDirectory schema works, changing the default schema can cause data corruption. If you modify classes or attributes and then deploy the modified schema into a tree where these classes are in use, one of the following problems can occur:

- ◆ Those objects can become unknown.
 - ◆ eDirectory synchronization errors can occur.
-

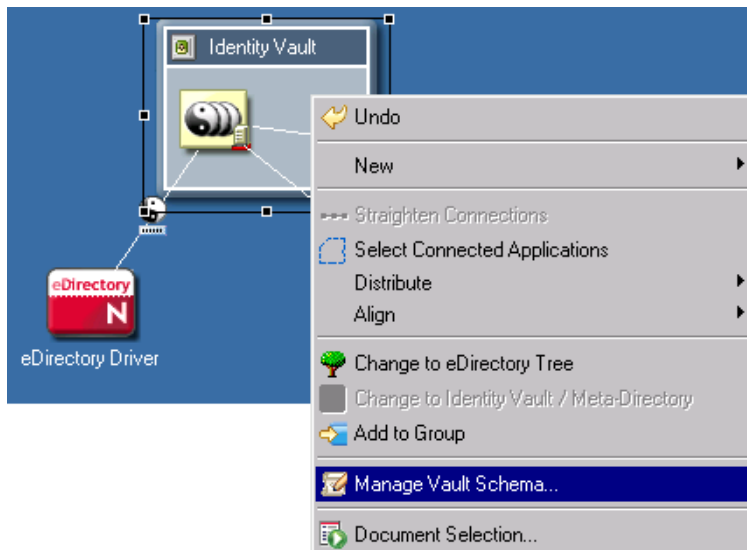
To understand the basics of the schema, see [Managing the Schema \(http://www.novell.com/documentation/edir88/index.html?page=/documentation/edir88/edir88/data/a4a9bz0.html\)](http://www.novell.com/documentation/edir88/index.html?page=/documentation/edir88/edir88/data/a4a9bz0.html) in the online documentation for eDirectory 8.8.

If you subscribe to LogicSource[®], see [Novell[®] LogicSource for eDirectory \(http://support.novell.com/subscriptions/articles/novell_logicsource.html\)](http://support.novell.com/subscriptions/articles/novell_logicsource.html) for additional information. LogicSource is a subscription-based service that Novell provides to its customers.

- ◆ [Section 11.1, “Accessing the Manage Schema Tool,” on page 188](#)
- ◆ [Section 11.2, “Creating Classes and Attributes,” on page 189](#)
- ◆ [Section 11.3, “Modifying the Schema,” on page 201](#)
- ◆ [Section 11.4, “Deploying the Schema into the eDirectory Tree,” on page 204](#)
- ◆ [Section 11.5, “Exporting the Schema to a File,” on page 206](#)
- ◆ [Section 11.6, “Importing the Schema,” on page 209](#)
- ◆ [Section 11.7, “Managing a Copy of an Application Schema,” on page 216](#)
- ◆ [Section 11.8, “Mapping eDirectory to an LDAP Schema,” on page 220](#)

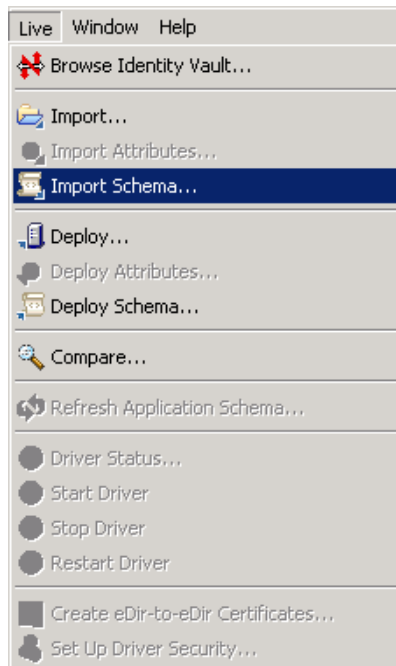
11.1 Accessing the Manage Schema Tool

- 1 Right-click an Identity Vault, then select *Manage Vault Schema*.



You can import the schema by using the *Live* menu:

Figure 11-2 *The Live Option*



If a custom schema in the production environment needs to be tested, you can import the schema into Designer. After you have tested and modified the schema, you can deploy it into the production environment.

11.2 Creating Classes and Attributes

Designer allows you to create eDirectory classes and attributes to fit the needs of your environment. You can test and use the new schema with the Identity Manager drivers in Designer before implementing it in the production environment.

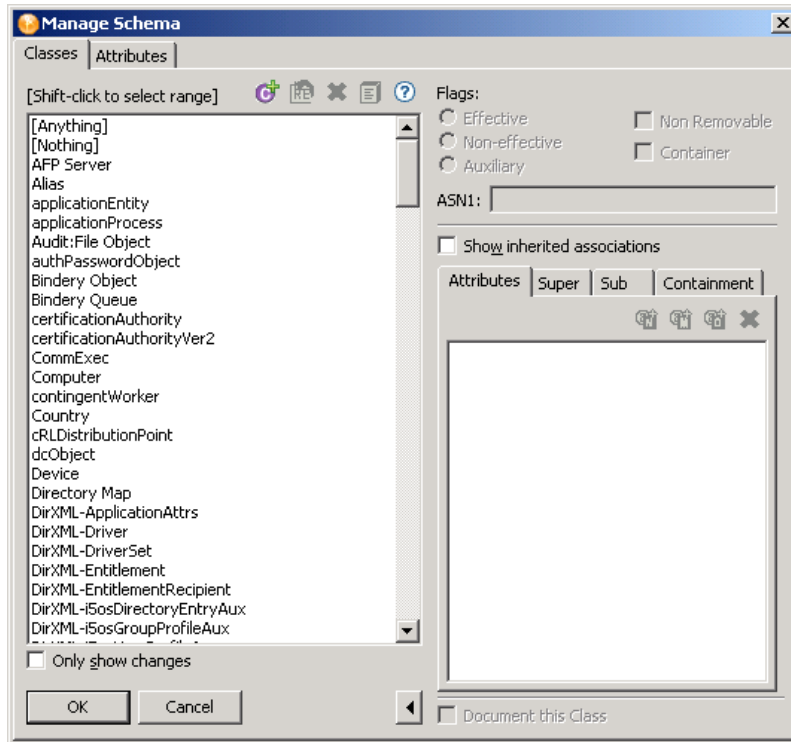
- ◆ [Section 11.2.1, “Creating eDirectory Classes,” on page 189](#)
- ◆ [Section 11.2.2, “Creating eDirectory Attributes,” on page 194](#)

11.2.1 Creating eDirectory Classes

- ◆ [“Adding a Class” on page 190](#)
- ◆ [“Icons and Fields in the Classes Page” on page 193](#)
- ◆ [“Adding a Note” on page 194](#)

Adding a Class


- 1 In the Modeler, right-click the Identity Vault, then select *Manage Vault Schema*.

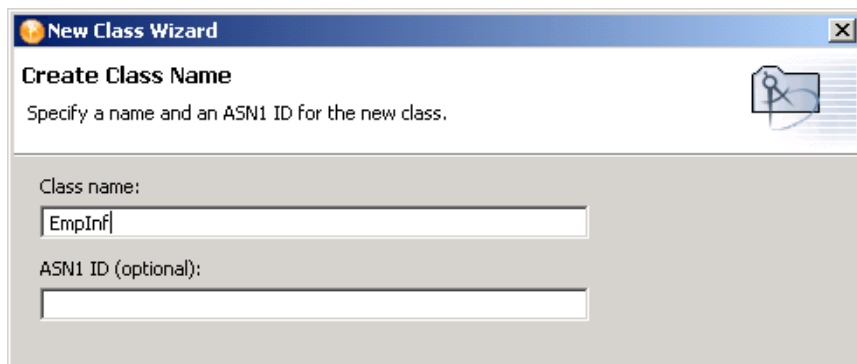


The *Classes* tab lists all classes that are defined in the schema and stored in Designer.

- 2 Decide whether to show changes.

The *Only show changes* option is below the Classes pane. This option enables you to see only classes that are different from the base schema (stored in the `BaseIVSchema.xml` file). If the base schema is unchanged, the list is empty.

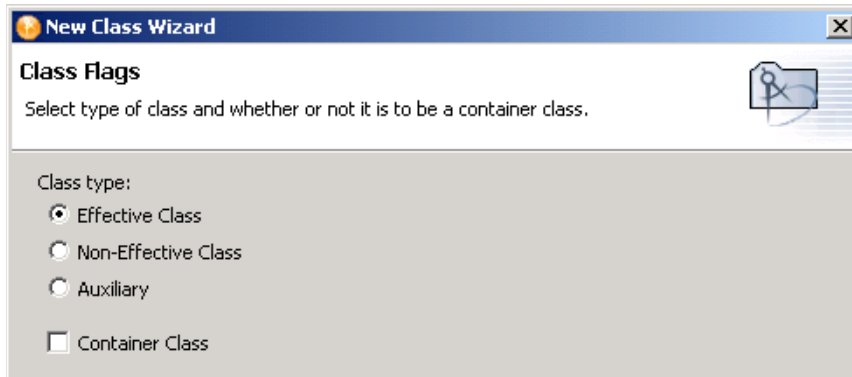
- 3 Click the *Add a Class* icon  on the Classes toolbar.
- 4 Specify the name of the class (for example, `EmpInfo`), and (if applicable) an ASN1 ID, then click *OK*.



If you register your schema definition with Novell, Novell provides an ASN1 number. It resolves the possibility of schema collisions caused by duplicate schema names with different

definition structures. You register your schema definition if you want to make the schema definition publicly available.

- 5 Select the type of class (for example, Effective Class), then click *Next*.



Flag	Description
<i>Effective</i>	You can create an instance of the defined object in the eDirectory tree.
<i>Noneffective</i>	Only used to define other classes. You cannot create an object of a noneffective class.
<i>Auxiliary</i>	Combines attributes to be added to other classes by extending the object class attribute.
<i>Non Removable</i>	Sets the class so it cannot be removed from the schema.
<i>Container</i>	Sets the object as a container object instead of a leaf object. If it is set to container, this object can contain other objects.

- 6 Select the classes that the new class will inherit from, then click *Next*.

Select a class in the *Available classes* pane and move it to the *Inherited classes* pane.

- 7 Select the mandatory attributes, then click *Next*.

The inheritances that display in the *Inherited mandatory attributes* pane depend upon the class that you selected.

- 8 Select optional attributes, then click *Next*.

The *Inherited optional attributes* pane lists default optional inheritances.

- 9 Select the naming attributes, then click *Next*.

The eDirectory schema allows for inheritance from other classes. A class that another class inherits from is called a super class. A class can inherit attributes from more than one super class.

Every class inherits from the super class Top. No class exists above Top. For example, Group inherits directly from Top, but User inherits from Organizational Person. Organizational Person inherits from Person. Person inherits from ndsLoginProperties, and ndsLoginProperties inherits from Top.

10 Select the containment classes.

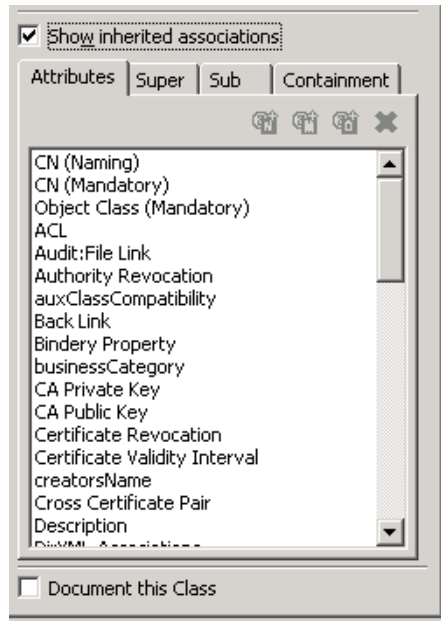
In eDirectory, an object is a leaf object or a container object. For example, if you select the class Group, the Manage Schema tool lists the domain, Organization, and Organizational Unit classes, which can contain the Group class

11 Review the summary, then click *Finish*.

The new class appears in the Classes pane.

12 Save the settings by clicking *OK*.

13 Show inherited associations.



Attributes: If you select *Show inherited associations*, Designer lists all attributes that are associated with a class, whether assigned or inherited. If you don't select this option, only the assigned attributes are listed.

The *Attributes* tab also lists the type of attributes. For example, the class Group has a naming attribute of CN and a mandatory attribute of CN. All of the other attributes in the list are optional attributes.

You can add and delete attributes from the selected class by selecting the proper icon.





Super: The eDirectory schema allows for inheritance from other classes. A class that another class inherits from is called a super class. The Super tab lists the selected class's immediate super class.

A class can inherit attributes from more than one super class. The super class that every class inherits from is Top. No class exists above Top. For example, Group inherits directly from Top, but User inherits from Organizational Person. Organizational Person inherits from Person. Person inherits from ndsLoginProperties, and ndsLoginProperties inherits from Top.

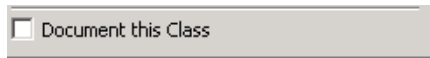
Sub: The Sub tab list all classes that inherit from the selected class. If a class isn't listed in the Sub tab, no classes inherit from the selected class.

Containment: In eDirectory, an object is a leaf object or a container object. The Containment tab lists other classes that can contain the selected class.

For example, if you select the class Group, the Manage Schema tool lists the domain, Organization, and Organizational Unit classes, which can contain the Group class.

Class Field	Description
Add Naming 	Adds a naming attribute to the class that is selected.
Add Mandatory 	Adds a mandatory attribute to the class that is selected.
Add Optional 	Adds an optional attribute to the class that is selected.
Delete 	Deletes an attribute from being associated with the select class.

14 Document the class.








If you select Document this Class, Designer documents information on the class when you run Document Generation.

Icons and Fields in the Classes Page

The following tables explains icons in the Classes toolbar:

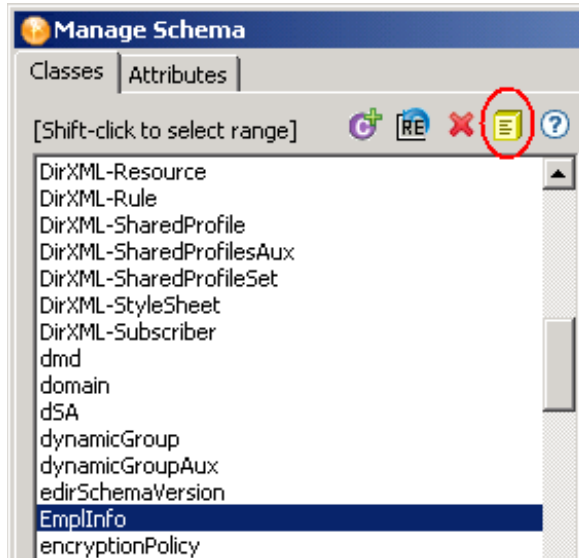
Table 11-1 Icons in the Classes Toolbar

Icon	Description
Add a Class 	Creates an eDirectory class.
Rename cClass 	Renames any class that is not a base class. Not an available option for base classes.
Delete Class 	Deletes any class that is not a base class. Not an available option for base classes.
Schema Notes 	Enables you to add notes about any class you create. Not available for base classes.
Help 	Launches the help for the Manage Schema tool.

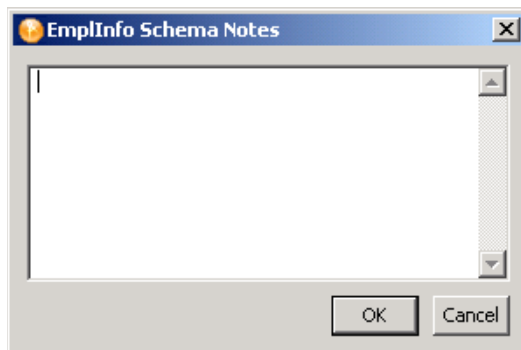
Adding a Note

Designer allows you to add notes about any class you create. The information is stored as `desc` in the `.ldif` file and as a note in the `.sch` file.

- 1 Select the class you want to add a note to, then click the *Schema Notes* icon.



- 2 Type the note in the window, then click *OK*.



11.2.2 Creating eDirectory Attributes

- ♦ [“How to Create eDirectory Attributes” on page 194](#)
- ♦ [“Syntax for Attributes” on page 198](#)

How to Create eDirectory Attributes

- 1 In the Modeler, right-click the Identity Vault, then select *Manage Vault Schema*.


By default, the *Classes* tab is active. If you want to view or modify attributes for a particular class, select that class before you select the *Attributes* tab.

- 2 Select the *Attributes* tab.

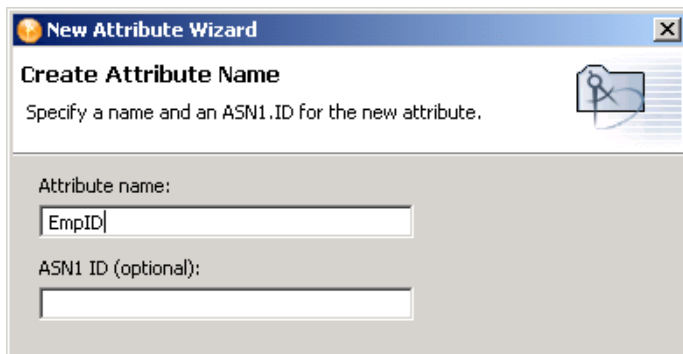
The *Attributes* pane lists all attributes that are defined in the schema and stored in Designer. You can view all attributes at once, or view the attributes associated with a specific class. When you select a class from the drop-down list, the attributes associated with that class are listed.

3 Decide whether to show changes.

The *Only show changes* option is below the *Attributes* pane. This option enables you to see only attributes that are different from the base schema (stored in the `BaseIVSchema.xml` file). If the base schema is unchanged, the list is empty.

4 Click the *Add an Attribute* icon .

5 Specify the name of the new attribute (for example, EmpID) and (if applicable) an ASN1 ID, then click *Next*.



If you register your schema definition with Novell, Novell provides an ASN1 number. It resolves the possibility of schema collisions caused by duplicate schema names with different definition structures. You register your schema definition if you want to make the schema definition publicly available.

6 Select the syntax, then click *Next*.

An attribute syntax defines a standard data type that an attribute uses to store its values in the eDirectory tree. Syntax is required for each attribute. See [Table 11-2 on page 199](#).

7 Select the flags for the attribute.

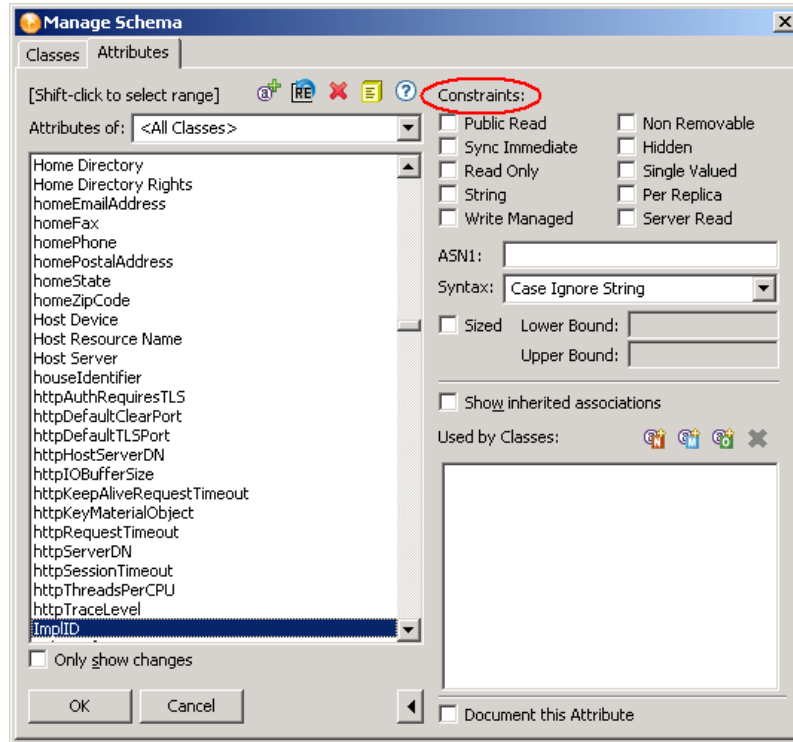
Flag	Description
Single Valued	Allows one value to be stored in the attribute.
Sync Immediate	When the attribute is modified, it is synchronized immediately to all of the servers in the replica ring.
Public Read	Allows anyone to read this attribute without the Read privilege specifically assigned. You can't use inheritance masks to prevent an object from reading attributes with this constraint.
Write Managed	Explicit rights are granted before this attribute can be changed. In order to modify this attribute, users must have managed rights on the object to change the attribute.
Per Replica	Allows one value to be stored in the attribute.

8 Review the summary, then click *Finish*.

The new attribute appears in the *Attribute* pane.

9 Select the constraints for the attribute.

The following figure illustrates the constraints that display on the Manage Schema tool.



The constraints restrict the information that is stored in the data type and constrain the operations of eDirectory and eDirectory clients.

Constraint	Description
Public Read	Allows anyone to read this attribute without the read privilege specifically assigned. You can't use inheritance masks to prevent an object from reading attributes with this constraint.
Sync Immediate	When the attribute is modified, it is synchronized immediately to all of the servers in the replica ring.
Read Only	The attribute is available to be read, but it cannot be modified.
String	Allows only string information to be stored in the attribute.
Write Managed	Explicit rights are granted before this attribute can be changed. In order to modify this attribute, users must have managed rights on the object to change the attribute.

Constraint	Description
Non Removable	The attribute cannot be deleted from schema.
Hidden	Only applications running on a NetWare® server can use this attribute.
Single Valued	Allows one value to be stored in the attribute.
Per Replica	Allows one value to be stored in the attribute.
Server Read	The attribute can be read by an NCP™ server object even though the right to read is not inherited or explicitly granted. The NCP server object is always able to read this attribute, regardless of the rights granted in the ACL.

10 Specify size.

If the size of the schema definition is limited, select *Sized*, then specify the size in decimal form.

The following figure illustrates the *Sized* field in the Manage Schema tool. The example in this figure has no value.

The screenshot shows a user interface with a checked checkbox labeled 'Sized'. To its right are two input fields. The first is labeled 'Lower Bound:' and contains the number '0'. The second is labeled 'Upper Bound:' and also contains the number '0'.

This value is the length of the schema definition in the eDirectory database, in bytes. The tool accepts the information in decimal form. In order to use the information, the eDirectory database converts it to hexadecimal.

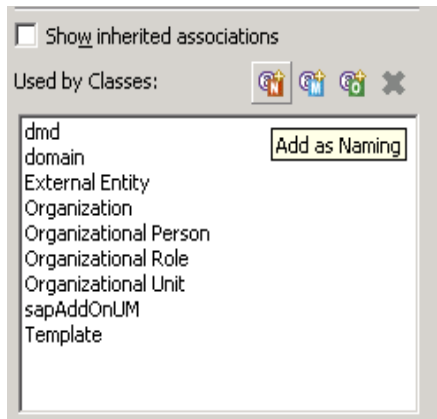
11 Show inherited associations.

The schema allows for inheritance of other attributes from super classes. If you select this item, all attributes that are associated with a class, whether assigned or inherited, are listed. If you don't select this item, only the attributes assigned are listed.

The screenshot shows a user interface with an unchecked checkbox labeled 'Show inherited associations'. Below it is a section titled 'Used by Classes:' with a list of class names. The list includes: applicationEntity (Naming), applicationProcess (Naming), Audit:File Object (Naming), Bindery Object (Naming), Bindery Queue (Naming), cRLDistributionPoint (Naming), Device (Naming), DirXML-Driver (Naming), DirXML-DriverSet (Naming), DirXML-Entitlement (Naming), DirXML-Job (Naming), and DirXML-Library (Naming). There are navigation icons (back, forward, search, close) above the list.

Used by Classes lists all classes that use the selected attribute. If you select *Show inherited associations*, the list includes classes that inherit the attribute.

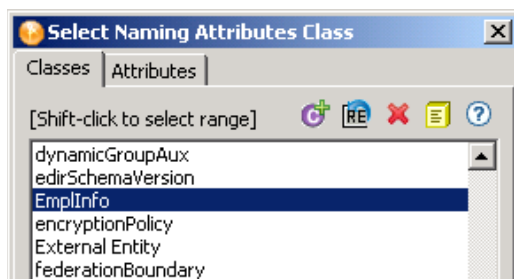
12 Specify an attribute type.



The attribute types help define the class.

Icon	Description
Add Naming	Adds the selected attribute to a class as a naming attribute.
Add Mandatory	Adds the selected attribute to a class as a mandatory attribute.
Add Optional	Adds the selected attribute to a class as an optional attribute
Delete	Deletes a class from being associated with the selected attribute.

- 13 Select the class to associate with the attribute.



- 14 Select whether to document this attribute.

If you select *Document this attribute*, the data is included when you run Generate Documentation.

- 15 Save changes by clicking *OK*.

Syntax for Attributes

The following information from Novell LogicSource for eDirectory gives details about the eDirectory schema. LogicSource is a subscription-based service Novell provides to its customers. This section contains only a small portion of the information available in LogicSource for

eDirectory. For more information about LogicSource, see [Novell Technical Subscriptions \(http://support.novell.com/subscriptions/articles/novell_logicsource.html\)](http://support.novell.com/subscriptions/articles/novell_logicsource.html).

Table 11-2 *Syntax for Attributes*

Syntax	Description
Back Link	The remoteID field identifies the backlinked object on the server, and the objectName field identifies the server holding an external reference.
Boolean	Two Boolean attributes match for equality if they are both True or both False. True is represented as one (1), and False is represented as zero (0). Any attribute defined using this syntax is single valued.
Case Exact String	Attributes using this syntax can set size limits. Two Case Exact Strings match for equality when they are of the same length and their corresponding characters are identical.
Case Ignore List	Two Case Ignore Lists match for equality if the number of strings in each is the same, and all corresponding strings match. For two corresponding strings in the list to match, they must be the same length and their corresponding characters must be identical (according to the rules for case ignore strings).
Case Ignore String	Used in attributes whose values are strings and where the case (upper or lower) is ignored.
Class Name	Used to match two class names where the case (upper or lower) is ignored.
Counter	The attribute is single valued. The syntax is similar to Integer, except that any value added to an attribute is arithmetically added to the total, and any value deleted is arithmetically subtracted from the total.
Distinguished Name	The attribute is the distinguished name of the object up to 256 Unicode* characters. This is not case sensitive.
E-Mail Address	Used to match attributes whose values are e-mail addresses and whose lengths and corresponding characters are identical; however, it ignores case (upper and lower). Only the E-Mail Address attribute uses this syntax.
Facsimile Telephone Number	Facsimile Telephone Number values are matched based on the telephone number field. The rules for matching fax telephone numbers are identical to those for the Case Exact syntax except that all space and hyphen (-) characters are skipped during the comparison. Only the Facsimile Telephone Number attribute uses this syntax.
Hold	This syntax is an accounting quantity, which is an amount tentatively held against a subject's credit limit, pending completion of a transaction. In the wire format, the Subject field is the distinguished name of the object. eDirectory treats the Hold amount similarly to the Counter syntax, with new values added to or subtracted from the base total. If the evaluated Hold amount goes to 0 (zero), the Hold record is deleted.
Integer	The attribute is an integer. Attributes using this syntax can set size limits.
Interval	The Interval value is the number of seconds in a time interval.
Net Address	Stores the network address as a binary string. The string is the literal value of the address. It lists the type of communication protocol used.

Syntax	Description
Numeric String	Two numeric strings match for equality when they are of the same length and their corresponding characters are identical. It matches the digits 0-9 and spaces if they are contained in the numeric string.
Object ACL	An Object ACL value can protect either an object or an attribute. The protected object is always the one that contains the ACL attribute. If an ACL entry is to apply to the object as a whole, the protected attribute name should be left empty (NULL). If a specific attribute is to be protected, it should be named in the ACL entry.
Octet List	<p>A presented octet list matches a stored list if the presented list is a subset of the stored list. Octet strings are so designated because they are not interpreted by the Directory. They are simply a series of bits with no Unicode implications.</p> <p>The length is the number of bits divided by 8 and rounded to the nearest integer. Thus, each octet represents eight bits of data. The number of data bits is always evenly divisible by 8.</p>
Octet String	For two octet strings to match, they must be the same length and the corresponding bit sequence (octets) must be identical. When comparing two strings, the first pair of octets that do not match are used to determine the order of the strings. Octet strings are not Unicode strings.
Path	The string represented by the path field is compared for equality using the same rules that Case Exact String uses. That is, two paths match for equality when their lengths and corresponding characters, including case, are identical.
Postal Address	An attribute value for Postal Address is typically composed of selected attributes from the MHS Unformatted Postal O/R Address version 1 according to Recommendation F.401. The value is limited to 6 lines of 30 characters each, including a Postal Country Name. Normally the information contained in such an address could include a name, street address, city, state or province, postal code, and possibly a postal office box number depending on the specific requirements of the named object.
Printable String	<p>The following characters are in the printable string character set.</p> <p>A...Z a...z 0...9 Space Character ' Apostrophe (Left Parenthesis) Right Parenthesis + Plus Sign , Comma - Hyphen . Period / Slash : Colon = Equal Sign ? Question Mark</p> <p>Two printable strings match for equality when they are the same length and their corresponding characters are identical. Case (upper or lower) is significant when comparing printable strings. For example, as printable strings, "Jones" and "JONES" do not match.</p>

Syntax	Description
Replica Pointer	<p>Each value of the replica pointer syntax is composed of five parts:</p> <ul style="list-style-type: none"> ◆ The complete name of the server that stores the replica. ◆ A value describing a the capabilities of this copy of the partition: master, secondary, read-only, or subordinate reference. ◆ A value indicating the current state of the replica (new, dying, locked, changing state, splitting, joining, or moving). ◆ A number representing the replica. All replicas for a partition have a different number assigned when the replica is created. ◆ A referral that contains a count of the addresses and one or more network addresses that hints at the node where the server probably resides. Because servers are accessible over different protocols, the server might have an address for each supported protocol.
Stream	Streams are files of information. The data stored in a stream file has no syntax enforcement of any kind. It is purely arbitrary data, defined by the application that created and uses it. The attribute is singled valued.
Telephone Number	The length of telephone number strings must be between 1 and 32. Two telephone numbers string match for equality when they are of the same length and their corresponding characters are identical. All spaces and hyphen (-) characters are skipped during the comparison.
Time	A time value consists of a whole number of seconds, where zero equals 12:00 midnight, January 1, 1970, UTC.
Timestamp	<p>A Timestamp value contains three components:</p> <ul style="list-style-type: none"> ◆ The <i>wholeSeconds</i> field consists of the whole number of seconds, where zero equals 12:00 midnight, January 1, 1970, UTC. ◆ The <i>replicaNum</i> field identifies the server that created the Timestamp. A replica number is assigned whenever a replica is created on a server. ◆ The <i>eventID</i> field is an integer that orders events occurring within the same whole-second interval. The event number restarts at one for each new second.
Typed Name	<p>The syntax names an eDirectory object and attaches two numeric values to it:</p> <ul style="list-style-type: none"> ◆ The level of the attribute indicates the priority. ◆ The interval indicates the frequency of references. <p>The the objectName or Distinguished Name identifies the eDirectory object referred to by the Typed Name.</p>
Unknown	Unknown syntax is used to stop the loss of data, if the eDirectory database becomes corrupted. When an object becomes Unknown, there is information stored in this attribute that can allow the object to be recovered. This syntax is used by eDirectory.

11.3 Modifying the Schema


Designer allows you to modify the eDirectory schema. The following sections describe fields and definitions used in the Manage Schema tool for classes and attributes.

- ◆ [Section 11.3.1, “Deleting Schema Definitions,” on page 202](#)

- ♦ [Section 11.3.2, “Modifying Classes or Attributes,” on page 202](#)
- ♦ [Section 11.3.3, “Renaming Schema Definitions,” on page 202](#)

11.3.1 Deleting Schema Definitions

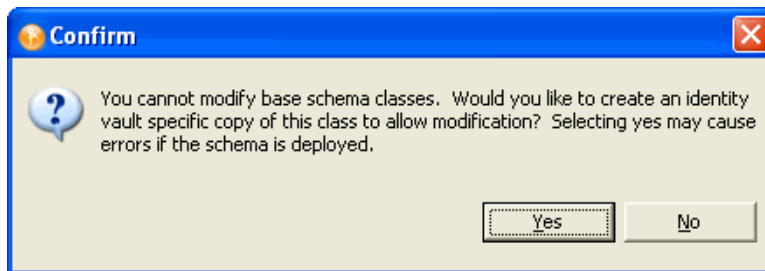
You can delete an extended schema definition. You cannot delete any base schema. If you select a base schema item, the *Delete* icon is dimmed, indicating that it is unavailable.

- 1 In the Modeler, right-click an Identity Vault, then select *Manage Schema*.
- 2 Select the class or attribute that you want to delete, then click the *Delete* icon .

11.3.2 Modifying Classes or Attributes

- 1 In the Modeler, right-click an Identity Vault, then select *Manage Vault Schema*.
- 2 Select the class or attribute that you want to modify.
- 3 Select a field.

If you select a base schema class or attribute, the following pop-up message appears:



It is best to modify only the extended schema and not the base schema. Modifying the base schema can cause data corruption and eDirectory synchronization errors.

- 4 Modify the item, then click *OK*.

11.3.3 Renaming Schema Definitions

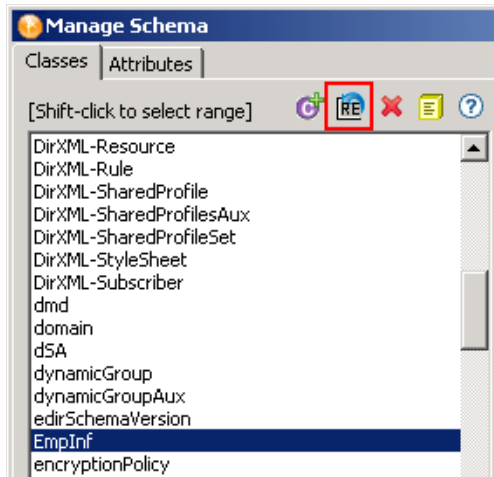
You can rename extended schema definitions. You cannot rename any base schema classes or attributes. If you select a base schema item, the *Rename* icon is dimmed, indicating it is unavailable.

- ♦ [“Renaming a Class” on page 202](#)
- ♦ [“Renaming an Attribute” on page 203](#)

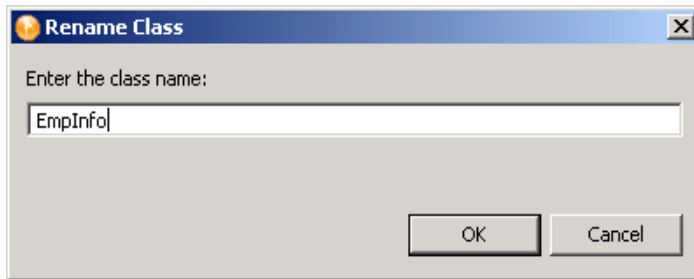
Renaming a Class

- 1 In the Modeler, right-click an Identity Vault, then select *Manage Vault Schema*.

- 2 In the *Class* page, select a class that you want to rename, then click the *Rename Class* icon.

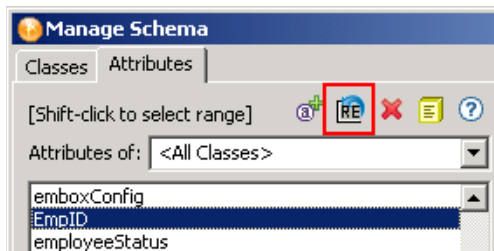


- 3 Specify the new name of the class, then click *OK*.

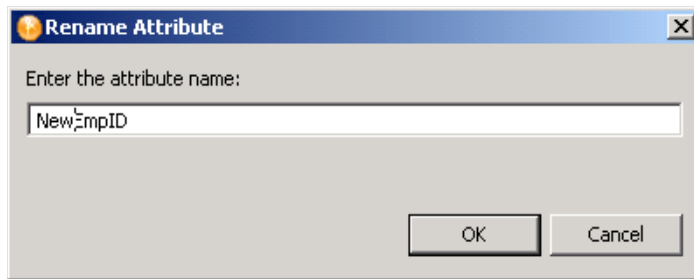


Renaming an Attribute

- 1 In the Modeler, right-click the *Identity Vault* icon, then select *Manage Vault Schema*.
- 2 Select an attribute you want to rename in the attribute tab, then click the *Rename an Attribute* icon.



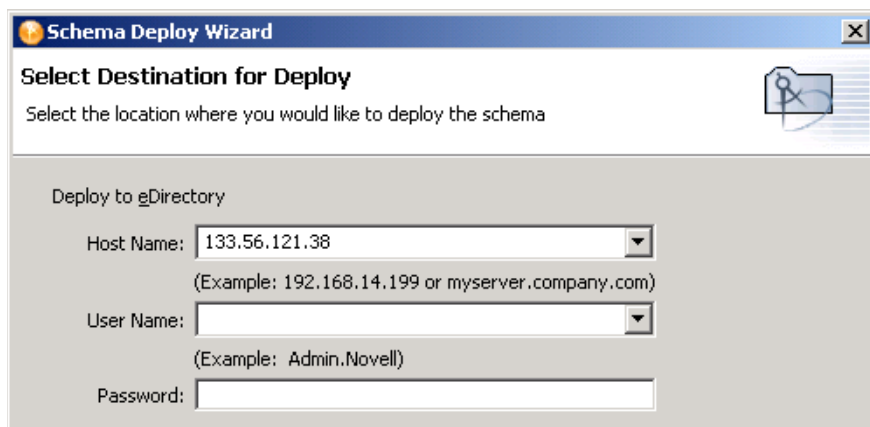
- 3 Specify the new name of the attribute, then click *OK*.



11.4 Deploying the Schema into the eDirectory Tree

After the Identity Manager driver is tested with the new schema, you can deploy the modified schema into eDirectory.

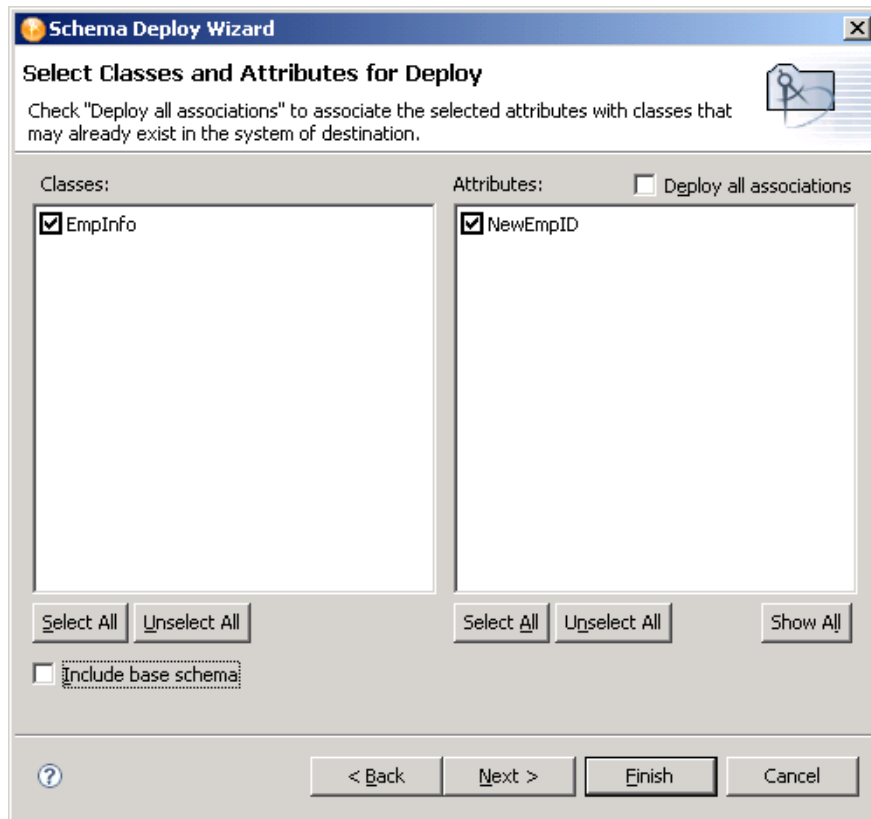
- 1 In the Modeler, select the Identity Vault.
- 2 Select *Live > Deploy Schema*.
- 3 Specify the *Host Name*.



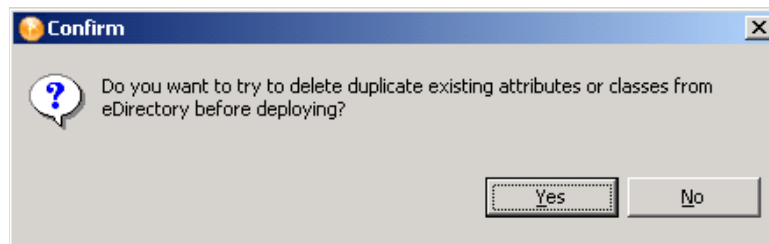
The host name can be the server's IP address or the DNS name of the server.

- 4 Specify the *User Name*, which must be a user with administrative rights to the schema.
- 5 Specify the user's password, then click *Next*.

- 6 Select the classes and attributes to deploy into the eDirectory schema, then click *Next*.

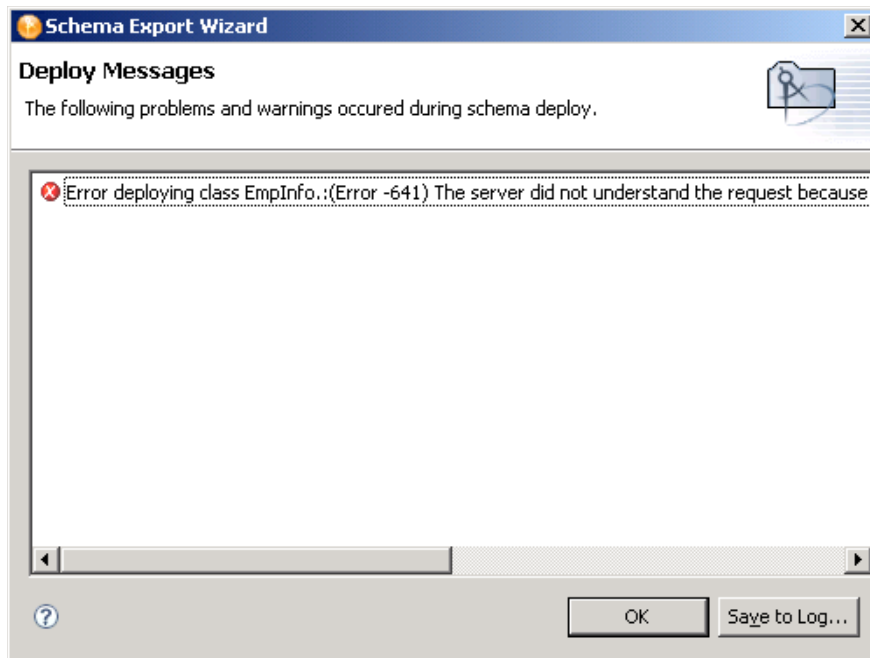


- 7 Review the summary of classes and attributes to be deployed, then click *Finish*.
If you have selected duplicate attributes or classes, a warning box appears



- 8 Select *Yes* or *No*, depending upon whether you want to resolve the duplicate classes or attributes.

- 9 Review errors or warnings, then click *OK*.



11.5 Exporting the Schema to a File

- Section 11.5.1, “Exporting the Schema to a .sch File,” on page 206
- Section 11.5.2, “Exporting the Schema to an LDIF File,” on page 208

11.5.1 Exporting the Schema to a .sch File

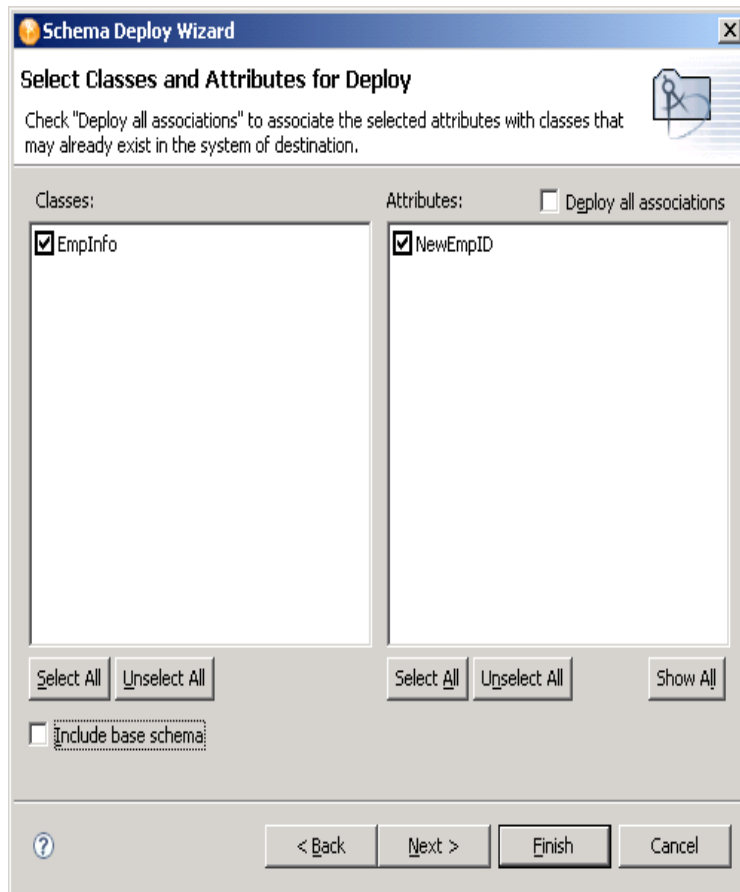
- 1 In the Modeler, right-click an Identity Vault, then select *Export to File > Schema*.
- 2 Select *.sch format*.



- 3 Specify a path or browse to where you want to save the schema file, specify a filename, then click *Save > Next*.

Designer appends the *.sch* extension when you export the file.

- 4 Select the classes and attributes to export to the .sch file, then click *Finish*.

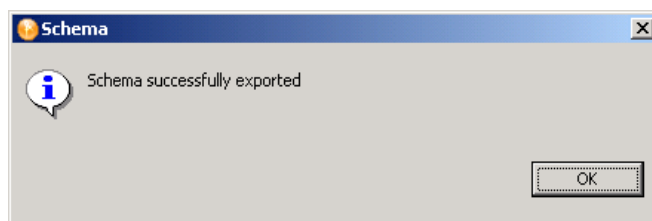


Deploy all associations (above the *Attributes* pane) enables you to associate the selected attributes with the classes that might already exist in the eDirectory tree. If you do not select this box, the new attributes that should be associated with the class are not associated.

For example, if the Employee Photo attribute is associated with the User class, and *Deploy all associations* is not selected, Employee Photo is not associated with the User class.

The classes and attributes that are in Designer are listed in the two columns. All classes and attributes are selected by default. To prevent a class or attribute from being deployed, deselect it. To add all classes and attributes, click *Select All*. To remove all classes and attributes, click *Unselect All*.

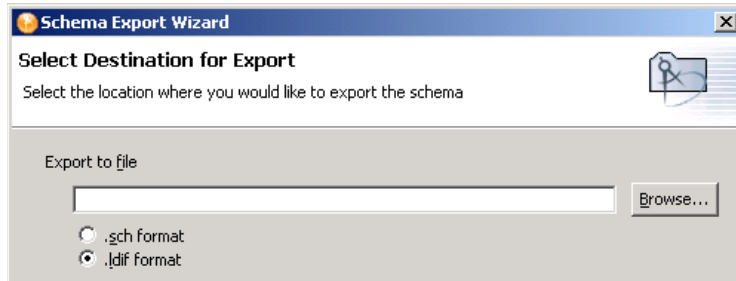
- 5 Click *Finish*.



If errors occur, a deploy summary screen lists them.

11.5.2 Exporting the Schema to an LDIF File

- 1 In the Modeler, right-click the Identity Vault, then select *Export to File > Schema*.
- 2 Select *Export to file*.
- 3 Select *.ldif format*.



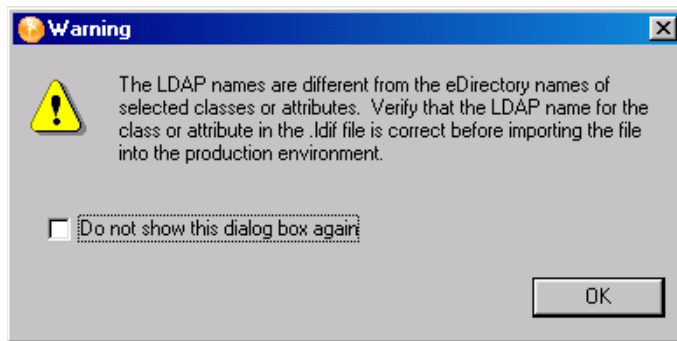
- 4 Specify a path or browse to where you want to save the schema file, specify a filename, then click *Save > Next*.

Designer appends the `.ldif` extension when you export the file.

- 5 Select the classes and attributes to export to the `.ldif` file, then click *Finish*.

The classes and attributes that are in Designer are listed in the two columns. All classes and attributes are selected by default. To prevent a class or attribute from being deployed, deselect it. To add all classes and attributes, click *Select All*. To remove all classes and attributes, click *Unselect All*.

- 6 Click *OK* in the warning.



The class and attribute names in Designer are the eDirectory names. The names for the classes and attributes in the LDIF file are the LDAP names. The eDirectory names differ from the LDAP names. Verify that the names listed in the LDAP file are correct for your environment before importing the file. For a list of eDirectory class and attribute names mapped to LDAP class and attribute names, see [Section 11.8, "Mapping eDirectory to an LDAP Schema," on page 220](#).

- 7 Click *OK*.

If errors occur, a deploy summary screen lists them.

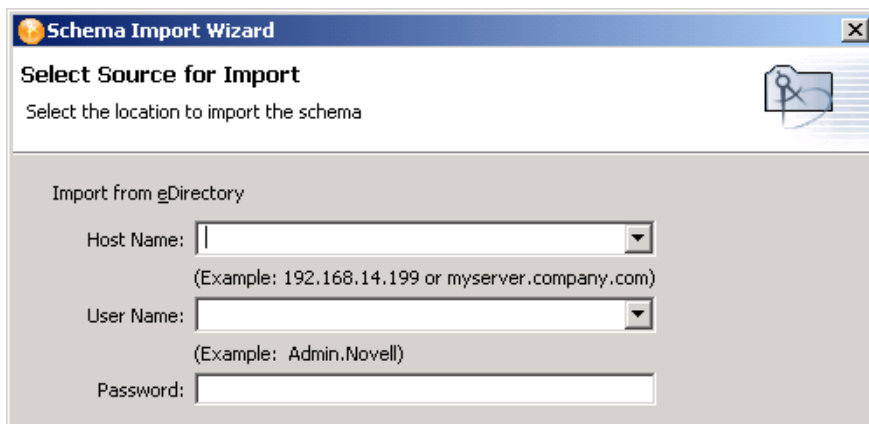
11.6 Importing the Schema

Designer allows you to import the schema from your production environment to do in-depth testing with the Identity Manager drivers.

- ♦ [Section 11.6.1, “Importing the Schema from the eDirectory Tree,” on page 209](#)
- ♦ [Section 11.6.2, “Importing the Schema from a File,” on page 211](#)

11.6.1 Importing the Schema from the eDirectory Tree

- 1 In the Modeler, select an Identity Vault.
- 2 From the main menu, select *Live > Import Schema*.
- 3 Specify information for the server that has the schema that you want to import.

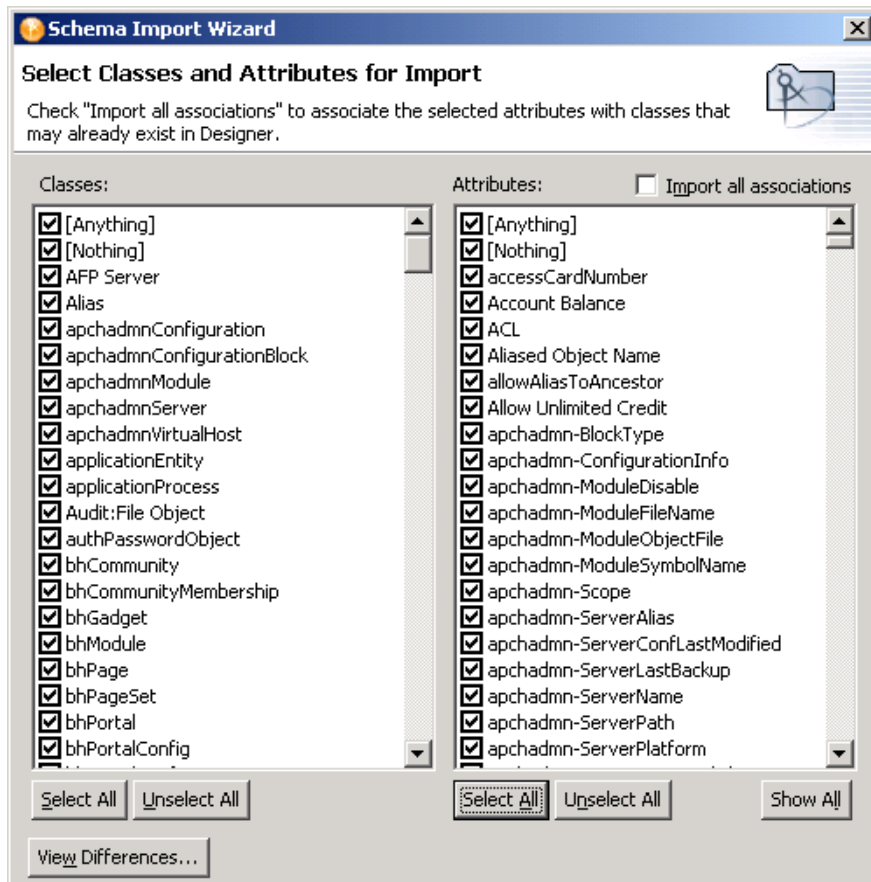


The host name can be the server’s IP address or the DNS name of the server.

The user name must be a user with administrative rights to the schema.

- 4 Click *Next*.

- 5 Select the classes and attributes to import into Designer, then click *Next*.



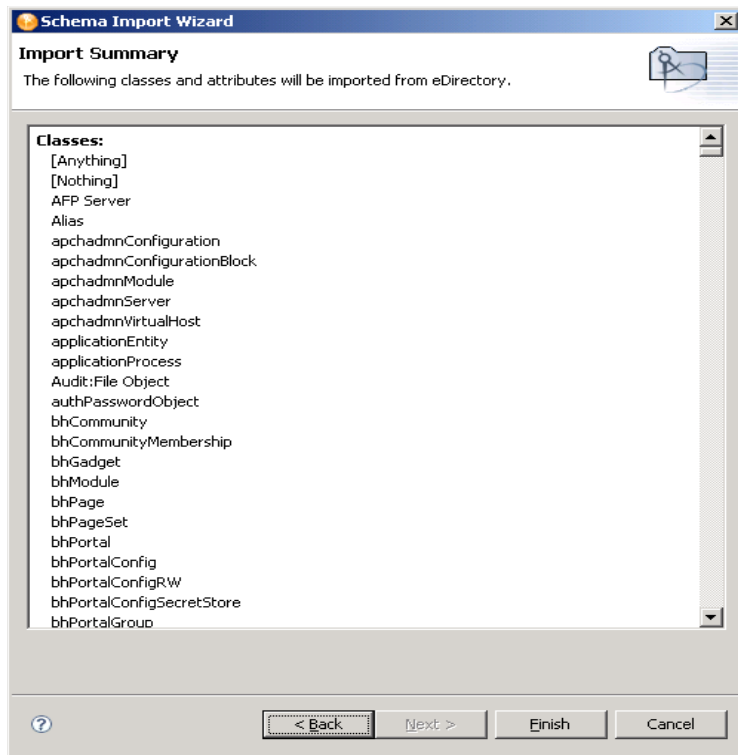
Import All Associations: Enables you to associate the selected attributes with the classes that might already exist in Designer. If you do not select this box, the new attributes that should be associated with the class are not associated.

For example, if the attribute of Employee Photo is associated with the User class, and you do not select *Import all associations*, Employee Photo is not associated with the User class.

View Differences: Enables you to view the differences in the schema between the eDirectory tree and Designer.

When you click *View Differences*, Designer lists the differences in a separate window. To select all classes and attributes that are different from what is in Designer, click *Select All*. If you do not want the different items automatically selected, click *Cancel*.

6 Review classes and attributes that Designer will import.



7 Click *Finish*.

If errors occur, a summary screen lists them.

11.6.2 Importing the Schema from a File

NetWare servers use `.sch` files. Servers on other platforms use `.ldif` files.

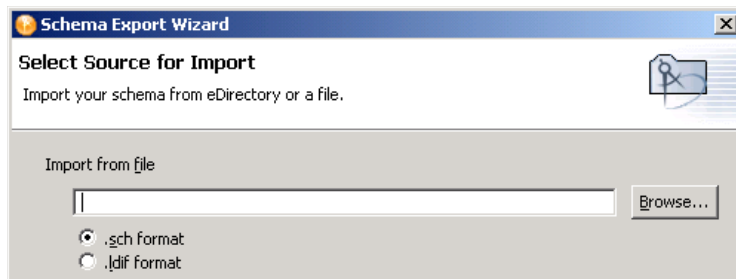
When you created an Identity Vault in the Modeler, Designer created a base schema in your project. If a `.sch` file or `.ldif` file has been saved, you can quickly add classes and attributes for your drivers by importing classes and attributes from the saved file.

- ◆ “Importing the Schema to a `.sch` File” on page 211
- ◆ “Importing the Schema to an LDIF File” on page 213

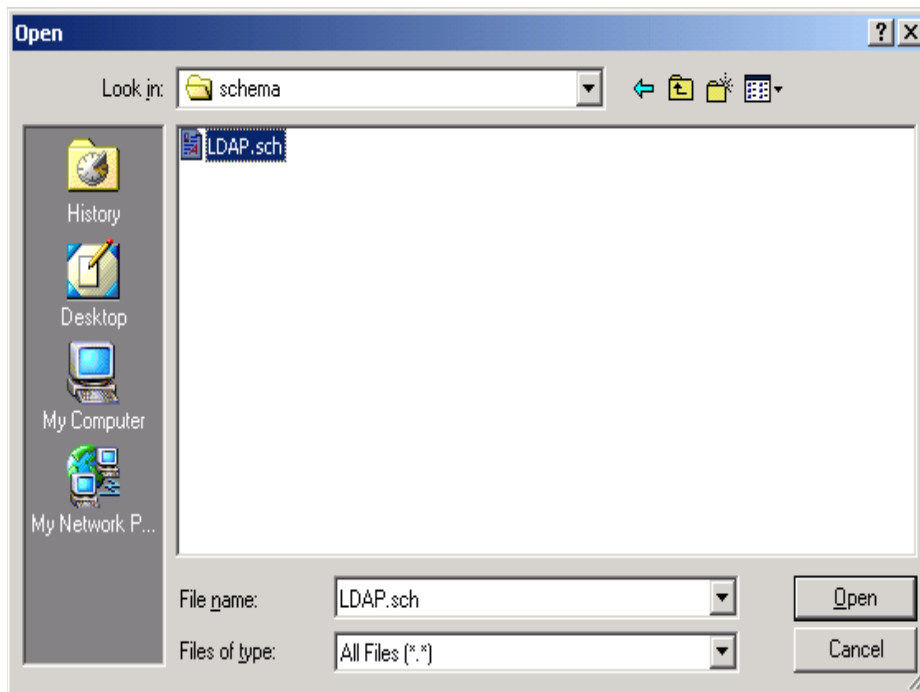
Importing the Schema to a `.sch` File

- 1 In the Modeler, right-click the Identity Vault that will use the imported `.sch` file.
- 2 Select *Import Schema from File*.

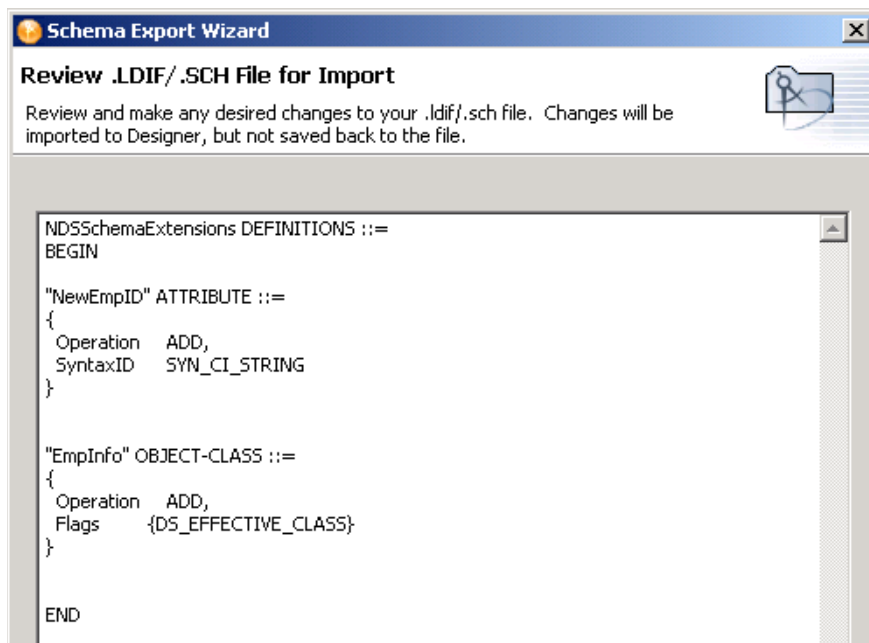
3 Select *.sch* format.



4 Browse to and select the *.sch* file that you want to use, then click *Open*.

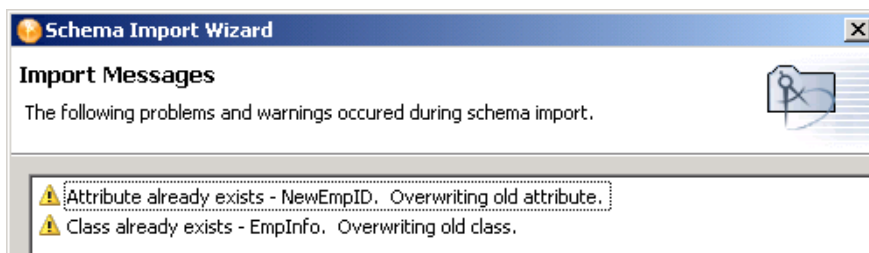


5 Click *Next*, then review the .sch file.



6 Make changes if necessary, then click *Finish*.

7 Click *OK*.



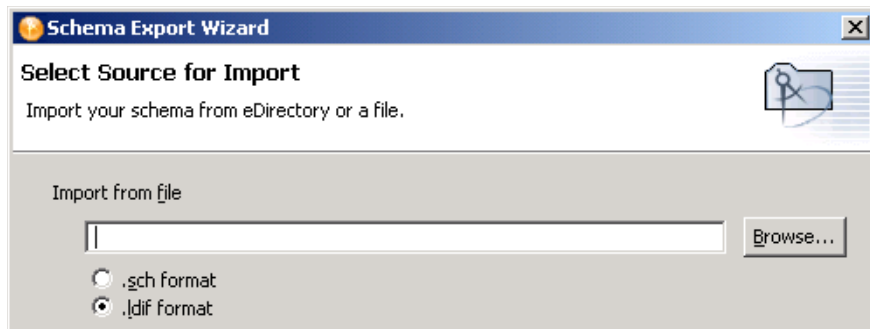
If errors occur, a deploy summary screen lists them.

Importing the Schema to an LDIF File

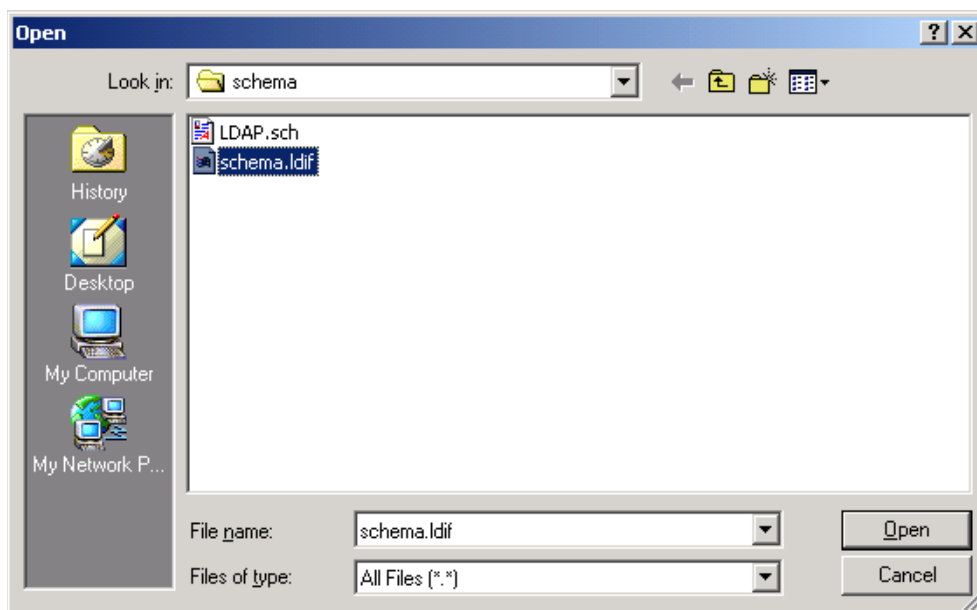
1 In the Modeler, right-click the Identity Vault that will use the imported .ldif file.

2 Select *Import Schema from File*.

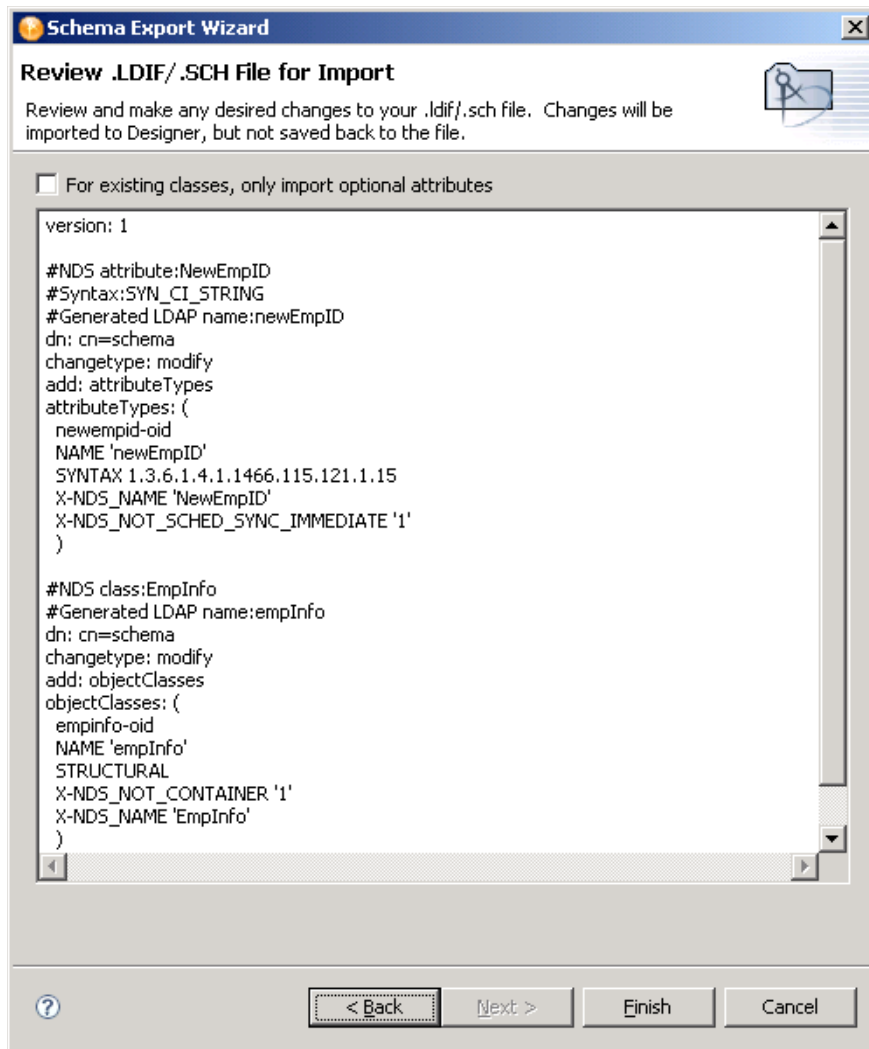
3 Select *.ldif* format.



4 Browse to and select the *.ldif* file that you want to use, then click *Open*.

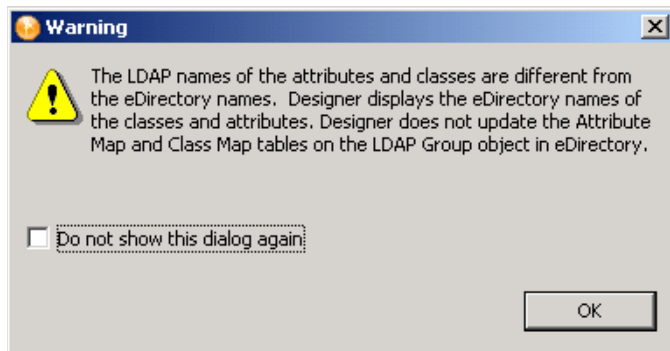


5 Click *Next*, then review the .ldif file.



6 Make changes if necessary, then click *Finish*.

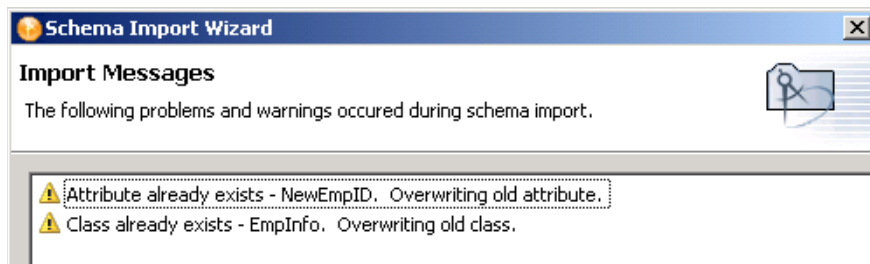
7 In the Warning dialog box, click *OK*.



The class and attribute names in Designer are the eDirectory names. The names for the classes and attributes in the LDIF file are the LDAP names. The eDirectory names differ from the

LDAP names. Verify that the names listed in the LDAP file are correct for your environment before importing the file. For a list of eDirectory class and attribute names mapped to LDAP class and attribute names, see [Section 11.8, “Mapping eDirectory to an LDAP Schema,” on page 220](#).

8 Click *OK*.



If errors occur, a deploy summary dialog box lists them.

11.7 Managing a Copy of an Application Schema

The Identity Manager engine currently uses the application schema for the following:

- ♦ DirXML Script uses the dn-format/dn-delims to figure out how to parse or convert DN's coming from and going to the application.
- ♦ The multi-valued flag on attributes are used during the attribute merge process that happens as part of a match, resync, or migrate.
- ♦ [Section 11.7.1, “Editing an Application’s Schema,” on page 217](#)
- ♦ [Section 11.7.2, “Refreshing the Application Schema,” on page 219](#)

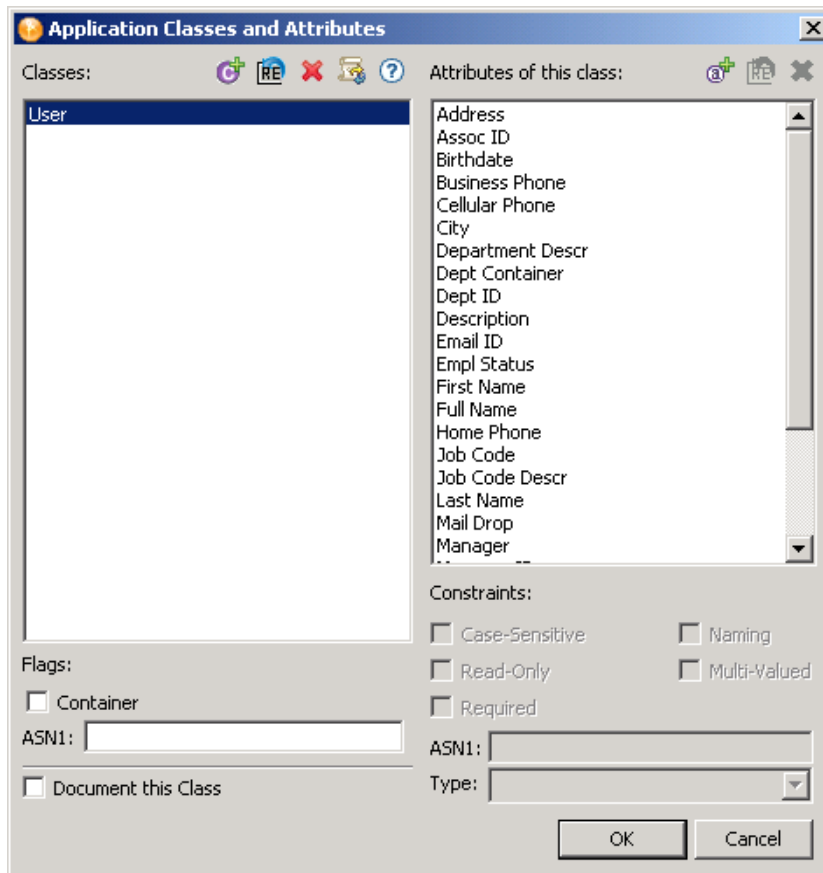
11.7.1 Editing an Application's Schema

Designer enables you to manage a copy of the connected system's schema. You can make changes to a copy of the application schema so that you can test the Identity Manager drivers in Designer. The schema changes cannot be deployed into the live application schema.

- 1 Right-click the driver connection in Designer, then select *Manage Application Schema*.



2 Add, rename, or delete the application's classes or attributes, then click *OK*.



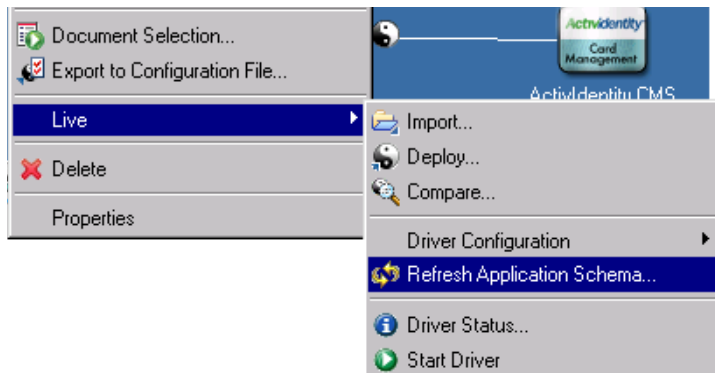
Field	Description
<i>Classes</i>	Lists all of the classes stored in Designer from the application's schema.
<i>Add a class</i>	Adds a new class.
<i>Rename class</i>	Renames the selected class.
<i>Delete class</i>	Deletes the selected class.
<i>Refresh application schema</i>	Provides a new copy of the application's schema. This option is useful if the application schema changes.
<i>Help</i>	Launches the Help documentation for the Manage Schema tool.
<i>Flags: Container</i>	Specifies whether the class is a container.
<i>ASN1</i>	The unique ID of the class.

Field	Description
<i>Document This Class</i>	Adds the schema information into the Document Generator. For more information see Chapter 16, "Documenting Projects," on page 299.
<i>Attributes of This Class</i>	Lists all of the attributes stored in Designer for the selected class from the application's schema.
<i>Add an attribute</i>	Adds a new attribute for the selected class.
<i>Rename attribute</i>	Renames the selected attribute.
<i>Delete attribute</i>	Deletes the selected attribute.
<i>Constraints</i>	Specifies the details of the attribute. To edit the constraints, select an attribute.
<i>Type</i>	Specifies the syntax of the attribute. To view the syntax, select an attribute. To change the syntax, select an option from the drop-down list.

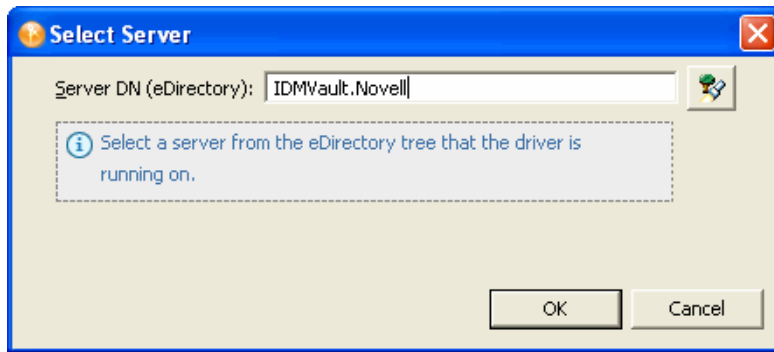
11.7.2 Refreshing the Application Schema

If the application schema changes, you can get a new copy of the application's schema by refreshing the application schema.

- 1 Right-click the driver connection, then select *Live > Refresh Application Schema*.



2 Click the browse icon.



3 Browse to and select the server where the driver is installed, then click *OK* twice.

11.8 Mapping eDirectory to an LDAP Schema

When eDirectory is accessed through LDAP, the names of classes and attributes might be different than when accessed through the standard NCP-based APIs.

For more information about how that mapping is performed, see

- ♦ [Class and Attribute Mappings \(http://www.novell.com/documentation/edir88/index.html?page=/documentation/edir88/edir88/data/h0000007.html#a5bwxyz\)](http://www.novell.com/documentation/edir88/index.html?page=/documentation/edir88/edir88/data/h0000007.html#a5bwxyz)
- ♦ [NDK: Novell eDirectory Schema Reference \(http://devsup.novell.de/ndk/doc/ndslib/index.html?page=/ndk/doc/ndslib/schm_enu/data/h4q1mn1i.html\)](http://devsup.novell.de/ndk/doc/ndslib/index.html?page=/ndk/doc/ndslib/schm_enu/data/h4q1mn1i.html) at the Novell Developer Support Web page

Managing the Flow of Data

12

Designer allows you to manage how the data flows between the Identity Vault and the connected systems. You can see how the data flows between all of the connected systems, make changes as needed, create reports about the data, and view the flow of passwords between the systems.

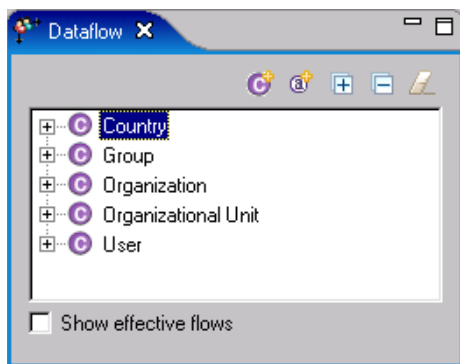
The Dataflow view and the Dataflow editor manage the data. The Dataflow view displays the flow of data in the modeler per driver. The Dataflow editor displays a more granular view.

- ◆ [Section 12.1, “The Dataflow View,” on page 221](#)
- ◆ [Section 12.2, “The Dataflow Editor,” on page 227](#)
- ◆ [Section 12.3, “Adding Items in the Dataflow Editor,” on page 235](#)
- ◆ [Section 12.4, “Editing Items,” on page 239](#)
- ◆ [Section 12.5, “Generating HTML Reports,” on page 243](#)
- ◆ [Section 12.6, “Integrating Passwords,” on page 245](#)

12.1 The Dataflow View

The following figure illustrates the Dataflow view. You can use it to control the flow of data between the Identity Vault and connected systems. The Modeler displays the dataflow.

Figure 12-1 *The Dataflow View*

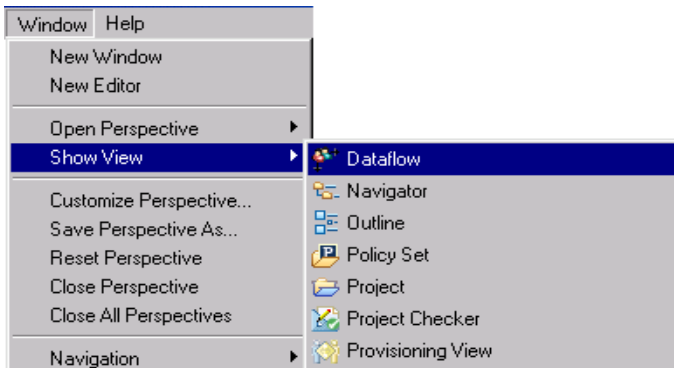


- ◆ [Section 12.1.1, “Accessing the Dataflow View,” on page 221](#)
- ◆ [Section 12.1.2, “Flow Arrows in the Modeler,” on page 222](#)
- ◆ [Section 12.1.3, “Viewing How Attributes Are Synchronized,” on page 224](#)
- ◆ [Section 12.1.4, “Changing the Data Flow,” on page 224](#)

12.1.1 Accessing the Dataflow View

If you have closed the Dataflow view, you can access it by selecting *Window > Show View > Dataflow*.

Figure 12-2 *Selecting the Dataflow View*

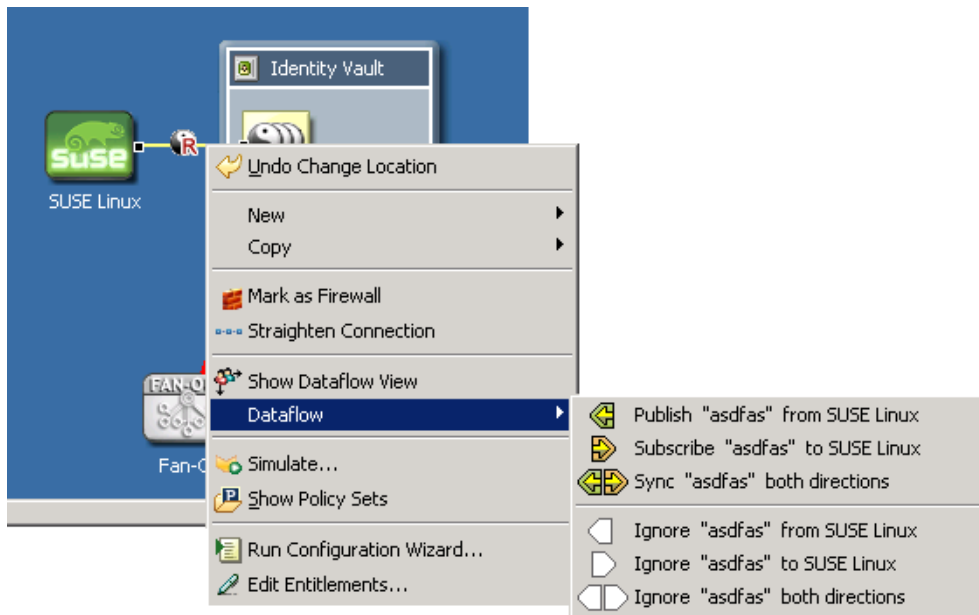


If the Dataflow view is blank and no project is displayed in the Modeler:

- 1 Expand a project in the Project view.
- 2 Open the project by double-clicking *System Model*.
Objects and icons appear in the Dataflow view.

If you want to change how the data flows from the Modeler:

- 1 Right-click a driver or application in the Modeler.

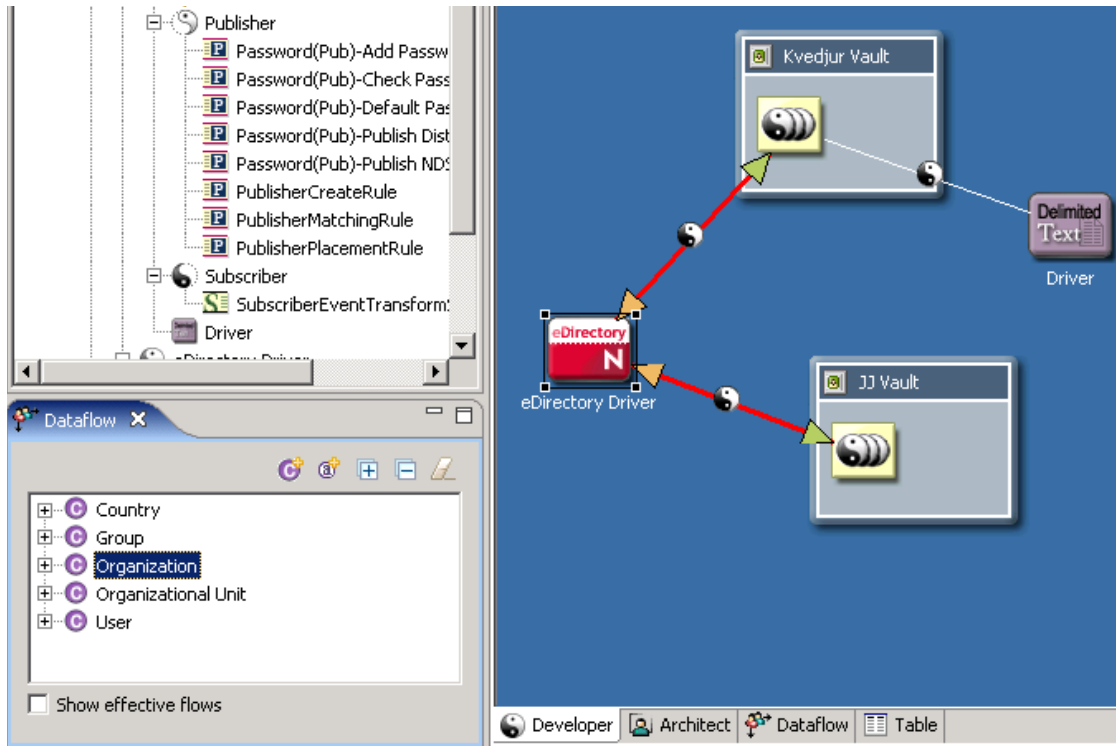


- 2 Select *Dataflow*, then select how you want the data flow to change.

12.1.2 Flow Arrows in the Modeler

When the Dataflow view opens, it automatically reads the filters and shows the classes and attributes. (If a filter with classes and attributes doesn't exist, create one.)

Figure 12-3 *Flow Arrows in the Modeler*



As you select a class (for example,) or attribute in the Dataflow list, the appropriate driver lines are highlighted in red in the Modeler. Icons enable you to see Sync, Notify, Reset, and Ignore filter settings all at the same time.

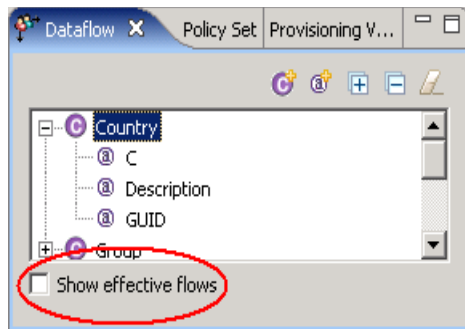
Table 12-1 *Dataflow Icons*

Icon	Description
	Green arrow: the Publisher channel is synchronized.
	Orange arrow: the Subscriber channel is synchronized.
	Bell: the attribute is set to Notify.
	Reset arrow: the attribute is set to Reset.
No decoration	The attribute is set to Ignore.

The color coding matches the Dataflow icons in the Filter editor and the Dataflow editor.

12.1.3 Viewing How Attributes Are Synchronized

Figure 12-4 Show Effective Flows

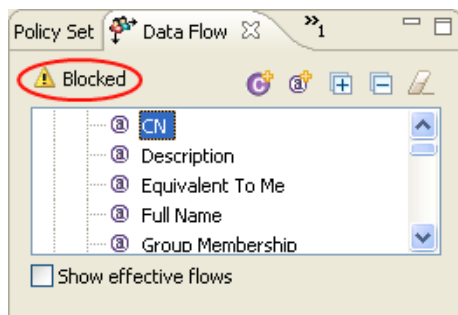


To view whether attributes are synchronized or whether they will be notified, select *Show effective flows*. When you select this check box, the synchronize arrows don't show if the parent class isn't set to synchronize. Therefore, you view an accurate diagram of actual flows.

However, if you want to view how attributes are configured to synchronize, regardless of the parent class, deselect *Show effective flows*. The synchronize arrows indicate which items are synchronized.

If you select an attribute that can't synchronize (whether or not *Show effective flows* is selected), you see a *Blocked* warning in the upper left. This warning indicates that this attribute can't be synchronized or notified because the parent class isn't synchronized.

Figure 12-5 The Blocked Text and Icon



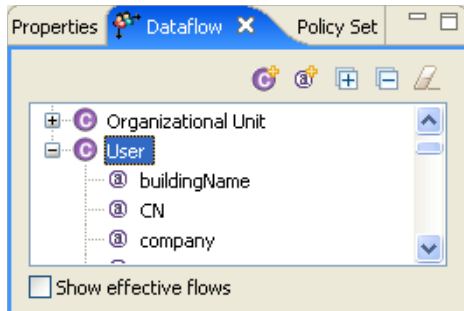
To view an explanation, hover the mouse over the *Warning* icon.

12.1.4 Changing the Data Flow

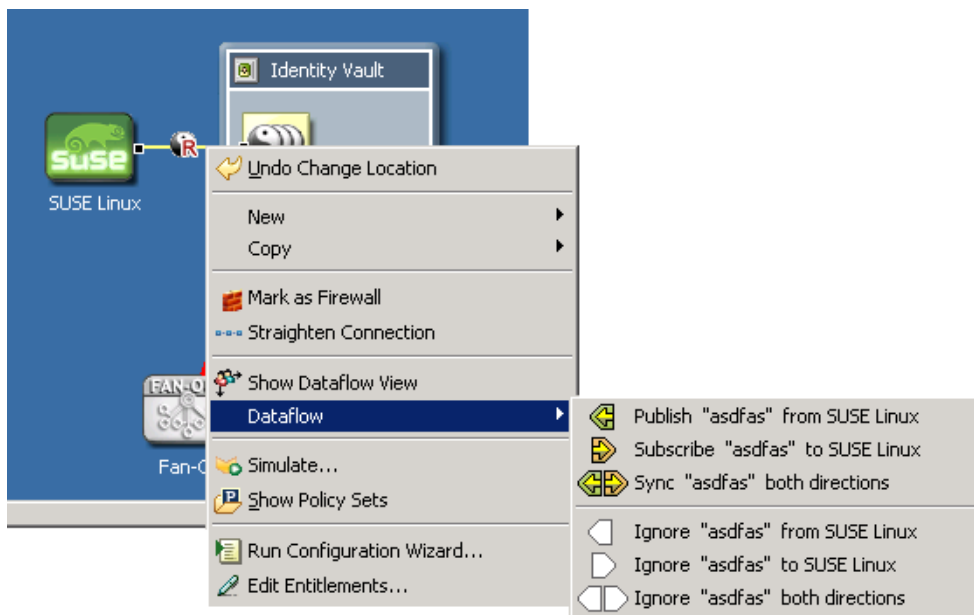
You can change how the data flows for classes and attributes from the Dataflow view.

To change the flow for a class:

- 1 Select a class in the Dataflow view.



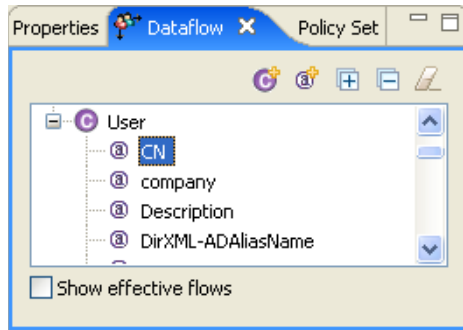
- 2 Right-click a driver line in the Modeler.
- 3 Select *Dataflow*.



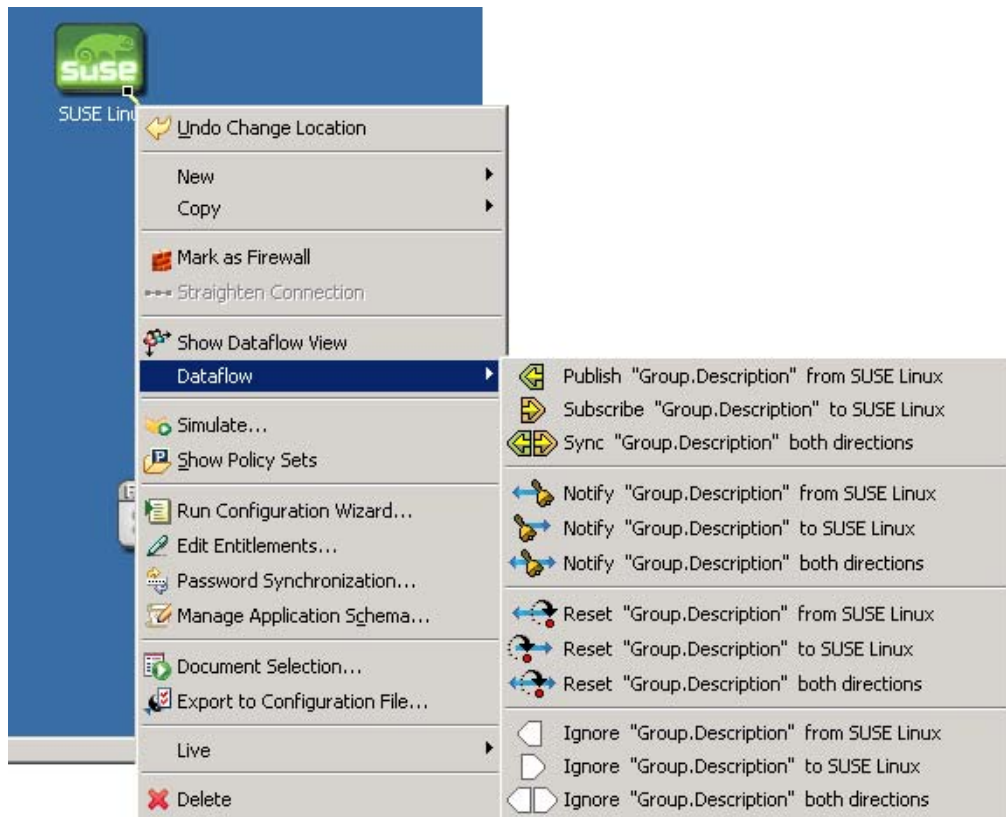
- 4 Select the option to change the data flow for the class.

To change the flow for an attribute:

- 1 Select an attribute in the Dataflow view.



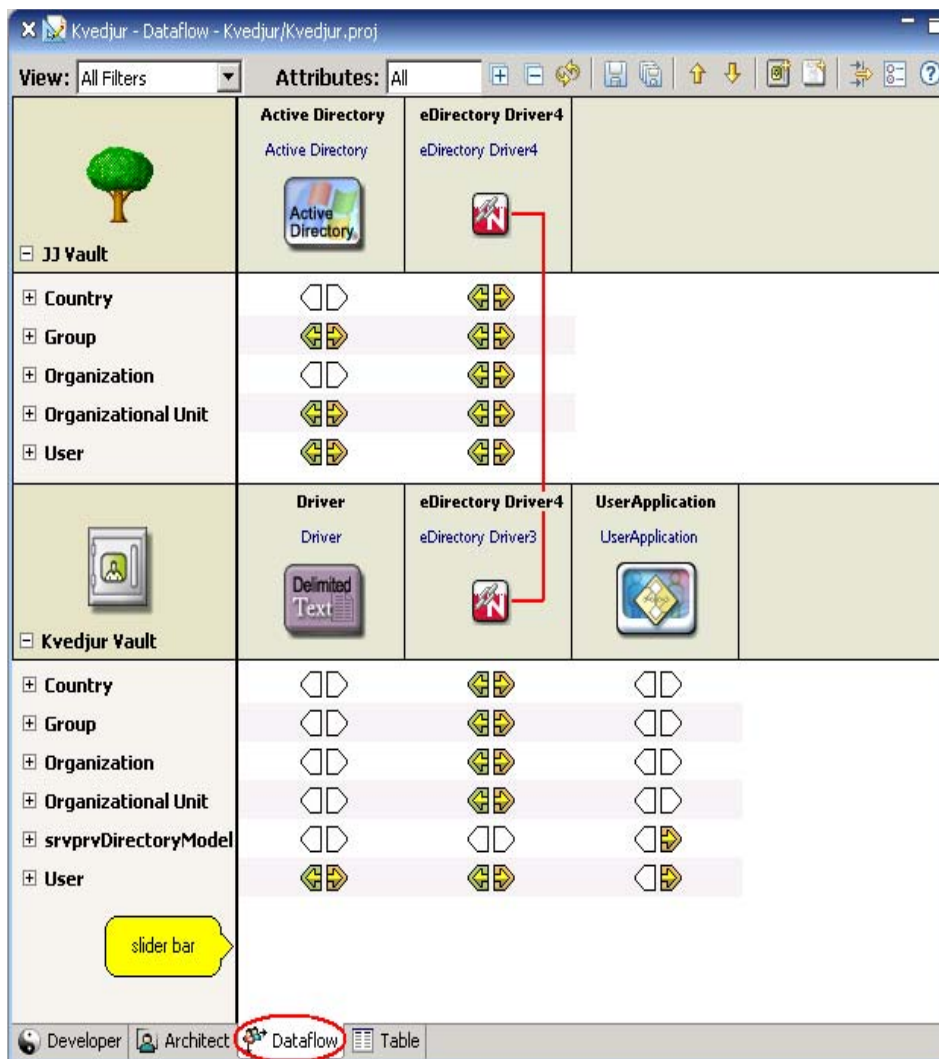
- 2 Right-click a driver line in the Modeler.
- 3 Select *Dataflow*.



- 4 Select the option to change the data flow for the attribute.

12.2 The Dataflow Editor

Figure 12-6 The Dataflow Editor



The Dataflow editor enables you to do the following:

- ◆ Use filters to display how data flows between all systems and Identity Vaults.
- ◆ View how passwords flow from each server.
- ◆ Generate reports of the data.

When object additions, deletions, changes, and selections synchronization occur, the Dataflow editor synchronizes with the Modeler and the Outline view.

To access the Dataflow editor, click the Dataflow tab.

To adjust the area for the Identity Vaults, move the slider bar. This setting persists and is restored the next time you run the editor.

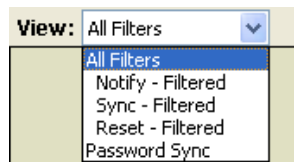
- ◆ [Section 12.2.1, "Filtering Views," on page 228](#)

- ◆ Section 12.2.2, “Filtering Identity Vaults and Applications,” on page 230
- ◆ Section 12.2.3, “Pinning the Identity Vault,” on page 230
- ◆ Section 12.2.4, “Expanding and Collapsing the Identity Vault,” on page 231
- ◆ Section 12.2.5, “Switching to an eDirectory Tree Icon,” on page 233
- ◆ Section 12.2.6, “Viewing an eDir-to-eDir Driver,” on page 234
- ◆ Section 12.2.7, “Keyboard Support,” on page 235

12.2.1 Filtering Views

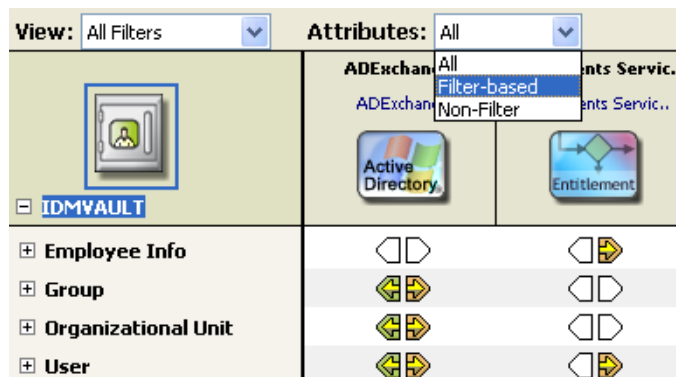
By default, the Dataflow editor shows all dataflows. The View drop-down list (in the upper left corner of the Dataflow editor, not in the Dataflow view), enables you to view notification, synchronization, reset, or Password Sync information. These filtered views do not allow you as much editing capability as the main view, but just what is necessary in that filter. For example, you can’t add attributes, vaults, or applications, because by default they wouldn’t appear in the filter.

Figure 12-7 Options to Filter Views in the Dataflow View



If you are in the All Filters view, you can further filter with the *Attributes* list. Because the Dataflow editor provides non-filter attributes, you can choose to view regular filter-based attributes, non-filter attributes, or both.

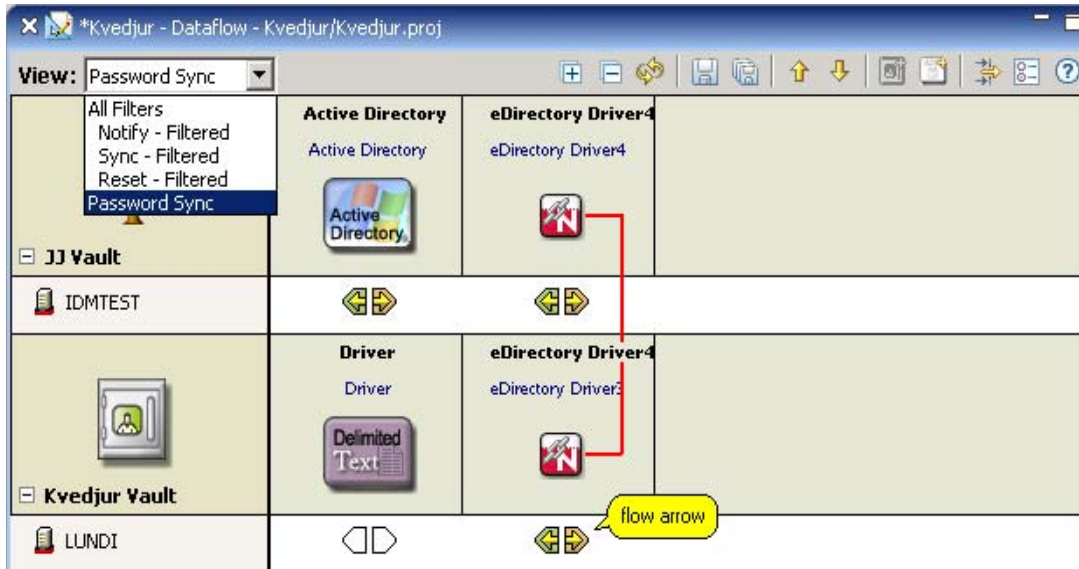
Figure 12-8 Options on the Attributes Combo Box



Synchronizing Passwords

The Password Sync view enables you to see and edit how all passwords flow in the project. Designer displays the information on a per-server basis and shows how passwords flow among all of the applications.

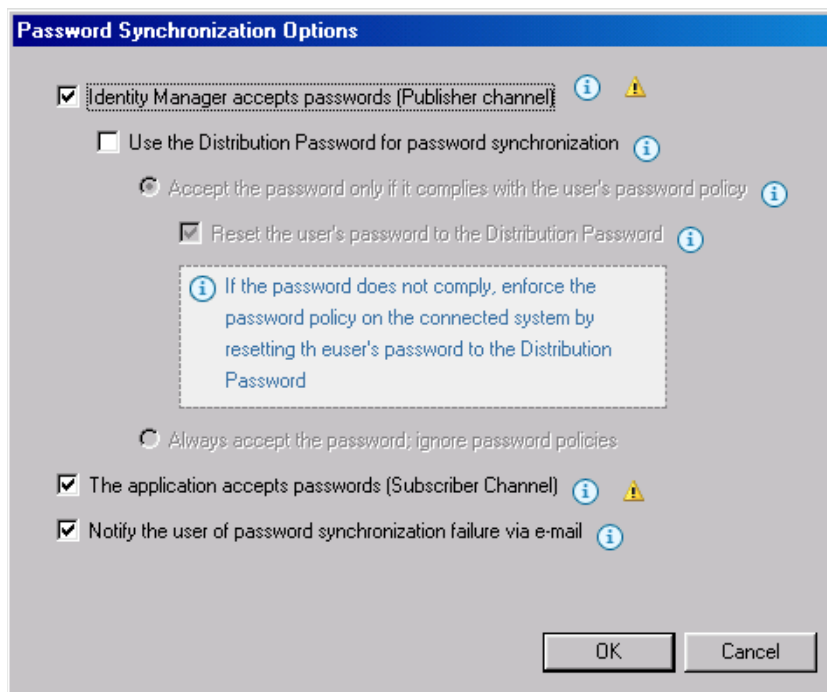
Figure 12-9 The Password Flow



To edit the password flow:

- 1 Select *Password Sync* in the View filter.
- 2 Double-click the flow arrow.
You can also right-click, then select *Password Synchronization*.
- 3 Edit the password synchronization options.

For more information about password synchronization, see “Password Synchronization across Connected Systems” in the *Novell Identity Manager Administration Guide* (<http://www.novell.com/documentation/idm35/index.html>).

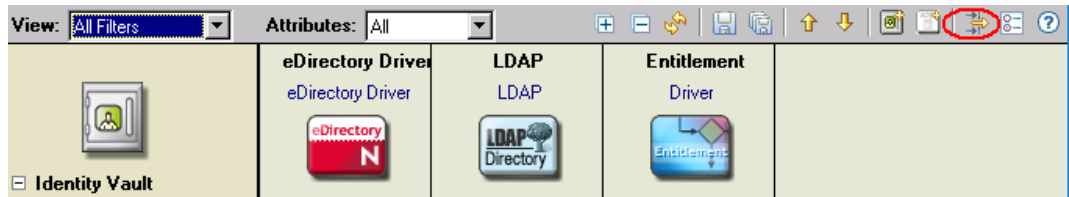


4 Click *OK*.

12.2.2 Filtering Identity Vaults and Applications

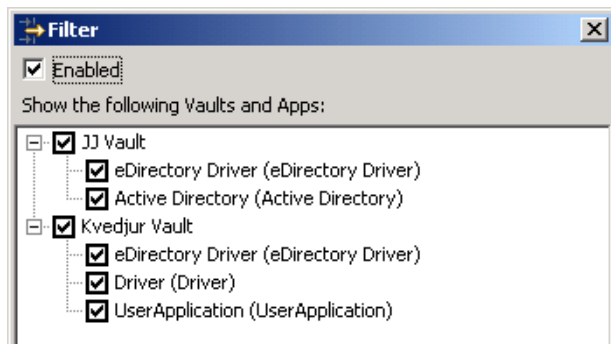
You can select the Identity Vaults and applications that you want to view in the editor.

1 In the Dataflow editor, click the *Filter View* icon.



2 Select *Enabled*.

The Identity Vaults and applications that you select here are included in the HTML reports. For more information, see [Section 12.5, “Generating HTML Reports,” on page 243](#).

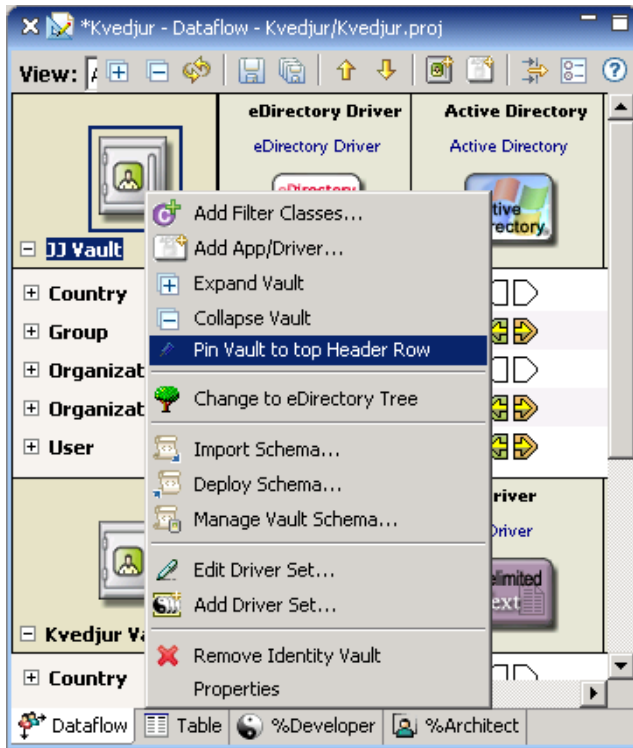


You can scroll and resize the dialog box. Also, you can interact with the Dataflow editor in the background, in any mode. This is convenient if you want to scroll a different section into view while this dialog box is up.

12.2.3 Pinning the Identity Vault

To change the scope of the editor to show a single Identity Vault, right-click the vault, then select *Pin Vault to Top Header Row*.

Figure 12-10 Pinning an Identity Vault



With a medium or large-sized project, the dataflow table can contain hundreds of rows and thousands of items. If you have multiple vaults and want to narrow the scope to more easily edit a vault without excessive scrolling, you might want to pin a vault. When an Identity Vault is pinned, a pin icon displays in the upper right corner.

Figure 12-11 A Pinned Identity Vault



To unpin the vault, right-click the Identity Vault, then select *Unpin Vault from Top Header Row*.

12.2.4 Expanding and Collapsing the Identity Vault

- ♦ “Expanding an Identity Vault” on page 231
- ♦ “Expanding All Identity Vaults” on page 232
- ♦ “Expanding Classes” on page 233

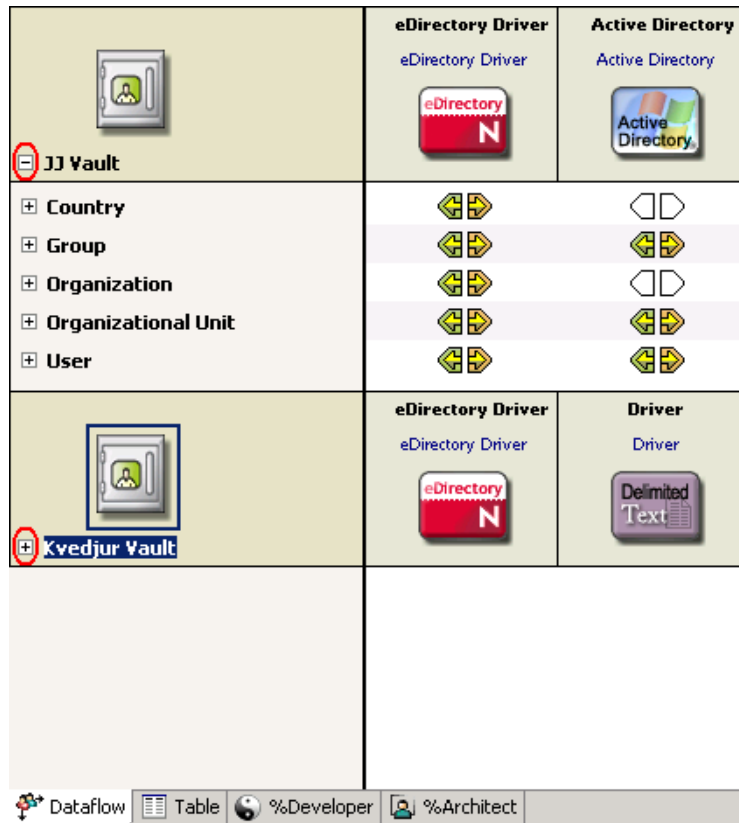
Expanding an Identity Vault

When the editor first loads, all vaults are expanded at the top level by default.

To collapse or expand the list of classes and attributes in an Identity Vault, do one of the following:

- ◆ Click the - or + icon below the Identity Vault icon.

Figure 12-12 Icons to Expand or Collapse the List of Classes

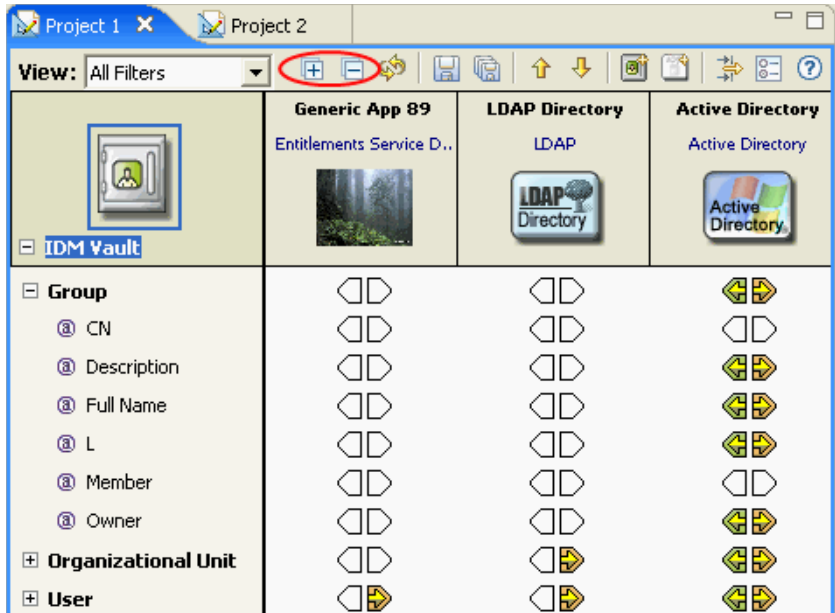


- ◆ Select the Identity Vault, then press the Right-arrow key to expand the information, or press the Left-arrow key to collapse the information.

Expanding All Identity Vaults

To expand or collapse the list of classes and attributes for all Identity Vaults, click the + or - icon on the toolbar.

Figure 12-13 Icons to Expand All Identity Vaults

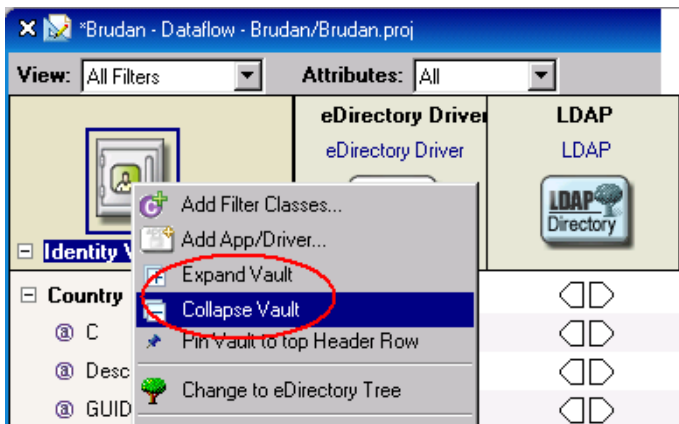


Expanding Classes

To view all attributes in a class, select the class, then press the Right-arrow key. To collapse the list of attributes, press the Left-arrow key.

To view all classes and attributes in an Identity Vault, right-click the Identity Vault icon, then select *Expand Vault*. To list just classes in an Identity Vault, right-click the Identity Vault, then select *Collapse Vault*.

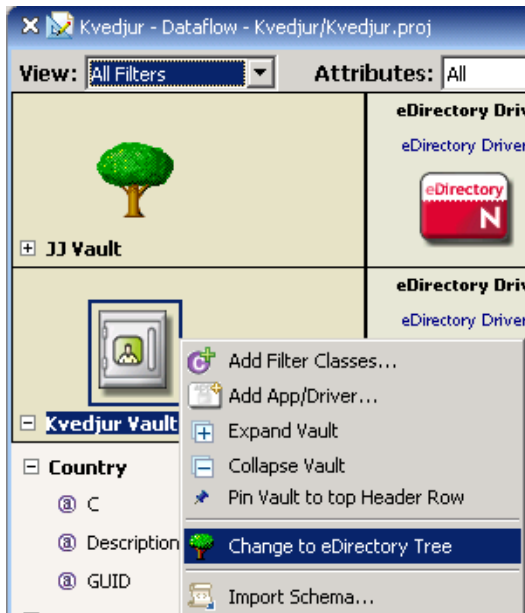
Figure 12-14 Menu Options to Expand an Identity Vault



12.2.5 Switching to an eDirectory Tree Icon

To switch from an Identity Vault icon to an eDirectory™ tree icon, right-click the Identity Vault, then select *Change to eDirectory Tree*.

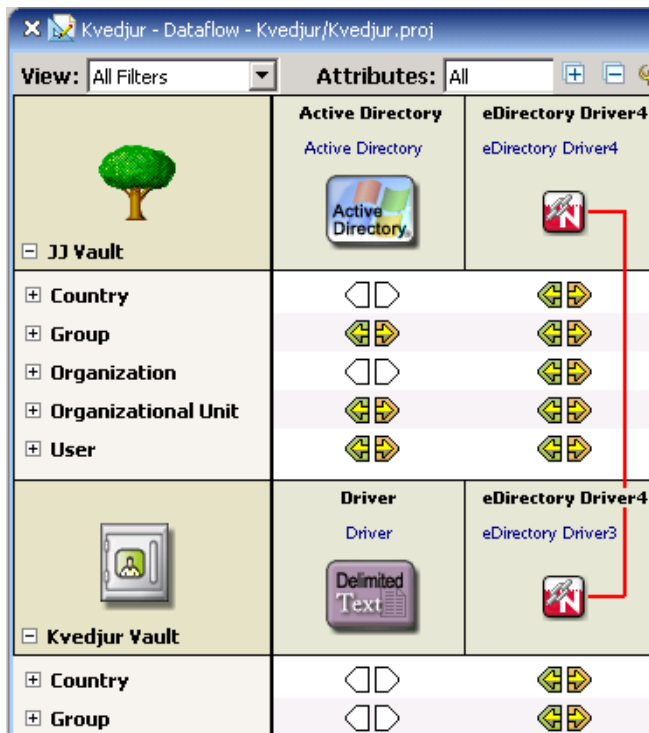
Figure 12-15 Changing to an eDirectory Tree



12.2.6 Viewing an eDir-to-eDir Driver

You can easily view both ends of an eDir-to-eDir connection so that you can configure the dataflows on both sides. Designer automatically detects the two eDirectory applications and aligns them in the same table column. A red line connects them.

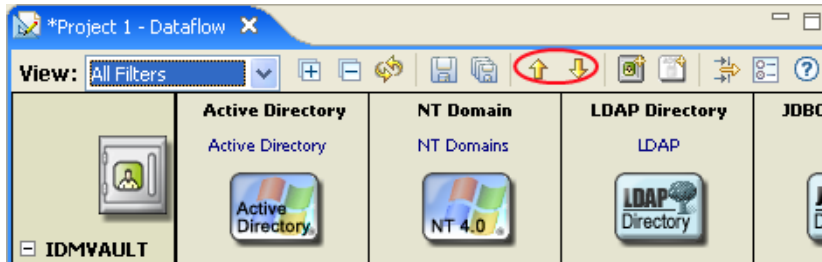
Figure 12-16 An eDir-to-eDir Connection



12.2.7 Keyboard Support

You can navigate by using the Up-arrow, Down-arrow, Left-arrow, and Right-arrow keys as well as PageUp, PageDown, Home, and End. In addition, you can navigate from one Identity Vault to another by clicking the Up-arrow or Down-arrow on the toolbar.

Figure 12-17 Navigation Icons

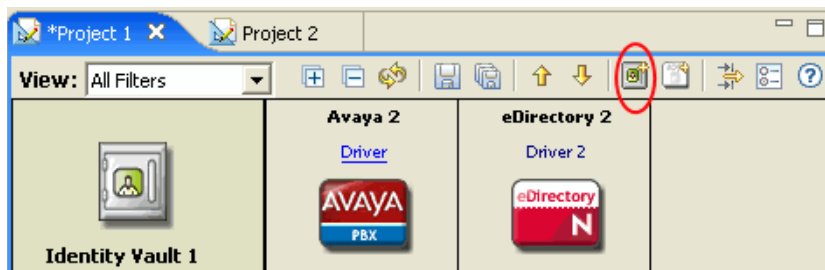


12.3 Adding Items in the Dataflow Editor

- ♦ Section 12.3.1, “Adding an Identity Vault in the Dataflow Editor,” on page 235
- ♦ Section 12.3.2, “Adding a Driver in the Dataflow Editor,” on page 236
- ♦ Section 12.3.3, “Adding an Application in the Dataflow Editor,” on page 236
- ♦ Section 12.3.4, “Adding Classes and Attributes,” on page 237
- ♦ Section 12.3.5, “Adding Non-Filter Attributes,” on page 238

12.3.1 Adding an Identity Vault in the Dataflow Editor

Figure 12-18 The Dataflow Editor’s Toolbar



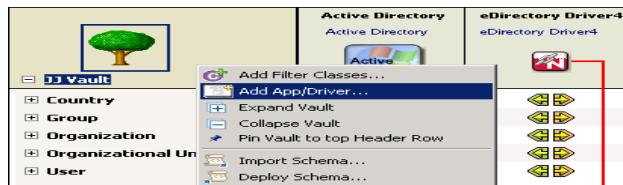
To add an Identity Vault, click the *Add Identity Vault* icon on the toolbar.

To configure the Identity Vault, double-click it.

To delete an Identity Vault, select it, then press Delete.

12.3.2 Adding a Driver in the Dataflow Editor

Figure 12-19 An Identity Vault in the Dataflow Editor

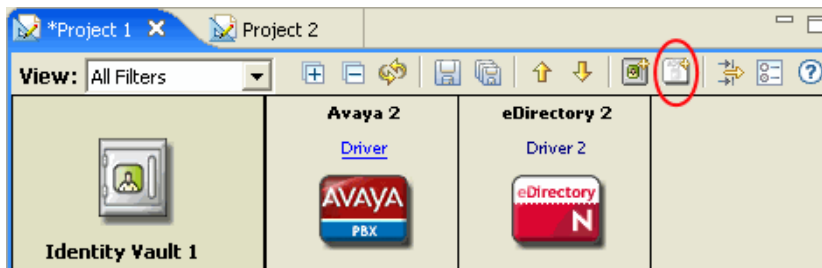


To add a driver while in the Dataflow editor, right-click an Identity Vault, then select *Add App/Driver*.

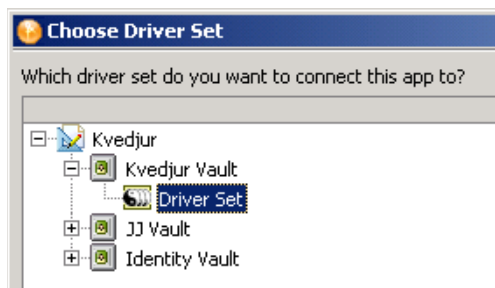
To delete an Identity Vault or driver, select it, then press the Delete key.

12.3.3 Adding an Application in the Dataflow Editor

- 1 On the toolbar, click the *Add Application* icon.



- 2 Browse to and select the driver set that you want this application to connect to, then click *OK*.



- 3 Select the driver you want to create, then click *OK*.



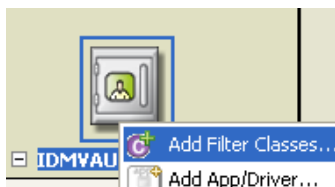
Designer creates a skeleton of the driver. It does not launch the Driver Configuration Wizard. If you want to configure the driver, right-click the connection icon in the Modeler, then select *Run Configuration Wizard*.

12.3.4 Adding Classes and Attributes

You can add classes and attributes to the dataflow.

To add a class:

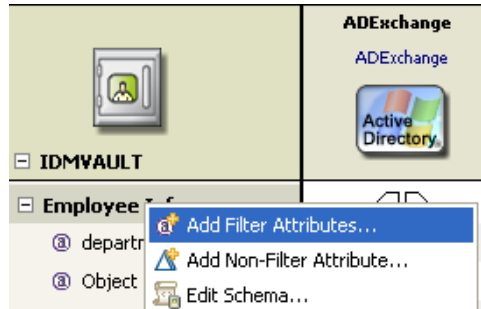
- 1 Right-click an Identity Vault, then select *Add Filter Classes*.



- 2 Select the class that you want to add, then click *OK*.
If you want to add more than one class, press *Ctrl* and select the classes.

To add an attribute:

- 1 Right-click a class, then select *Add Filter Attribute*.



- 2 Select the attribute that you want to add, then click *OK*.

If you want to add more than one attribute, press Ctrl and select the attributes.

12.3.5 Adding Non-Filter Attributes

The Dataflow editor provides non-filter attributes. By default, all classes and attributes in the Dataflow editor come directly from all of the filter policies of the drivers. However, in production environments, it is common to cause data to flow a certain way directly in your Policy Script code, XSLT, or in external code that you call out to.

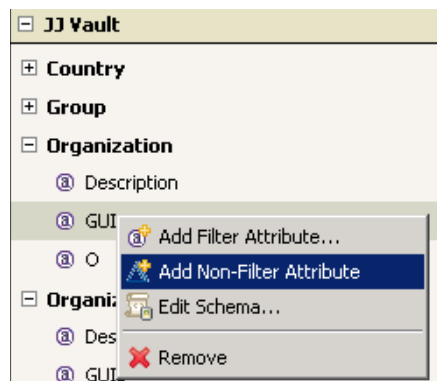
Usually, these non-filter attributes aren't defined in a policy filter (unless you're describing "augmented" processing) and aren't in the schema map. This is because they are generated outside of normal driver operations and you need them in the schema mapping rule only if the engine processes them.

Normally, non-filter attributes are operated on in the Publisher Command Transformation policy set or the Subscriber Output Transformation policy set.

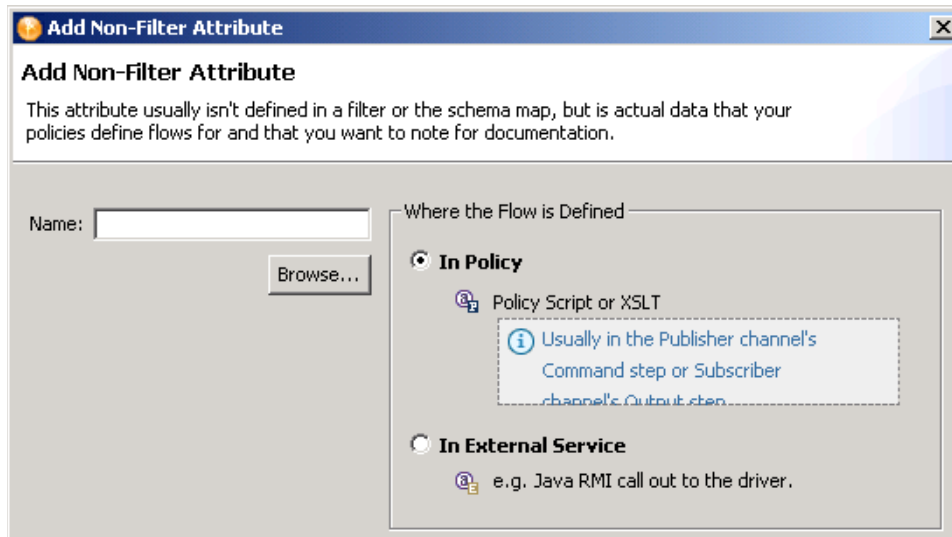
The Dataflow editor lets you add the non-filter attributes to the table for documentation purposes so that you can capture the attributes and have an accurate picture of your actual enterprise dataflows.

To add a non-filter attribute:

- 1 Right-click the class or attribute name, then select *Add Non-Filter Attribute*.



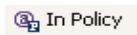
- 2 Specify the name of the attribute or class, or click *Browse*, then browse to and select the attribute or class.



- 3 Click *OK*.
- 4 Select where the flow of the attribute or class is defined.
In Policy: The dataflow is defined in a policy script or an XSLT style sheet.
In External Service: The dataflow is defined in a Java* RMI call to the driver.
- 5 Click *OK*.

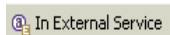
If the non-filter attribute is defined by a policy, a small P is added to the icon. This icon distinguishes a non-filter attribute from a regular filter attribute.

Figure 12-20 A Non-Filter Attribute



If the attribute is defined by an external service, a small E is added to the icon.

Figure 12-21 A Non-Filter External Attribute



12.4 Editing Items

- ♦ [Section 12.4.1, “Editing within the Dataflow Editor,” on page 240](#)
- ♦ [Section 12.4.2, “Editing Non-Filter Attributes,” on page 241](#)
- ♦ [Section 12.4.3, “Managing Schema,” on page 241](#)
- ♦ [Section 12.4.4, “Removing a Flow,” on page 242](#)
- ♦ [Section 12.4.5, “Changing Data Flows,” on page 242](#)

12.4.1 Editing within the Dataflow Editor

As a convenience, you can edit many items within the Dataflow editor. This capability turns the Dataflow editor into a full project editor that allows you to have all the tools you need in one place. You can edit Identity Vault properties, classes, attributes, drivers, and applications.

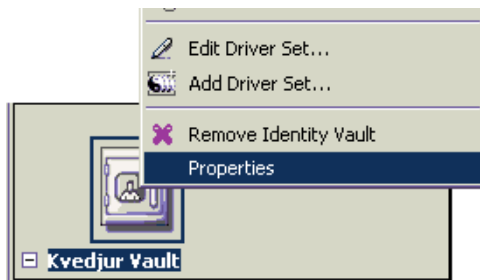
- ◆ “Identity Vault Properties” on page 240
- ◆ “Classes and Attributes” on page 240
- ◆ “Drivers” on page 241
- ◆ “Applications” on page 241

Identity Vault Properties

Access the Identity Vault’s properties pages by doing one of the following:

- ◆ Double-click the Identity Vault.
- ◆ Select the Identity Vault, then press Enter.
- ◆ Right-click the Identity Vault, then select *Properties*.

Figure 12-22 The Properties Option



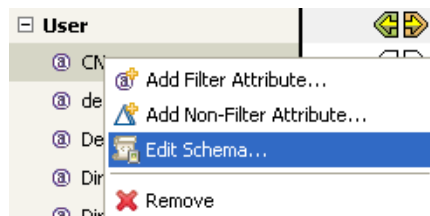
Classes and Attributes

Launch the Manage Schema tool by doing one of the following:

- ◆ Double-click the class or attribute.
- ◆ Select the class or attribute, then press Enter.
- ◆ Right-click the class or attribute, then select *Edit Schema*.

This tool enables you to modify classes and attributes. For more information, see [Chapter 11, “Managing the Schema,” on page 187](#).

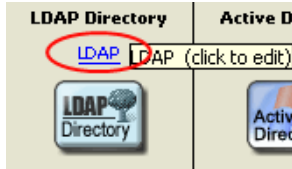
Figure 12-23 The Edit Schema Option



Drivers

To access the driver's property pages, click the driver name below the application name.

Figure 12-24 Location of a Driver Name

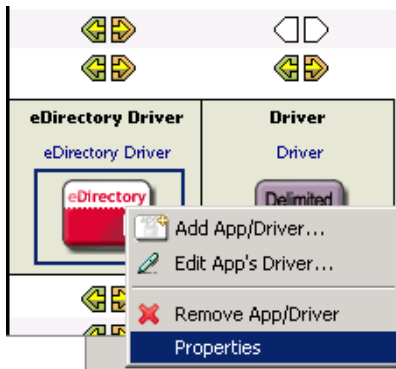


Applications

Access the properties pages for the application by doing one of the following:

- ◆ Double-click the application.
- ◆ Select the application, then press Enter.
- ◆ Right-click the application, then select *Properties*.

Figure 12-25 The Properties Option for an Application



12.4.2 Editing Non-Filter Attributes

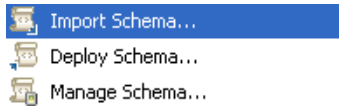
The directional flow of these attributes is edited in the same way as other attributes. Right-click the arrows and select *Publish*, *Subscribe*, *Ignore*, *Reset*, or *Remove from Filter*.

Reset means that you have the value reset under certain conditions. The attribute might be in a policy filter, but in addition, you might have some manual logic that resets the value. Occasionally, resets by manual logic occur in production environments.

12.4.3 Managing Schema

To import, deploy, and edit the schema in the Dataflow editor, right-click an Identity Vault. All schema changes made outside of this editor are synchronized. For more information, see [Chapter 11, "Managing the Schema," on page 187](#).

Figure 12-26 The Manage Schema Option

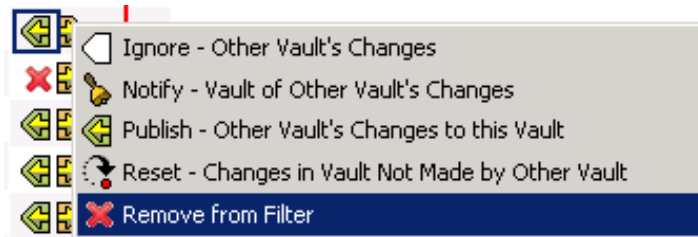


12.4.4 Removing a Flow

If a particular flow (Publisher or Subscriber channel) is not defined in the policy filter's XML, a red X replaces the Publisher or Subscriber channel arrow. This means that it's not in the policy and there will be no flow. This scenario is essentially the same as an Ignore Flow icon, which is an empty white arrow. However, the distinction is useful so that you know what is actually in your policy's XML.

To remove the flow from the XML:

- 1 Right-click the Publisher or Subscriber channel icon.
- 2 Select *Remove from Filter*.



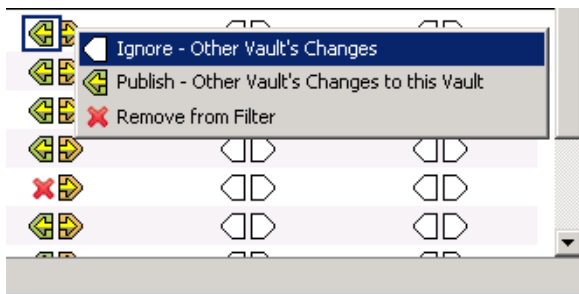
If a class or attribute is marked to be removed on both channels and nothing references it, Designer removes it from the Dataflow editor's table.

12.4.5 Changing Data Flows

To change the way data flows, right-click the arrow that displays the dataflow, then select the option that you want.

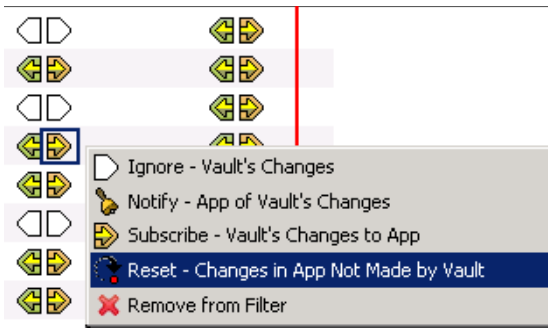
The Publisher channel is on the left. The Subscriber channel is on the right.

Figure 12-27 Changing the Subscriber Flow



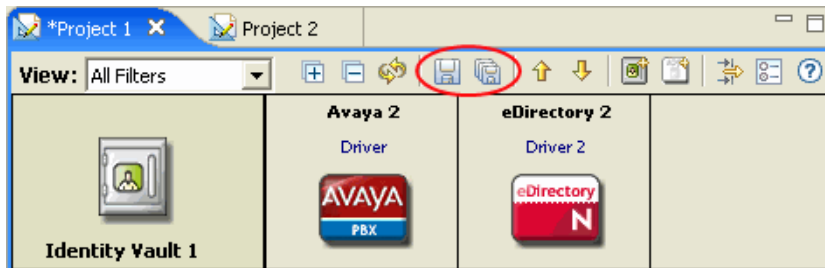
The *Reset-Changes in App Not Made by Vault* option is available when you select an attribute.

Figure 12-28 The Option to Reset Changes



12.5 Generating HTML Reports

- 1 Click the *Save Current View to HTML* icon or the *Save All Views to HTML* icon.

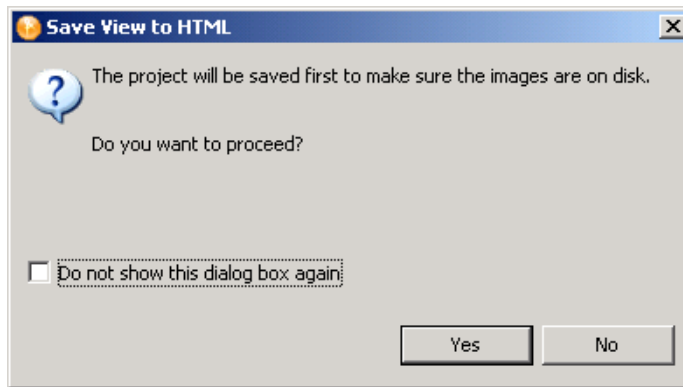


Save Current View to HTML generates a report for the current view. *Save All Views to HTML* generates nine reports.

Name	Modified	Size	Type
icons	12/8/2006 12:15 PM		File Folder
DataFlowAll.html	12/8/2006 12:29 PM	1 KB	HTML File
DataFlowFilter.html	12/8/2006 12:15 PM	1 KB	HTML File
DataFlowNonFile...	12/8/2006 12:15 PM	1 KB	HTML File
NotifyFlowAll.html	12/8/2006 12:15 PM	6 KB	HTML File
PasswordFlowAll...	12/8/2006 12:15 PM	7 KB	HTML File
ResetFlowAll.html	12/8/2006 12:15 PM	1 KB	HTML File
SyncFlowAll.html	12/8/2006 12:15 PM	101 KB	HTML File
SyncFlowFilter.html	12/8/2006 12:15 PM	101 KB	HTML File
SyncFlowNonFile...	12/8/2006 12:15 PM	1 KB	HTML File

The HTML files are automatically named. The descriptive names tell what the report is. All images that you need to support the HTML document are copied to an `icons` subdirectory where the HTML is saved. The process includes all of your custom application icons.

You are prompted to save the project to disk.



Saving is necessary to make sure that all of your icon information is in a state where it can be successfully copied.

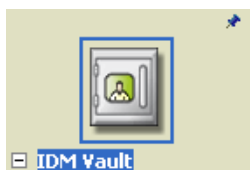
- 2 Click *Yes* to save the project.
- 3 Browse to and select the location where you want to save the reports, then click *OK*.

The directory you select for saving is stored in Designer's memory and becomes the default directory the next time you save.

- 4 Click *OK* in the Information dialog box that indicates where the report is saved.

If you pin an Identity Vault and then generate a report, the report is for that Identity Vault. The Identity Vault's name is included in the HTML name.

Figure 12-29 A Pinned Identity Vault




If the Dataflow editor has multiple applications, Designer wraps the table so that the applications appear as Continuation #1, Continuation #2, etc. in the report for the pinned Identity Vault.

Figure 12-30 Continuation Rows in a Report

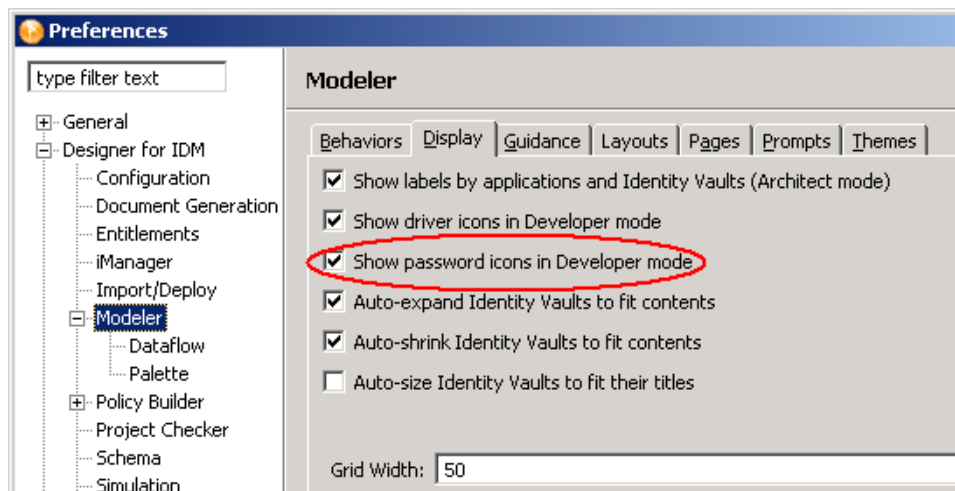
Kvedjur Vault	Driver Driver Delimited Text	UserApplication UserApplication	eDirectory Driver4 eDirectory Driver3	Avaya Avaya Driver AVAYA PBX	Delimited Text Text Driver Delimited Text	Generic Ap Generic Drive app
Country			↔			
③ c			↔			
③ Description			↔			
③ GUID			✖			
Group			↔			
③ CN			↔			
③ Description			↔			
③ GUID			✖			
Kvedjur Vault Continuation #2	LDAP Directory LDAP Driver LDAP Directory	Linux Linux/Unix Driver Linux	LoopBack LoopBack Driver Loopback	SAP HR SAP-HR Driver SAP HR		
Country						
③ c						
③ Description						
③ GUID						
Group						

12.6 Integrating Passwords

If a driver is synchronizing passwords (in at least one direction), a small password-field icon  appears under the driver icon. This icon enables you to know where passwords are being synchronized.

You can toggle this icon on or off in Preferences.

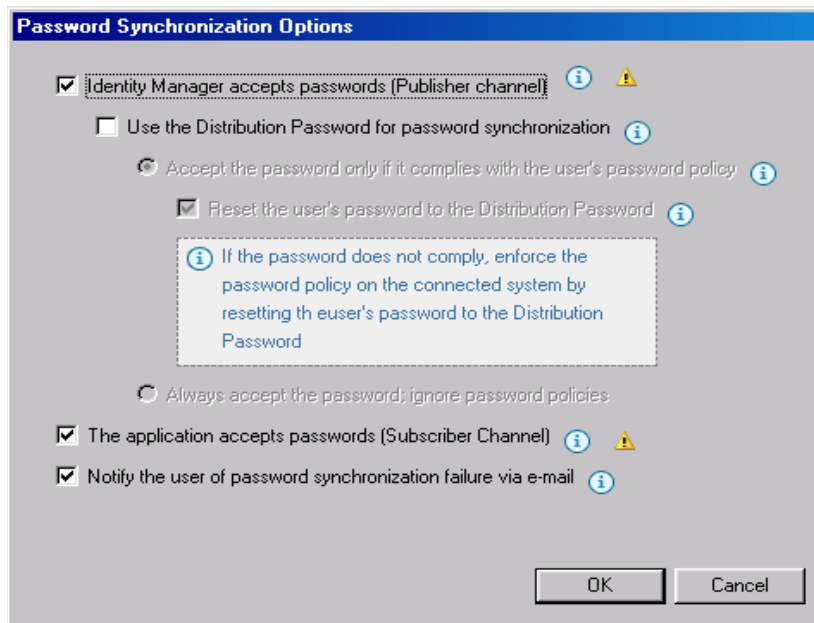
- 1 Select *Window > Preferences > Designer for IDM > Modeler*.
- 2 Click the *Display* tab.
- 3 Select or deselect *Show password icons in Developer mode*.



If you hover the mouse over the password icon in Developer mode, a helpful tip explains how your passwords are flowing for each server involved in the flow.

To configure the flow of password synchronization:

- 1 In Dataflow mode, select *Password Sync* in the *View* drop-down box.
- 2 Double-click the flow arrow.
- 3 Select options, then click *OK*.



Creating and Managing Policies

13

Policies enable you to customize the flow of information into and out of Novell® eDirectory™ for a particular environment.

For example, one company might use the inetorgperson as the main user class, and another company might use User. To handle this, a policy is created that tells the Metadirectory engine what a user is called in each system. Whenever operations affecting users are passed between connected systems, Identity Manager applies the policy that makes this change.

Policies also create new objects, update attribute values, make schema transformations, define matching criteria, maintain Novell Identity Manager associations, and many other things.

For more information on policies, refer to the following:

- ◆ *Understanding Policies for Identity Manager 3.5.1*
- ◆ *Policies in Designer 2.1*
- ◆ *Novell Credential Provisioning Policies for Identity Manager 3.5.1*
- ◆ *Identity Manager 3.5.1 DTD Reference*

Setting Up E-Mail Notification Templates

14

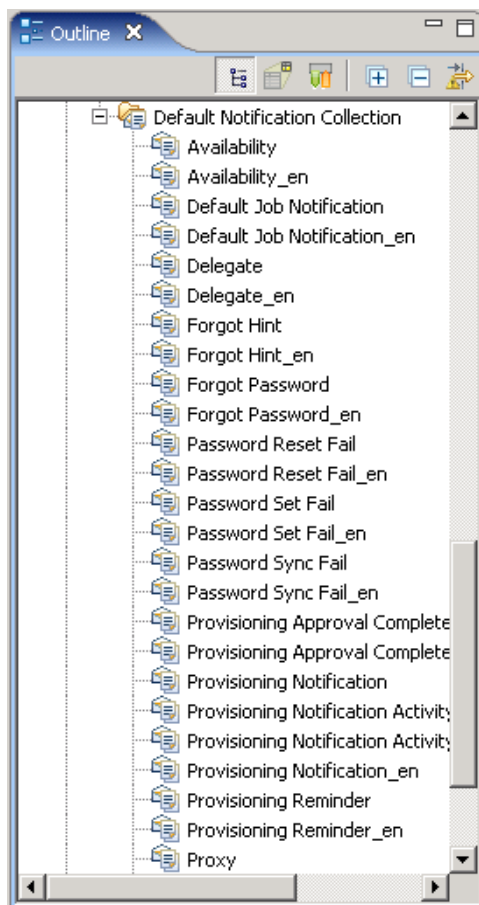
Notification templates enable you to customize and send e-mail messages that users receive when triggers occur.

- ♦ [Section 14.1, “Viewing Notification Templates,” on page 249](#)
- ♦ [Section 14.2, “Editing a Notification Template,” on page 251](#)
- ♦ [Section 14.3, “Adding a Notification Template,” on page 255](#)
- ♦ [Section 14.4, “Policy Builder and Notification Templates,” on page 256](#)
- ♦ [Section 14.5, “Configuring the E-Mail Server,” on page 256](#)

14.1 Viewing Notification Templates

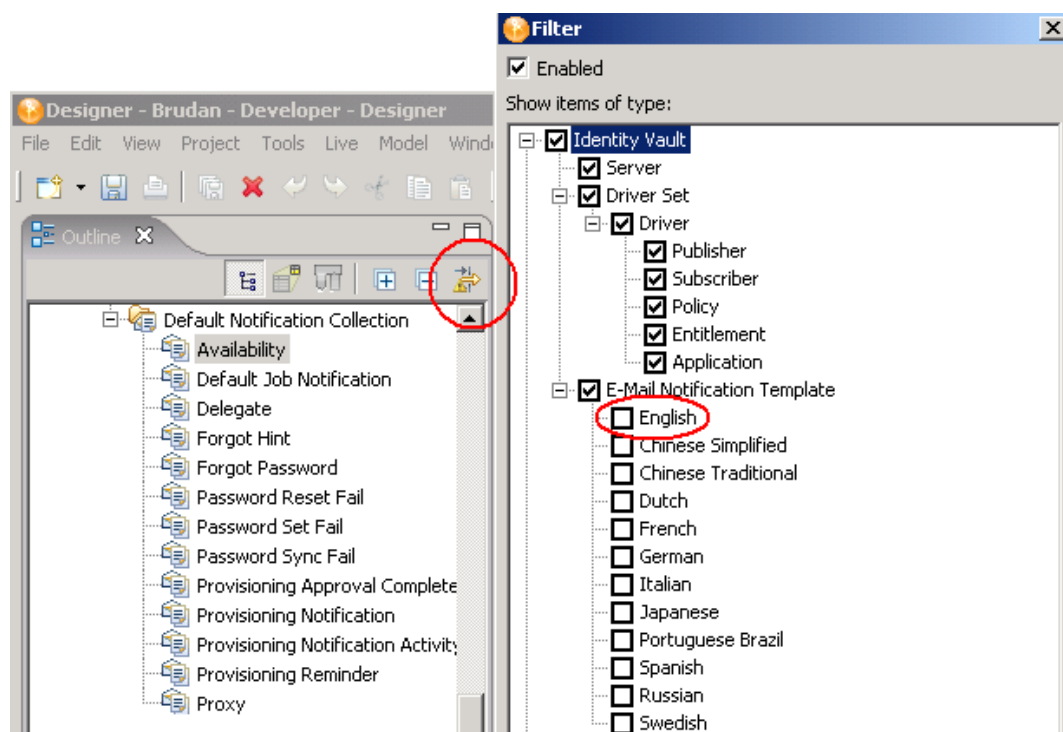
Designer provides default notification templates, which you can view or edit.

- 1 Select an Identity Vault in the Modeler.
- 2 In the Outline view, scroll to the *Default Notification Collection* for that Identity Vault.



If *Default Notification Collection* isn't expanded, expand it. The expanded tree shows the notification templates. If you selected *English* as the language option during the install, the expanded tree view lists an *_en* item and a *non_en* item for each template.

The install program installs all of the notification templates with Designer. However, templates are filtered out of the Outline view based on your workstation's locale settings. To view and edit the internationalized template files, click the *Filter* button in the Outline view, then select languages that you want to see. To eliminate the duplicate (*_en*) set of templates, deselect *English* in the filter.



You use the templates in the *Default Notification Collection* to send e-mail notifications to users in the *Identity Vault*. You can customize these templates with your own text.

3 Right-click a template (for example, *Forgot Hint*), then select *Edit*.

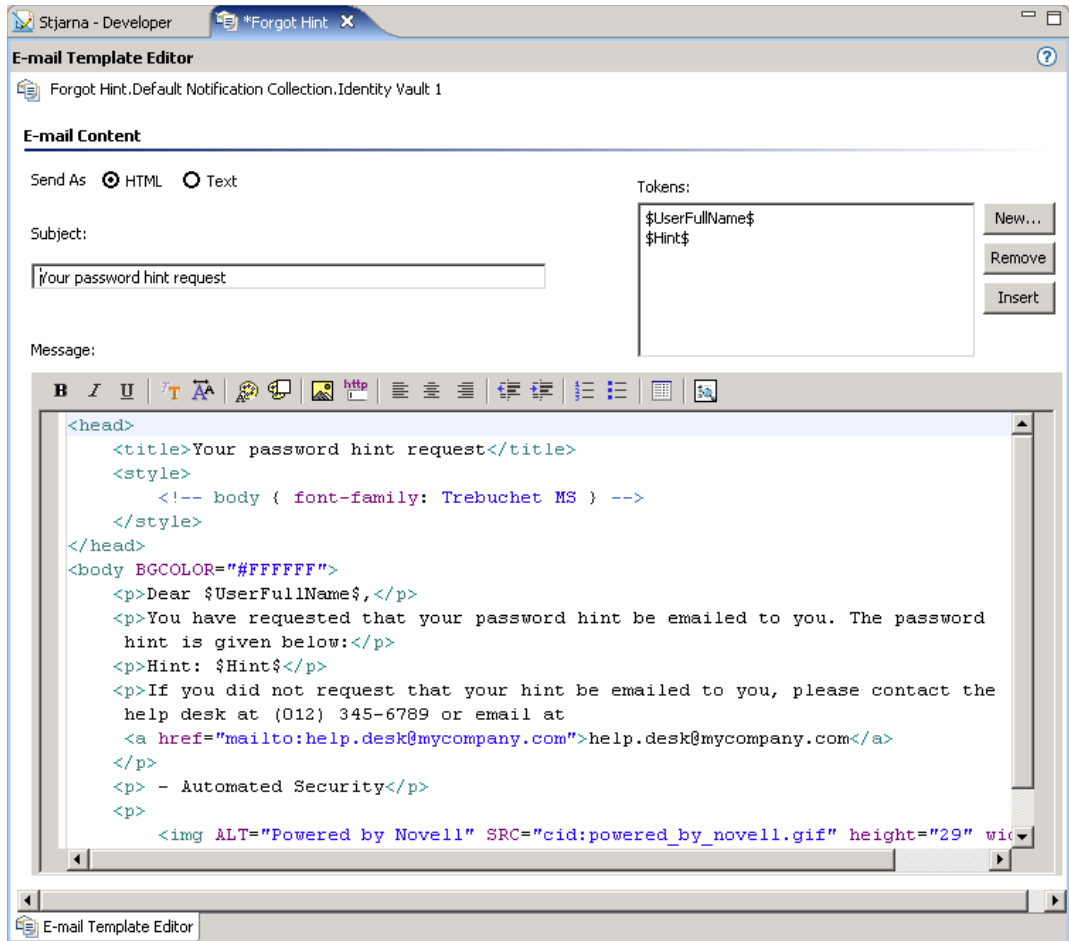
You can also open a template by double-clicking it.

Template Name	Description
Availability	Sends an e-mail notification when an availability setting has been created or modified.
Default Job Notification	Sends an e-mail notification to report results of the job as configured in the template. Contains the name of a job and any status information from the job.
Delegate	Sends an e-mail notification when a a delegate assignment has been created or modified
Forgot Hint	Sends an e-mail notification when a user forgets a password and requests a hint.

Template Name	Description
Forgot Password	Sends an e-mail notification when a user incorrectly enters a password.
Password Reset Fail	Sends an e-mail notification when a user tries to reset a password but doesn't meet password policy requirements.
Password Sync Fail	Sends an e-mail notification when a user's password fails to synchronize.
Provisioning Approval Completed Notification	Sends an e-mail notification when a workflow is completed. Indicates the overall workflow and provisioning decision.
Provisioning Notification	Sends an e-mail notification to a user or manager approval. Indicates that action is required from the user or manager.
Provisioning Reminder	Sends an e-mail notification when a user activity timeout expires. Reminds the user or manager to act.
Proxy	Sends an e-mail notification when a proxy assignment has been created or modified.

14.2 Editing a Notification Template

- 1 Select an Identity Vault.
- 2 In the Outline view, right-click a template (for example, *Forgot Hint*), then select *Edit*.
- 3 Select a format, specify a subject, add tokens, customize the message that users receive, then save and close the template.
 - ♦ [“Selecting a Format” on page 252](#)
 - ♦ [“Specifying a Subject” on page 252](#)
 - ♦ [“Working with Tokens” on page 253](#)
 - ♦ [“Editing a Template Message” on page 254](#)



Selecting a Format

Select whether users receive this e-mail notification in HTML or text format.

Figure 14-1 Options for Sending the Notification

Send As HTML Text

Specifying a Subject

The subject is the text that a user views in an e-mail's Subject headline or field. You can change the text in the Subject field. You can also use tokens here. The text or tokens don't determine the name of the template.

Figure 14-2 The Subject Field

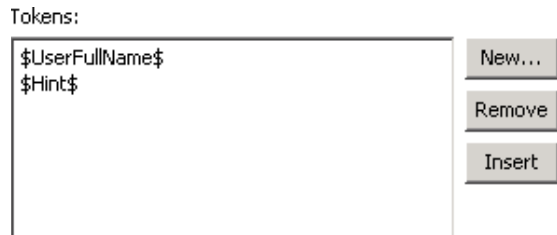
Subject:

Your password hint request

Working with Tokens

A token is a variable or replacement tag for items such as the user's name. Tokens help you personalize the message to the user.

Figure 14-3 The Tokens Dialog Box

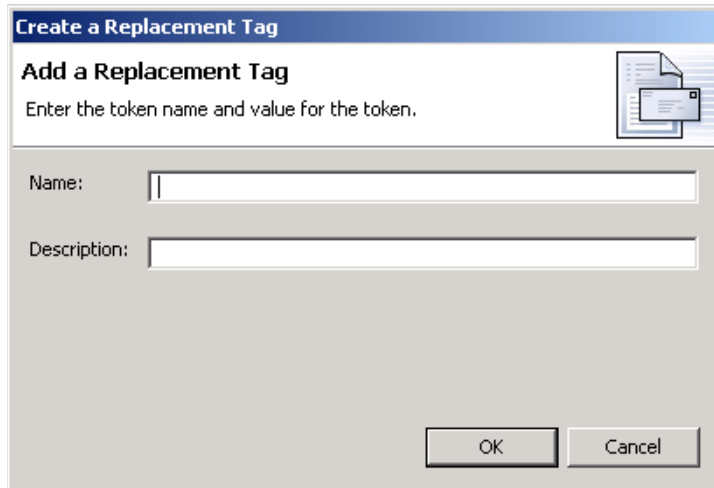


Each template includes default tokens. For example, the Forgot Password e-mail template for sending a password to the user includes the default replacement tag named `$CurrentPassword$`.

You can define other tokens for use in the body of the message or in the subject. Your ability to do so depends on the application that uses the templates. To find out how to define additional replacement tags, see the documentation for the application. For example, Identity Manager Password Synchronization can't use a replacement tag that you create unless the policy in the driver configuration that uses the template also contains the definition of the replacement tag.

Adding a Token

- 1 Click *New*.



- 2 In the *Create a Replacement Tag* dialog box, type a name for the token.
You don't need to type the \$ characters. Designer provides them.
- 3 Type a description for the token.
- 4 Click *OK*.

When you add a token, the tag is automatically added to the XML source for the template. After you add a tag, you can edit it only in the XML Source view.

Removing a Token

To delete a token, select it, then click *Remove*.

Make sure that you don't remove tags that are needed for the body of the message.

Inserting a Token

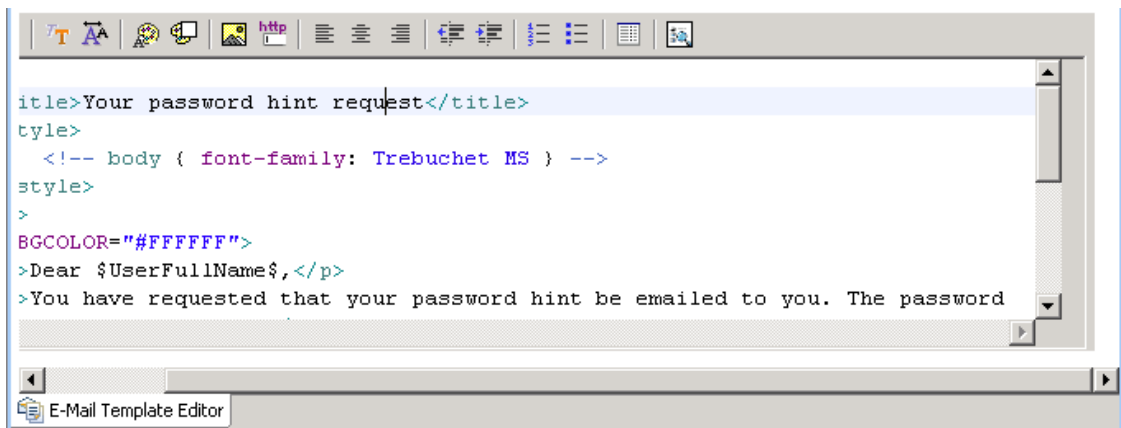
- 1 In the template, click where you want to insert a token.
- 2 Select a token.
- 3 Click *Insert*.

Designer inserts the selected token into the e-mail template.

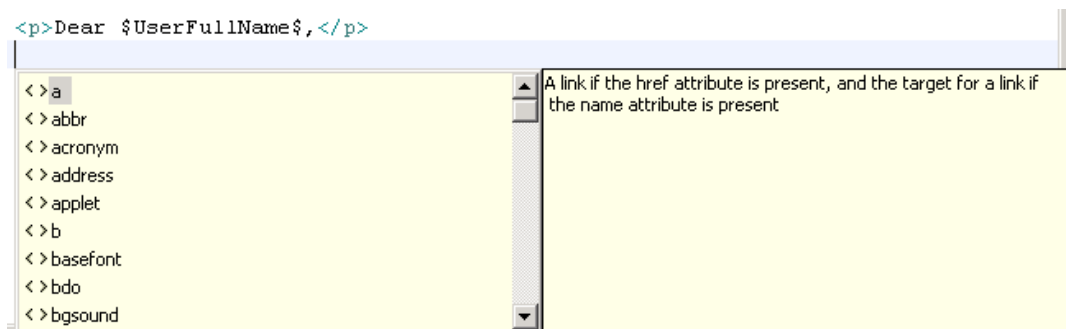
Editing a Template Message

The text of the e-mail message appears in the *Message* field. Customize the text so that it suits your environment. Use tokens to personalize the e-mail message.

Figure 14-4 *The Message Edit Box*




- 1 In the *E-Mail Template Editor*, place your cursor in the Message edit box, then press Ctrl+Spacebar.
- 2 Select an HTML tag by double-clicking a tag from the drop-down list.



- 3 Format text by using the toolbar.



4 Preview the text by clicking the Preview icon .

5 Save the template by selecting *File > Save*.

You can also click the *Save* icon.

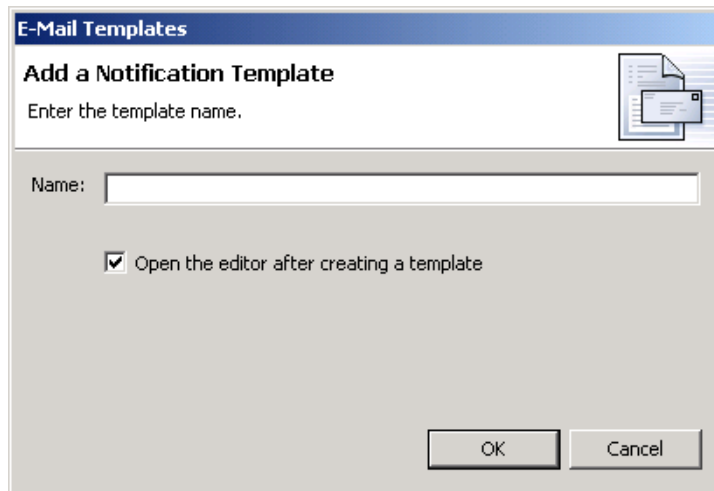
If the code isn't valid, you can't save the template.

14.3 Adding a Notification Template

1 Select an Identity Vault in the Modeler.

2 In the Outline view, scroll to *Default Notification Collection* for that Identity Vault.

3 Right-click, then select *New Template*.



4 Name the template.

5 If you want to automatically open the template editor so that you can view or edit the template, select *Open the editor after creating a template*.

6 Click *OK*.

7 Customize the text by editing the template message.

Importing a Notification Template

To import a notification template from a file:

1 In the Outline view, scroll to *Default Notification Collection* for an Identity Vault.

2 Right-click, then select *Import Template from File*.

3 Browse to and select the template.

4 Customize the text for your environment by editing the template message.

To import a notification template as a live operation:

1 In the Outline view, scroll to *Default Notification Collection* for an Identity Vault.

2 Right-click, then select *Live > Import*.

3 Specify the host name (IP address).

- 4 To authenticate, specify the user name and password.
- 5 Browse to and select the template, then click *OK > Continue > Import*.
- 6 Customize the text for your environment by editing the template message.

Deploying a Template

After you add or import a template, deploy it.

- 1 Customize the text for your environment by editing the template message.
- 2 Right-click the template.
- 3 Select *Live > Deploy*.

14.4 Policy Builder and Notification Templates

For information on using the Policy Builder interface to send e-mail notifications, see “[Send Email](#)” and “[Send Email from Template](#)” in the *Policies in Designer 2.1* guide.

14.5 Configuring the E-Mail Server

The e-mail server sends notification e-mails from applications that use the Notification Configuration templates.

- 1 Select an Identity Vault in the Modeler.
- 2 In the Outline view, scroll to *Default Notification Collection* for that Identity Vault.
- 3 Right-click, then select *Properties*.
- 4 Specify the host name, *From*, and authentication settings for your SMTP e-mail server.

Host Name: The host name of your SMTP e-mail server. This can also be an IP address.

From: When a user opens the e-mail, the text that you enter in the *From* edit box is displayed in the *From* field of the user’s e-mail header. Depending on your mail server settings, the text in this field might need to match a valid sender in the system (for example, `helpdesk@company.com` instead of descriptive text such as The Password Administrator). Such a match allows the mail server to do reverse lookups or authentication.

Authenticate to the server by using credentials: Use for a secured SMTP server.

If your server requires authentication before sending e-mail, specify the username and password here.

Although the authentication information is specified here, you might also need to specify it separately for the application that is sending the notification e-mails.

For example, Forgotten Password e-mail notifications can be sent by using the authentication information you specify here. However, notification e-mails for Identity Manager Password Synchronization require the authentication information to be provided in the driver policy that is used to send notification e-mails.

- 5 Click *OK*.

Designer's Import feature allows you to import the following items into defined projects:

- ◆ Projects from the File System or from the Identity Vault (eDirectory™)
- ◆ Driver sets
- ◆ Individual drivers
- ◆ Channels
- ◆ Policies
- ◆ Schema

Depending on a project's complexity, importing can save you time in building and rebuilding drivers, channels, and policies. For instance, after a driver, channel, or policy is built, you can import it into new projects and modify it to run in the new environment instead of starting from scratch on each new driver, channel, or policy.

You import projects, drivers, channels, schema, and policies from an existing eDirectory tree running the Identity Manager system (Identity Vault) or from an exported project located in the file system. In Designer, use the Deploy feature to save drivers, channels, and policies into a Metadirectory server in an Identity Vault. Use the Export feature to save project, drivers, channels, and policies to a local, removable, or network directory.

What you can import depends on the level you are at within a project. For instance, at the *File > Import* level, you can import a project from the file system or from an Identity Vault. At the project level, you can import driver sets, drivers, channels and policies; at the driver level, you can import channels and policies, and at the channel level, you can import policies.

During import, Designer does not import anything that is encrypted. This includes named passwords, e-mail notifications, existing certificates, and the Secure Login administrator password.

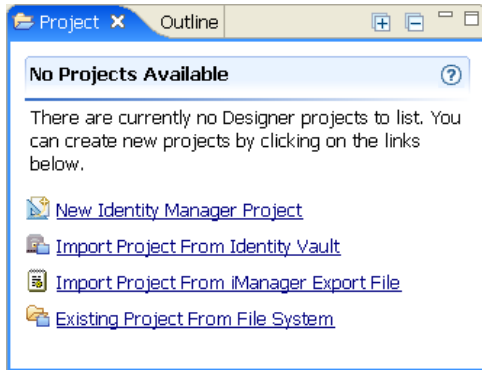
- ◆ [Section 15.1, "Importing Projects," on page 257](#)
- ◆ [Section 15.2, "Importing a Driver Set or Driver from the Identity Vault," on page 266](#)
- ◆ [Section 15.3, "Importing a Driver Configuration File," on page 275](#)
- ◆ [Section 15.4, "Importing Channels, Policies, and Schema Items from the Identity Vault," on page 281](#)
- ◆ [Section 15.5, "Using the Compare Feature When Importing," on page 291](#)
- ◆ [Section 15.6, "Error Messages and Solutions," on page 298](#)

15.1 Importing Projects

Designer's Import feature allows you to import projects from the File System or from an Identity Vault (eDirectory). What you are able to import depends on the level you are at within a project. To import an eDirectory object, you must have sufficient rights to access the eDirectory tree that is associated with the Identity Vault you are designing.

When you initially open the Designer utility and close the Welcome view, you have no projects that are currently available:

Figure 15-1 Initial No Projects Available View



For information on creating a new project, see [Chapter 6, “Creating a Project,”](#) on page 49. For information on importing projects, see the following sections:

- ◆ [Section 15.1.1, “Importing a Project from the Identity Vault,”](#) on page 258
- ◆ [Section 15.1.2, “Importing a Project from the File System,”](#) on page 262
- ◆ [Section 15.1.3, “Importing a Project from iManager,”](#) on page 264

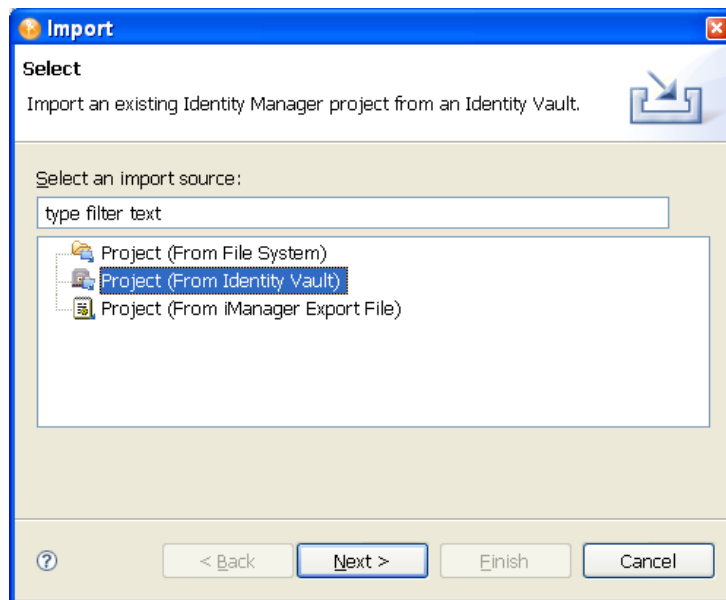
15.1.1 Importing a Project from the Identity Vault

1 In Designer, click *File > Import*.

or

Click *Import Project From Identity Vault* from the No Projects Available page in the Projects view, then skip to [Step 3](#).

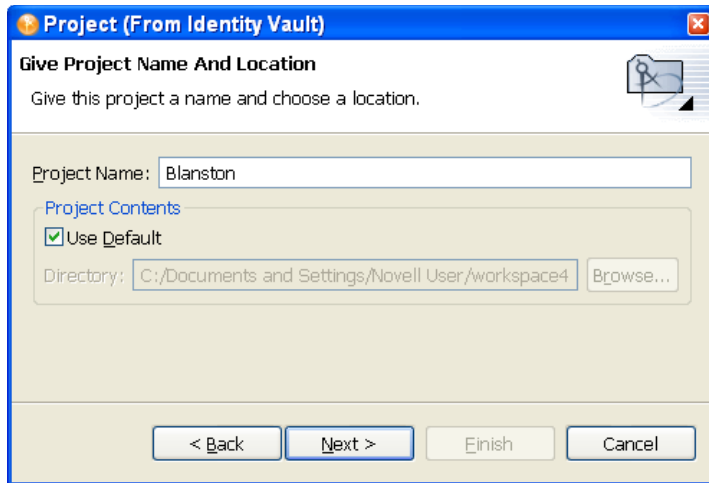
2 From the Import window, select *Identity Manager Project (From Identity Vault)* under the Designer for Identity Manager heading. Click *Next*.



The Import window allows selections under a number of tabs, including Designer for Identity Manager. Selections under the Designer for Identity Manager tab include:

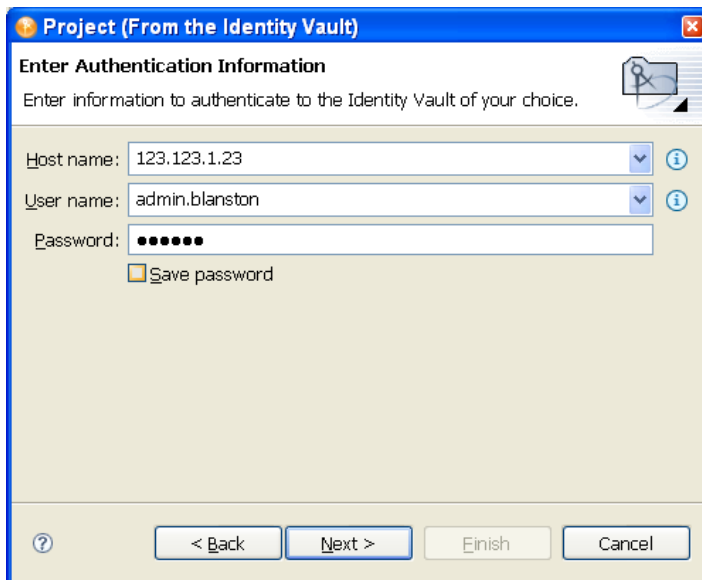
- ♦ Importing an existing Identity Manager project from the file system (the project must have a valid `.project` file).
- ♦ Importing an existing Identity Manager project from an Identity Vault.
- ♦ Importing an existing Identity Manager project from an iManager export file (Driver Set or Driver).

- 3 In the Project (From Identity Vault) window, give the new project a name. Select where the contents of this project are to reside (for Windows workstations, the default is `C:\Documents and Settings\Username\workspace`). Click *Next*.



The screenshot shows a dialog box titled "Project (From Identity Vault)" with a sub-header "Give Project Name And Location". The instruction reads: "Give this project a name and choose a location." The "Project Name" field contains "Blanston". Under "Project Contents", the "Use Default" checkbox is checked. The "Directory" field shows "C:/Documents and Settings/Novell User/workspace4" with a "Browse..." button next to it. At the bottom, there are four buttons: "< Back", "Next >", "Finish", and "Cancel".

- 4 In the Import Project From Identity Vault window, specify the information needed to authenticate to the Identity Vault (eDirectory) of your choosing. In the *Host Name* field, you can use either a tree name or the IP address of the Identity Vault.



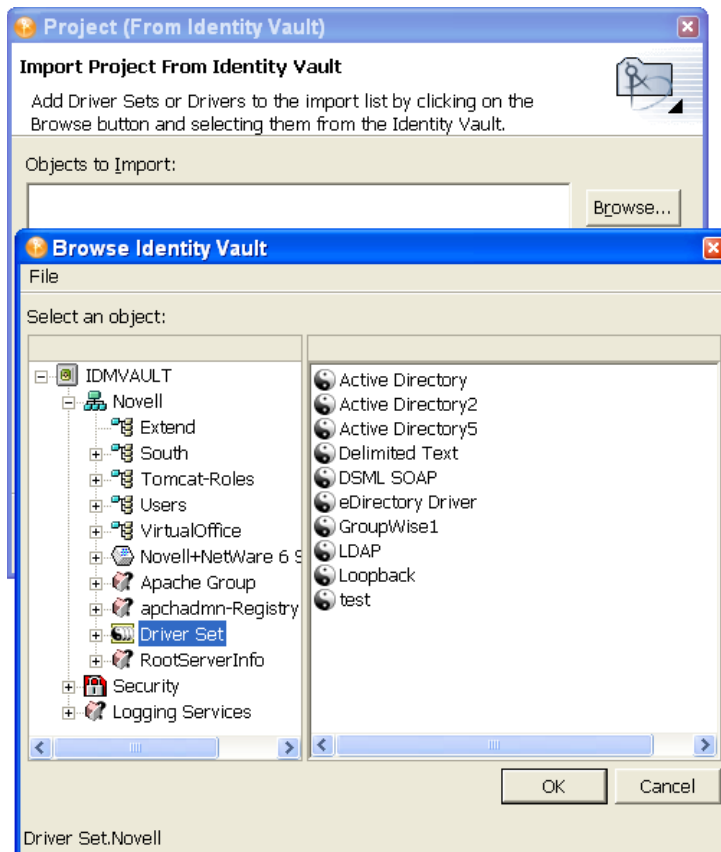
The screenshot shows a dialog box titled "Project (From the Identity Vault)" with a sub-header "Enter Authentication Information". The instruction reads: "Enter information to authenticate to the Identity Vault of your choice." The "Host name" field contains "123.123.1.23". The "User name" field contains "admin.blanston". The "Password" field is masked with "••••••". There is a "Save password" checkbox which is currently unchecked. At the bottom, there are four buttons: "?", "< Back", "Next >", "Finish", and "Cancel".

- 5 Fill in the *User Name* and the *Password* fields.

If you select *Save Password*, Designer remembers your password. Otherwise, the next time you close Designer, you need to re-authenticate when you import, deploy, or compare Designer objects with the Identity Vault.

You can use history drop-down lists to choose a previously entered value from a list.

- 6 Click *Next*.
- 7 In the Import Project From Identity Vault window, type the Distinguished Name of the driver set or driver you want to import, or click the *Browse* button to select the object within eDirectory. Click *OK*, then click *Finish*.

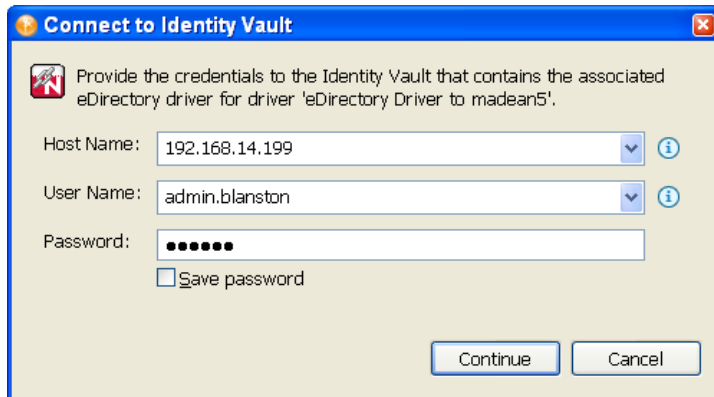


You can import multiple driver sets during the import operation. Just browse to the various objects that you want to import and add them to the list.

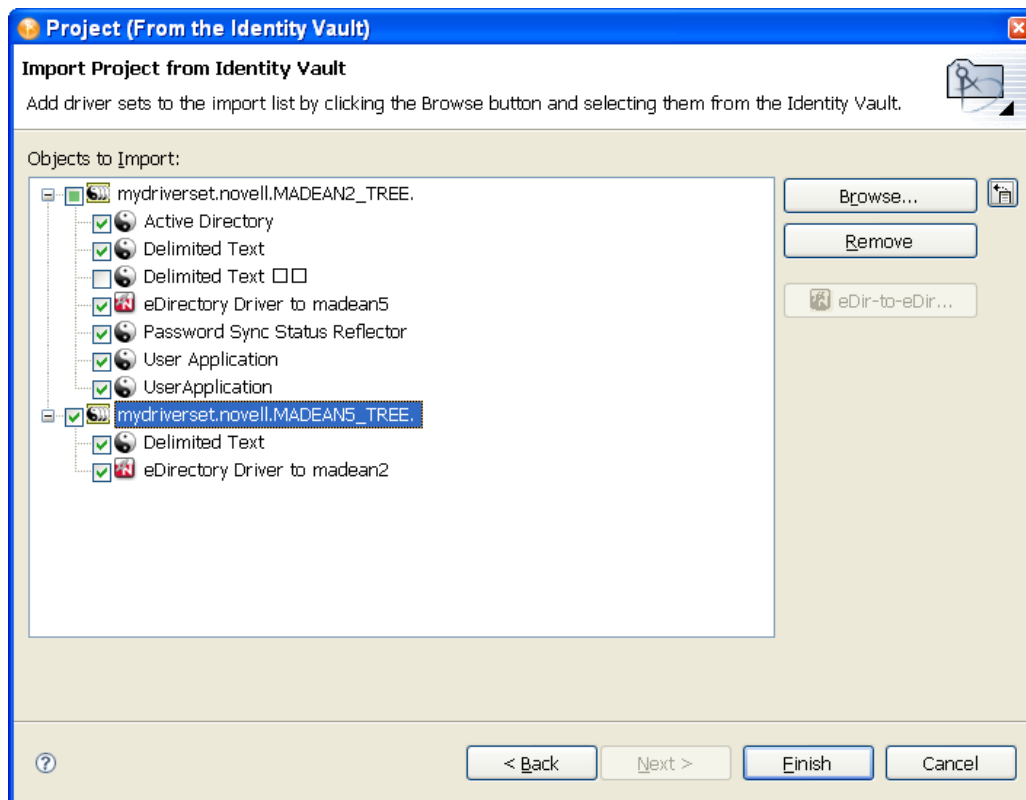
Driver sets that are not associated with a server have a red minus sign in the lower right portion of the driver set icon. These driver sets need a server association in order to be deployed.

- 8 (Conditional) If you are importing more than one eDirectory driver, select the eDirectory driver in the Objects to Import window and click the *eDir-to-eDir* button to display the Connect to Identity Vault window, where you can import the associated driver in the other eDirectory trees. Novell[®] recommends that you import both eDirectory drivers, especially if you have SSL/TLS

enabled. Provide the user name and password, then click *Continue*. (Skip this step if you only want to import one eDirectory driver.)

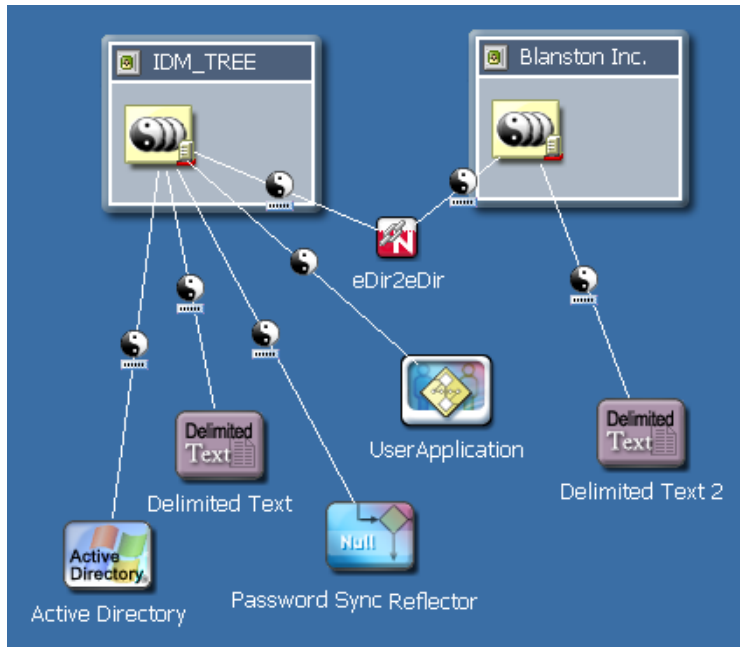


- 9 (Conditional.) If you specify the user name and password and select *Continue* to import both eDirectory drivers, you then see a Browse Identity Vault window where you select the corresponding eDirectory driver. Select the driver and click *Finish*.



You are returned to the Project (From Identity Vault) window, where you can select or deselect the drivers, allowing you to choose only the drivers in a driver set that you want to import.

- 10 Click *Finish*. By having both drivers available, you can view the complete data flow between the two eDirectory drivers, as well as the other drivers you selected.



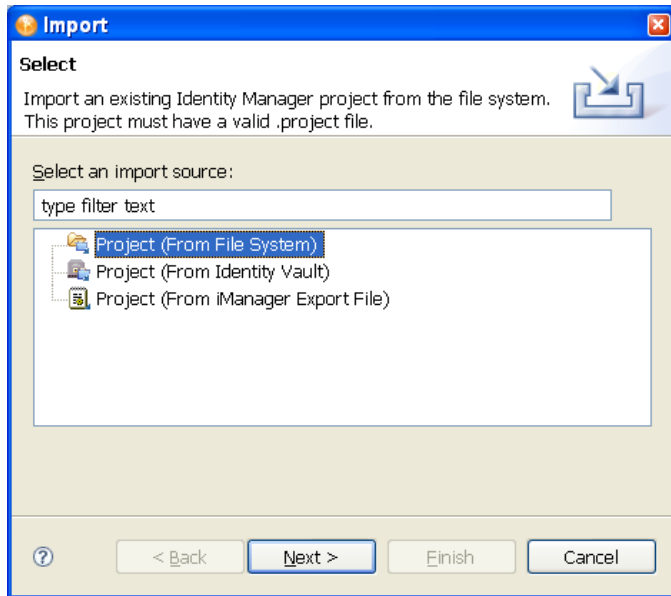
15.1.2 Importing a Project from the File System

- 1 In Designer, click *File > Import*.

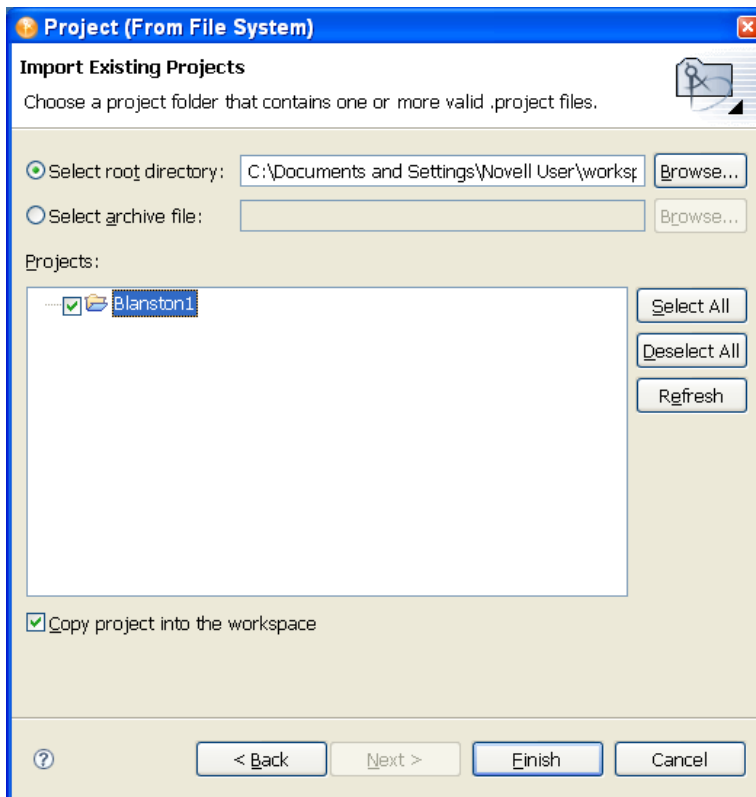
or

Click *Existing Project From File System* from the No Projects Available page in the Project's view, then skip to [Step 3](#).

- From the Import window, select *Designer for Identity Manager > Project (From File System)*, then click *Next*.



- Type the project name or browse to the directory where the project was saved.



- Select the directory where the exported project is saved with the Project extension.

There must also be a valid Project file in the selected directory, or the project does not display in the Projects window. If you have multiple projects you want to import and they are under the same directory, select the directory above the projects and Click *OK*. In the Projects window, you can select or deselect any of the projects, then click *Finish*.

- 5** Make sure the *Copy Project into Workspace* option is selected to copy the contents of the project into the workstation's local workspace.

Earlier versions of Designer did not copy the contents into the workspace, but pointed to the location where the project was imported. Any changes made to the project were saved to the imported location and not to the local workspace, which caused some confusion because all other imports were copied into the local workspace.

You can also use the *Import Project from File System* feature to import a project from another workspace.

You can also import multiple projects at the same time by entering the base or root directory where you want to start your search. The Import Wizard searches the selected directory and all of its subdirectories for valid Designer projects to import. You can then select the projects that you want to import into Designer by using the check boxes. If a project by the same name already exists in Designer, you can't import that project and you won't be able to select it from the list.

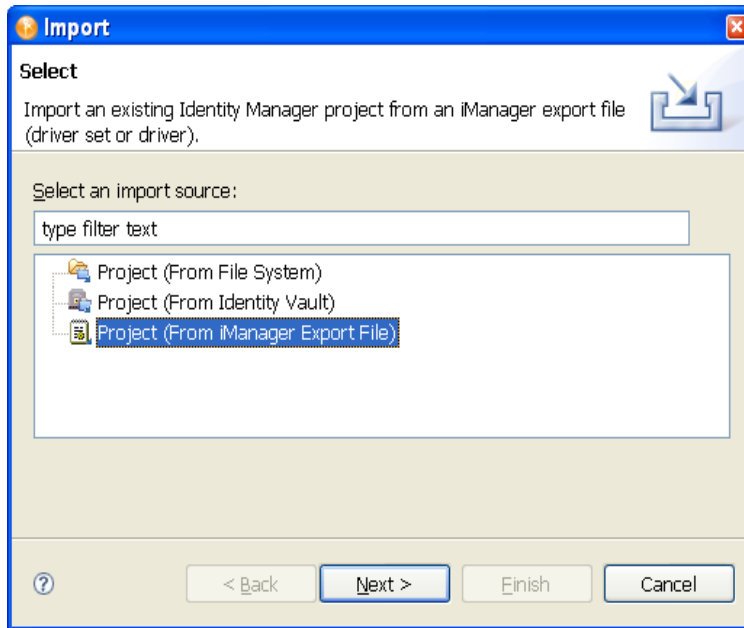
- 6** You can also import projects from ZIP or TAR archives. Click the *Select archive file* selection and select the directory where the exported project is saved with the *.zip* or *.tar* extension. The whole archive is searched for valid Designer projects to import.

15.1.3 Importing a Project from iManager

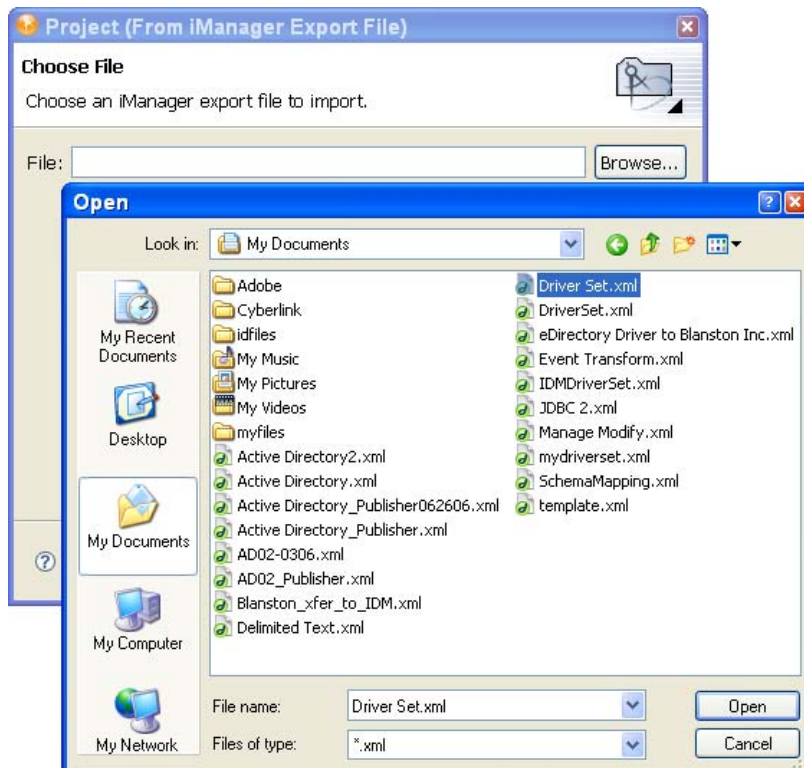
You can create a new Designer Project by selecting an iManager *.xml* export file. These include driver set and driver exports as well as Novell sample configuration files.

- 1** Click *File > Import > Designer for Identity Manager > Project (From iManager Export File)*.
or

Click *Project (From iManager Export File)* from the No Projects Available view, then skip to **Step 3**.



- 2 Type a project name. Use the default workspace directory for the project, or type or browse to the directory where you want to import the project. Click *Next*.



- 3 Select the directory where the exported project is saved with the .xml extension, click *Open*, then click *Finish*.
- 4 When importing a driver set or driver into a project, you are asked if you want to validate the values within the drivers you are importing. If you do not want to validate the driver configuration at this time, click *No*.

Otherwise, click *Yes* and continue importing the project.

You can only import the driver functionality that you saved to the .xml file. This file does not contain default driver configurations unless that is what you have saved.

- 5 Fill out the information requested in all of the Import Information Requested windows that you see for each driver in the driver set, or for each driver selected. The Import Information Requested windows contain different driver information from each selected driver. Click *Next* or *Finish* (depending on the number of pages presented).
- 6 Click *OK* to close the Import Configuration Results window.

15.2 Importing a Driver Set or Driver from the Identity Vault

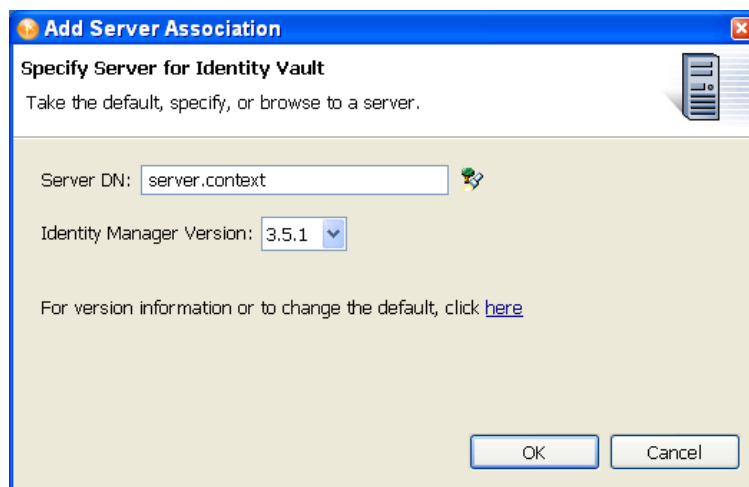
To import an eDirectory object, you must have access to the eDirectory tree that is associated with the Identity Vault.

- [Section 15.2.1, “Associating a Server to the Identity Vault,” on page 266](#)
- [Section 15.2.2, “Importing a Driver Set from the Identity Vault,” on page 268](#)
- [Section 15.2.3, “Importing a Driver from the Identity Vault,” on page 273](#)

15.2.1 Associating a Server to the Identity Vault

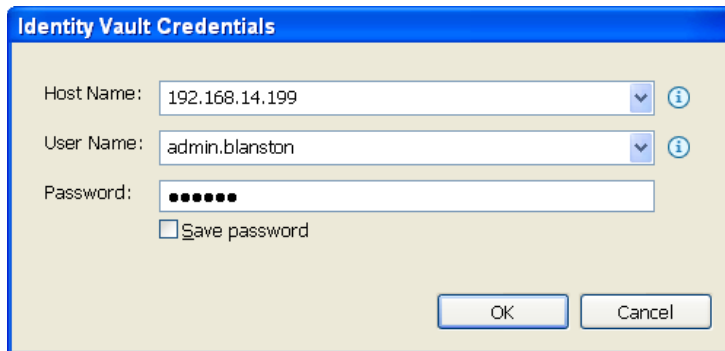
When you add a new Identity Vault to a project, you see the Add Server Association window, where you can accept the default server, specify a server, or browse to a server. The import and deploy features use the server association for later identification. To do this:

- 1 In the Modeler view, drag an Identity Vault icon from the Palette to the Modeler view to bring up the Add Server Association window.



Type the server's DN context in the Server DN field, or click *Browse*.

- 2 If you select *Browse*, fill in the appropriate host name, user name, and password at the Credentials to Identity Vault window. Click *OK*.



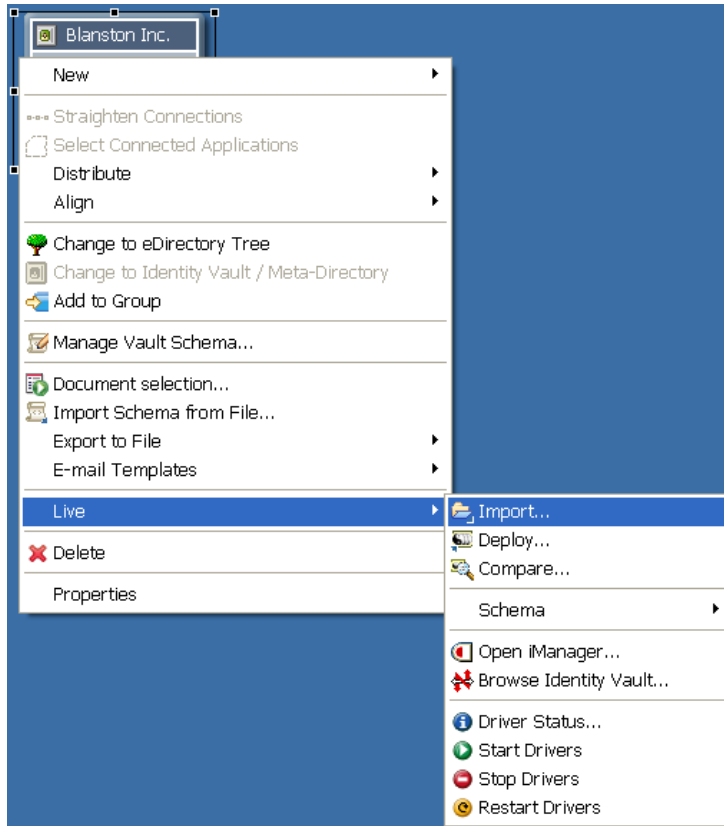
The screenshot shows a dialog box titled "Identity Vault Credentials". It contains three input fields: "Host Name" with the value "192.168.14.199", "User Name" with the value "admin.blanston", and "Password" with a masked password of seven dots. There are information icons (i) to the right of the Host Name and User Name fields. Below the Password field is a checkbox labeled "Save password" which is currently unchecked. At the bottom right of the dialog are "OK" and "Cancel" buttons.

- 3 At the Browse for Server Object window, select the server you want to associate with this driver set and click *OK*.
- 4 At the Add Server Association window, you also see the Identity Manager version displayed. This is important when importing and exporting driver sets and drivers, as you must match driver sets and drivers to the correct Identity Manager version. Click the *For version information or to change the default, click here* entry for more information.
- 5 Click *OK* to close the Add Server Association window and add an Identity Vault to your Modeler view.

15.2.2 Importing a Driver Set from the Identity Vault

To import an Identity Manager Driver Set object (and all contained drivers) into an Identity Vault object in the Modeler view:

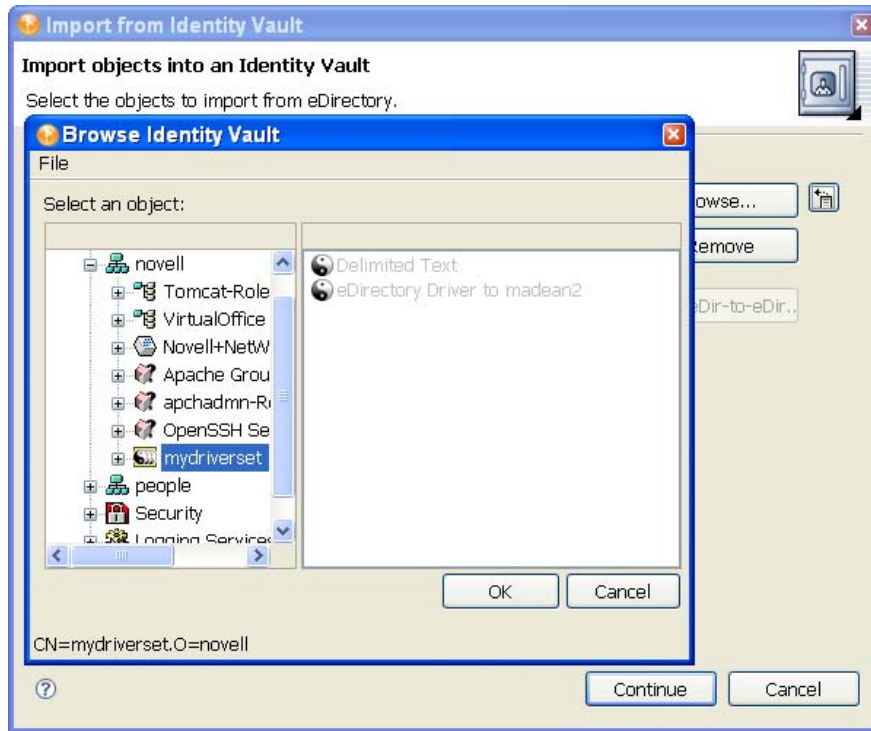
- 1 Right-click the Identity Vault in the Modeler view, then click *Live > Import*.



- 2 (Conditional) If the Driver Set that comes with the Identity Vault creation is empty, you are asked if you want to remove the default Driver Set icon from the selected Identity Vault. Click *Yes*.
- 3 (Conditional) If you filled out the authentication information when you initially created an Identity Vault icon in the Modeler view, go to the Properties view under the Project view. Specify authentication credentials for the selected Identity Vault, then skip to Step 5.
- 4 (Conditional) If you have not yet provided authentication information, specify it now. In the Identity Vault Credentials window, fill in the host name, the user name and password information, then click *OK*.

Use the drop-down lists if they apply to your connection and user information. The *Save Password* option allows Designer to keep password information for future connections to this Identity Vault. Otherwise, you will see the Identity Vault Credentials window the next time you open Designer.

- 5 In the Import from Identity Vault window, browse to the driver set by clicking the *Browse* button.



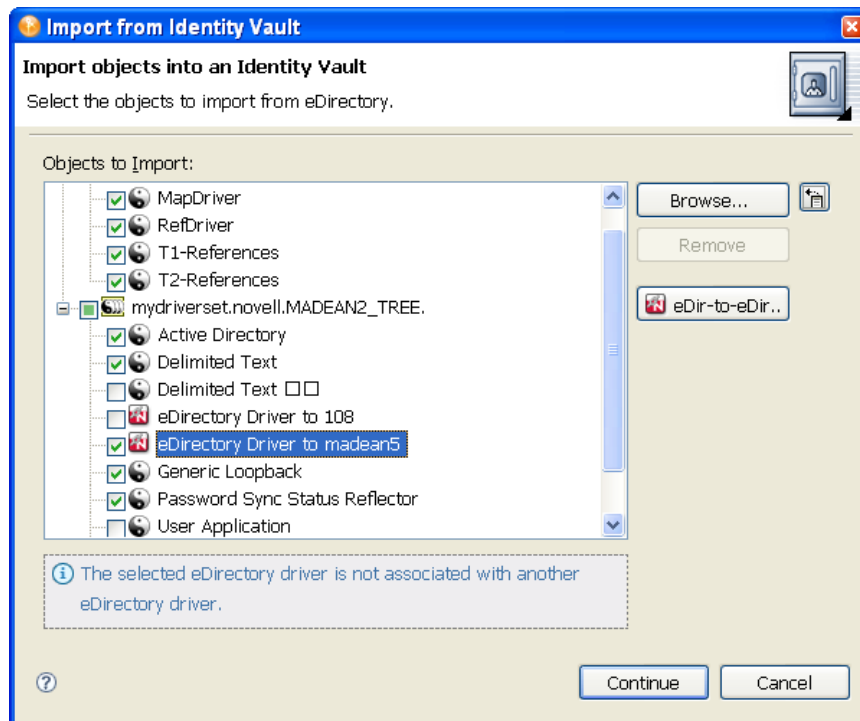
- 6 Select the driver set you want to import, click *OK* to place the driver set in the Objects to Import list in the Import Driver Set from eDirectory window. You can then deselect the drivers you do not want to import by deselecting the box next to the driver name. If you chose the wrong driver set, select the driver set and click *Remove*. Otherwise, click *Continue*.

You can import multiple driver sets during the import operation. Just browse to the various objects that you want to import and add them to the *Objects to Import* list.

Driver sets that are not associated with a server have a red minus in the lower right portion of the driver set icon. These driver sets need a server association in order to be deployed. An error displays if the application can't authenticate to the eDirectory tree you have selected.

- 7 (Conditional) If you are importing one or more eDirectory drivers, select the eDirectory driver in the Objects to Import window, then click the *eDir-to-eDir* button.

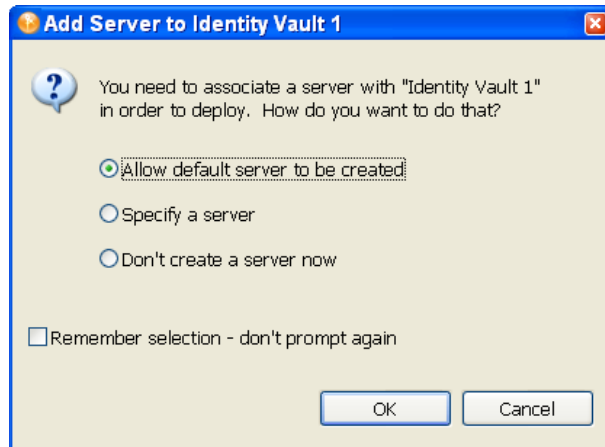
- 8 (Conditional.) If you fill out the user name and password and click *Continue* to import both eDirectory drivers, you then see a Browse Identity Vault window where you select the corresponding eDirectory driver. Select the driver and click *OK*.



You are returned to the Import Driver Sets from eDirectory window, where you can select or deselect the drivers, allowing you to choose only the drivers in a driver set that you want to import.

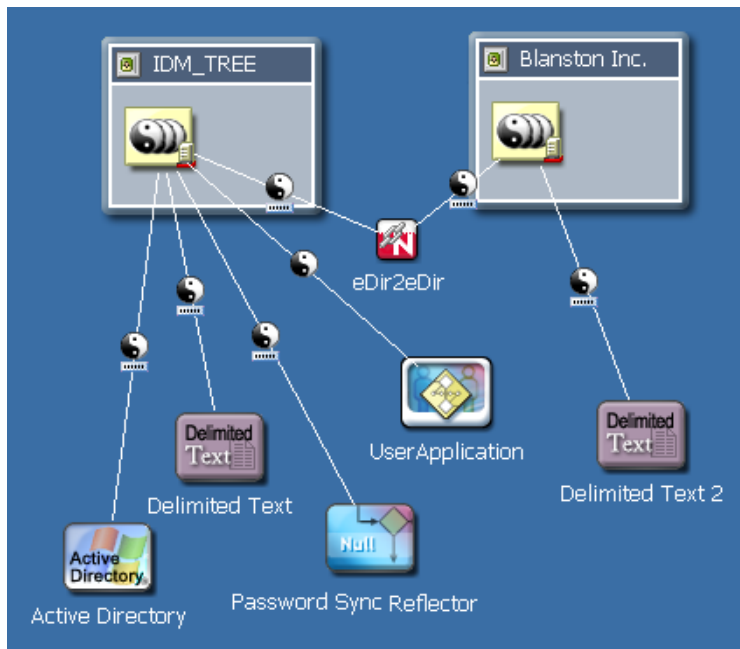
- 9 Click *Continue*. This brings up the Import Summary window, where you can see all of the Driver Set objects that are being imported into Designer. This summary uses the same format as the Compare window (see [Section 15.5, “Using the Compare Feature When Importing,”](#) on [page 291](#) for further information). Click *Import* to continue.

- 10 (Optional) As the import operation progresses, you are asked to associate a server with the Identity Vault. Select the option that best suits your needs.

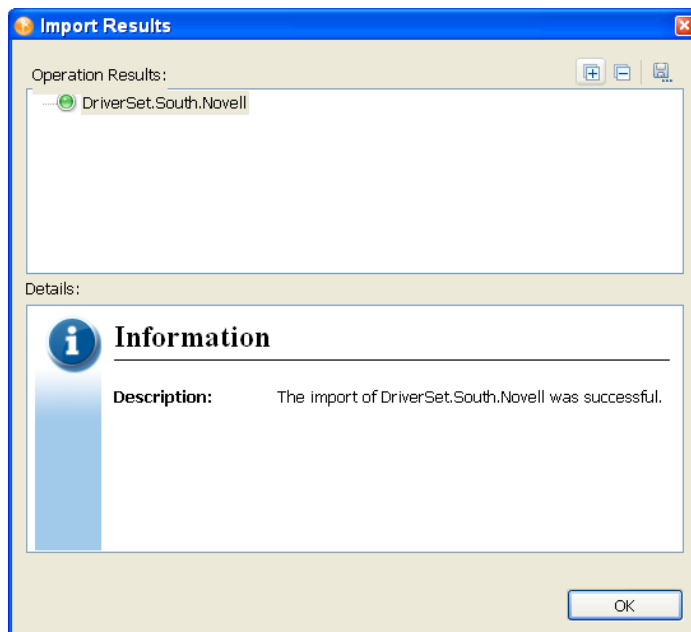


- ◆ **Allow default server to be created.** Creates a dummy server with global configuration values and other elements that are associated with this project until you specify an actual server for the project. You must make sure you have designated a correct Identity Vault server when you deploy the driver set.
 - ◆ **Specify a server.** Brings up the credentials screen, allowing you to designate a host server, a user name, and password for the Identity Vault server for this project.
 - ◆ **Don't create a server now.** Skips all associations for this project. You need to fill in the host information before you deploy this driver set.
 - ◆ **Remember selection--don't prompt again.** Continues to use whatever server option you chose the next time Designer needs to associate a server with an Identity Vault.
- 11 After you decide your plan of action and select the option you want, click *OK* to continue the import procedure.

- 12 Click *Finish*. If you select Step 7 to connect eDirectory drivers, you can view the complete data flow between the two eDirectory drivers, as well as the other drivers you selected.



- 13 When the driver set imports, you see the Import Results window showing you if there were any problems with the import procedure.



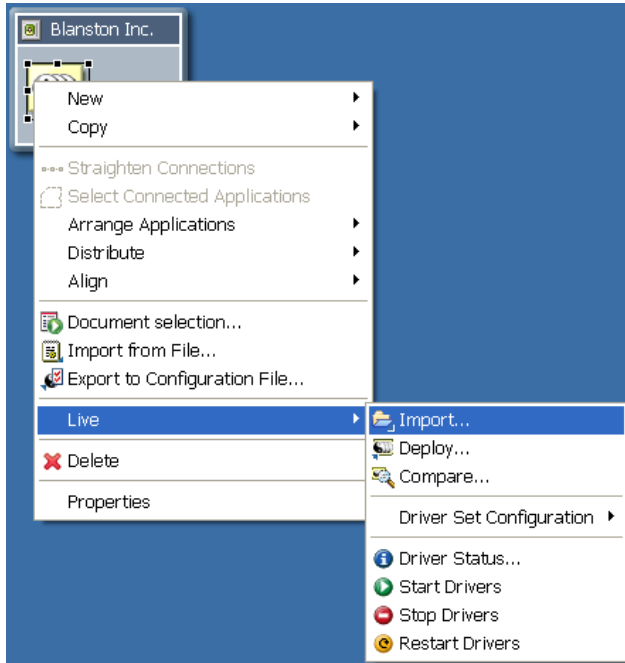
Errors during the import procedure are displayed with a red icon, and you see an Error description that is related to the operation results. If you have multiple errors, selecting an error displays the error's description in the *Details > Description* field. See [“Error Messages and Solutions” on page 498](#) for further information.

- 14 Click *OK* to finish the import process.

15.2.3 Importing a Driver from the Identity Vault

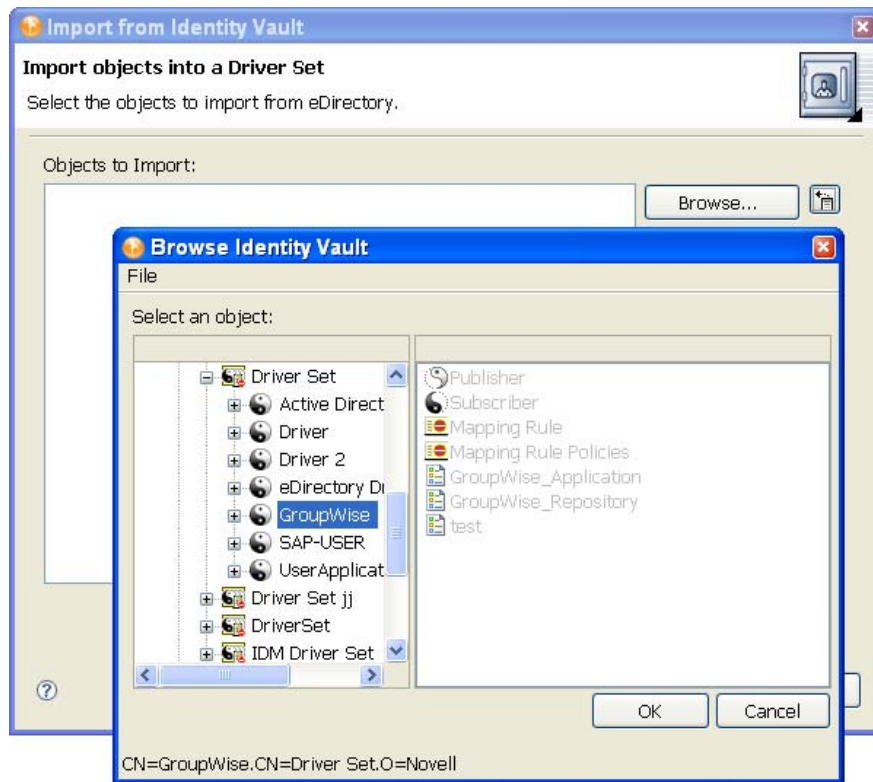
To import an Identity Manager Driver object (and all contained channels and policies) into a driver set:

- 1 Select an Identity Vault in the Modeler. If you add a new Identity Vault to a project, see [Section 15.2.1, “Associating a Server to the Identity Vault,” on page 266](#).
- 2 Verify that the authentication credentials in the Properties view for the Identity Vault are correct.
- 3 Right-click a DriverSet object within the Identity Vault, then select *Live > Import*.



- 4 (Conditional) If the Identity Vault is not authenticated to the eDirectory tree, you see the Identity Vault Credentials window asking for the host name, user name and password. Provide the information, then click *Next*.

- 5 In the Import Drivers from eDirectory window, click *Browse* to select a Driver object from the eDirectory tree.



- 6 Click *OK* to place the driver in the Import from eDirectory window, then click *Continue* to display the Import Summary window, where you can see all of the Driver objects that are being imported into Designer.

This summary uses the same format as the Compare window; however, because you are importing the object, you cannot use the *Reconcile Action* options (see [Section 15.5, “Using the Compare Feature When Importing,”](#) on page 291 for further information).

An error displays if the application can’t authenticate to the eDirectory tree you have selected.

- 7 Click *Import* to continue.
- 8 In the Import Results window, click *OK* if the import worked as planned.
If it does not, you see an Error description that is related to the operation results. If you have multiple errors, selecting an error displays the error’s description in the *Details > Description* field. See [“Error Messages and Solutions”](#) on page 498.
- 9 (Conditional) If you are importing one or more eDirectory drivers, select the eDirectory driver in the Objects to Import window and click the *eDir-to-eDir* button. Fill out the user name and password for the other eDirectory tree and select *Continue* to import both eDirectory drivers.
- 10 In the Browse Identity Vault window, select the corresponding eDirectory driver, then select the driver and click *OK*. You are returned to the Import Drivers from eDirectory window.
- 11 When the driver imports, you see the Import Results window showing you if there were any problems with the import procedure.

Errors during the import procedure are displayed with a red icon, and you see an Error description that is related to the operation results. If you have multiple errors, selecting an error

displays the error's description in the *Details > Description* field. See [“Error Messages and Solutions” on page 498](#) for further information.

12 Click *OK* to finish the import process.

15.3 Importing a Driver Configuration File

You can import an Identity Manager driver configuration file into the selected driver set for a project by using the Import from File option, which imports an XML configuration file that can be a driver set, an individual driver, driver channels, or policies. If you import a driver configuration file that contains only a policy, Designer creates the framework for parent containment objects, such as a channel, a driver, or a driver set. Such parent containment objects do not contain attributes; they are only the framework of the channel, driver, or driver set from where the policy came.

These configuration files are in XML format, which you can save to a local, removable, or network directory through the Export to Configuration File option. After you select the driver configuration file to import, you are prompted to provide driver-specific information for setting up the driver and building policies. The exported `.xml` files are compatible with those used by the iManager driver configuration file plug-ins for Identity Manager 2.0.2 and later. This allows you to export configuration files from Designer and import those files through iManager or through Designer's Import feature.

You can also use the Import from File option as a backup method. For example, if you are working on a group of policies in an Active Directory* driver, you can use the Export to Configuration File to back up your work, adding dates to the exported files so you can select a particular driver configuration that you want to use as the basis for your Active Directory driver. If something goes wrong, you can re-import the base driver. You can also use this export/import feature to export and import just the policies you are working on.

You can import a configuration from a file in three ways:

- ◆ [Section 15.3.1, “Importing an Identity Manager Project from the File System,” on page 275](#)
- ◆ [Section 15.3.2, “Importing a Driver Configuration from a File in the Modeler View,” on page 275](#)
- ◆ [Section 15.3.3, “Importing from a File through the Outline View,” on page 278](#)

15.3.1 Importing an Identity Manager Project from the File System

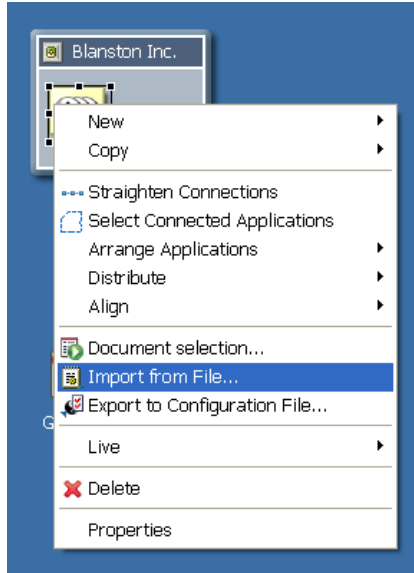
The *Import an Identity Manager Project from File System* option allows you to import an Identity Manager project that has been saved to the file system through the *Export > File System* option. The project must have a valid `.project` file and accompanying file structure for the project to completely import. This topic is covered under [Section 15.1.2, “Importing a Project from the File System,” on page 262](#).

15.3.2 Importing a Driver Configuration from a File in the Modeler View

A driver configuration file can contain one or more driver configurations. You can import a previously exported configuration file or one of the sample `.xml` driver configurations that are included with Designer.

To import a configuration file containing one or more Identity Manager driver configurations into a driver set:

- 1 Select an Identity Vault in the Modeler view.
- 2 Right-click a DriverSet object within the Identity Vault, then click *Import from File*.



- 3 In the Import a Driver Configuration File window, type the name of the configuration file, or browse to and select the .xml file.

If you use the Browse feature, by default Designer takes you to the `C:\Program Files\Novell\Designer\eclipse\plugins\com.novell.designer.idm_2.0.0.x\defs\driver_configs` directory where the Identity Manager driver templates that come with Designer are stored on Windows workstations. On Linux workstations, the default drivers are stored in the `C:\home\username\designer\eclipse\plugins\com.novell.designer.idm_2.0.0.x\defs\driver_configs` directory. (The “x” after the idm_2.0.0 number refers to Designer build numbers) You can use one of these driver files or you can browse to a directory containing an exported configuration file.

- 4 Complete the import by filling out the Import Information Requested forms for the driver configuration file.

The driver writer requested that the following information be supplied in order to import this driver configuration file.

Information requested: * Required

The name of the driver contained in the driver configuration file is "eDirectory Driver". Enter the actual name you want to use for the driver.

Driver name: *

eDirectory Driver

Enter the DNS host name or IP address, and communication port of the DirXML server in the remote tree. For example:
[151.155.144.23:8196]
[hostname:8196]

Remote Tree Address and Port: *

hostname : 8196

Data flow can be configured at this time for the driver. Select the data flow that you desire. Bi-directional means that both eDirectory trees are authoritative sources of the data synchronized between them. Authoritative means that the local tree will be the authoritative source. Subordinate means that the local tree is NOT an authoritative source.

Configure Data Flow:

Bi-directional

Choose the desired form of placement. Choose Mirrored to synchronize objects hierarchically between the local and remote trees. Choose Flat to synchronize all Users and Groups into specific containers. Choose Department to synchronize Users and Groups by department (OU).

Configuration Option:

Mirrored

OK Cancel

The form varies, depending on the driver configuration file selected. Saved files might only prompt for a driver name. You might have imported a single driver or a collection of drivers (a driver set). If you are importing a driver set, fill out an Import Information Requested template for each imported driver.

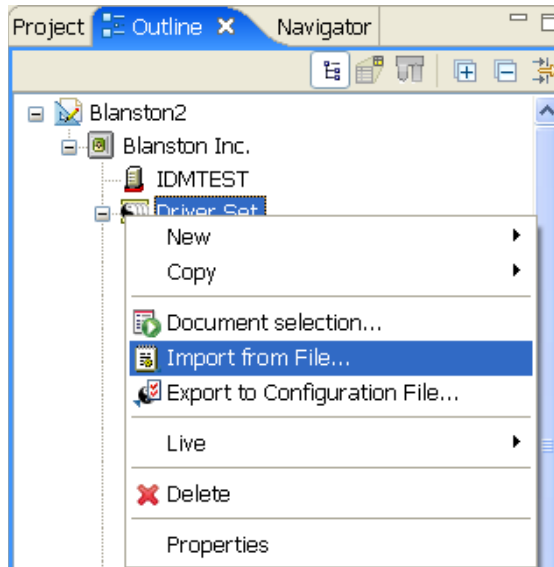
If you fail to provide the information requested in the Import Wizard page, your driver configuration is incomplete, and the exported driver is also incomplete. Do not use the Configuration Wizard on the driver configuration file that you exported. The wizard brings up the Import Information Requested forms, but these forms are pulled from the default driver file and overwrite the driver file you exported.

- 5 Click *OK*, then click *OK* in the Import Configuration Results window.

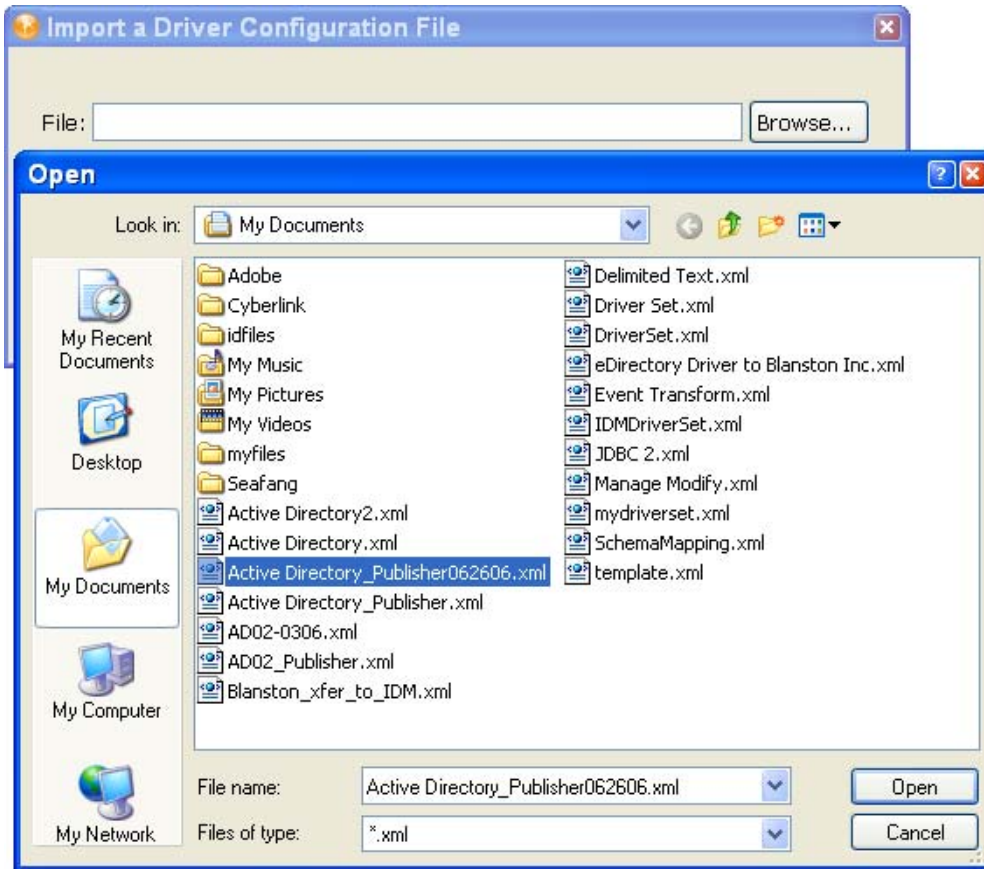
15.3.3 Importing from a File through the Outline View

You can use the Outline View to import driver sets, drivers, channels, and policy configuration files from the file system. The following example demonstrates how to import a driver, but the procedure also works for the other files.

- 1 Double-click the *System Model* icon under a project name in the Project view. This brings up the project model in the Modeler view.
- 2 Click the *Outline* tab.
- 3 Right-click the Driver Set object and select *Import from File*.



- 4 In the Import a Driver Configuration File window, type the name of the configuration file, or browse to and select the .xml file.



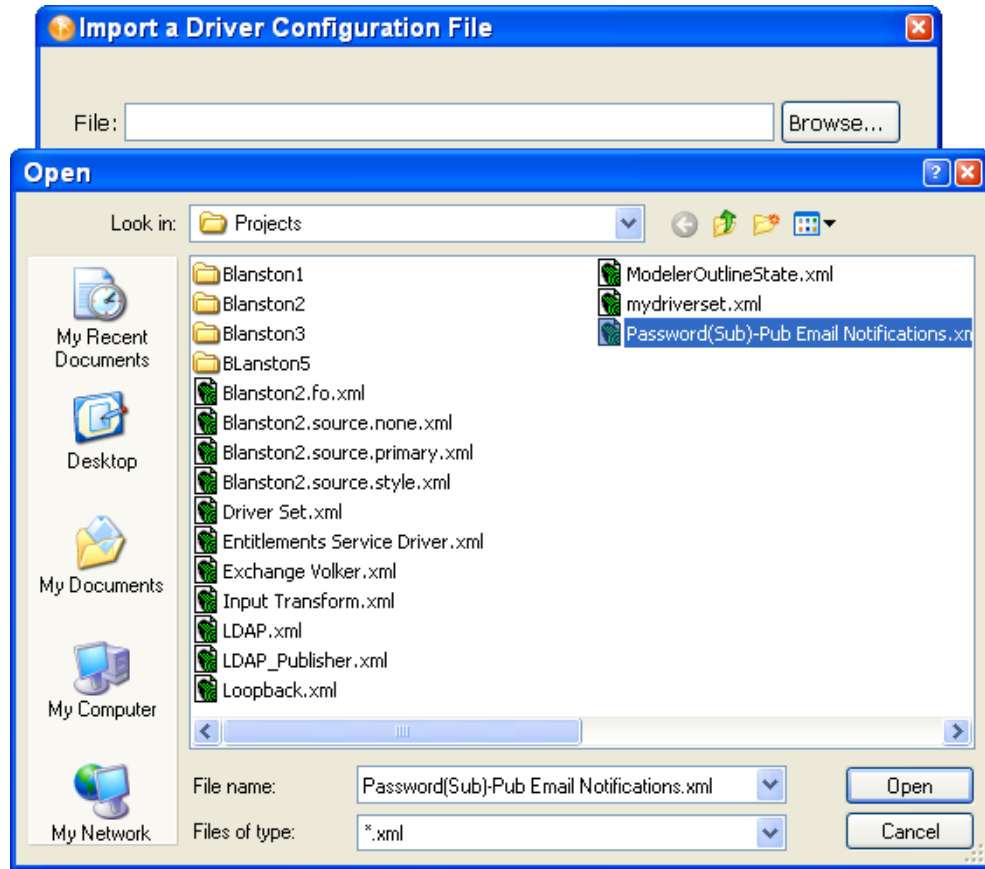
Then a driver is exported, Designer uses the name of the driver set or driver object, to which you can add dates. In this example, it is an Active Directory driver that was exported June 26th and is now being imported.

- 5 Click *Open*, then click *OK* to import the configuration file.

To import a policy into a driver set:

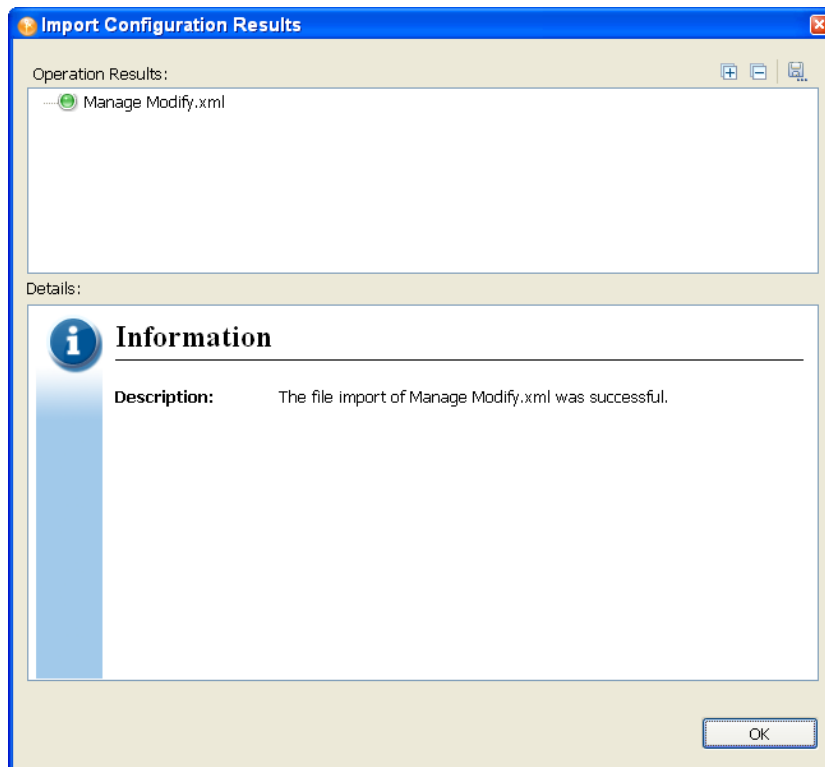
- 1 In the Outline view, click the *Driver Set* icon, then click *Import from File*.

- 2 In the Import a Driver Configuration File window, type the name of the configuration file, or browse to and select the .xml file.



- 3 Click *Open*, then click *Save* to import the selected policy.
Each policy is saved to its own .xml file. By default, Designer uses the name of the policy or rule selected.
- 4 In the Perform Prompt Validation window, you are asked if you want to fill in required driver information. If you answer *Yes*, you see the Import Information Requested dialog box as described in Step 5 and you must provide values to all of the required fields. If you answer *No*, you still see the Import Information Requested dialog box, but it isn't necessary to fill in the required information.
- 5 Complete the import by filling out the Import Information forms for the driver configuration file as necessary. Policies are saved with a skeleton driver configuration structure, which designates where the policy resides. In this case, the driver already existed and the imported policy was initially written for that driver.
- 6 Click *OK*.

- 7 When the policy or rule is imported, you see the Import Configuration Results window, which indicates if there were any problems with the import procedure.



Errors during the import procedure are displayed with a red icon, and you see an Error description that is related to the operation results. If you have multiple errors, selecting an error displays the error's description in the *Details > Description* field. See [“Error Messages and Solutions” on page 498](#) for further information.

- 8 Click *OK* to finish the import process and close the Import Configuration Results window.


15.4 Importing Channels, Policies, and Schema Items from the Identity Vault

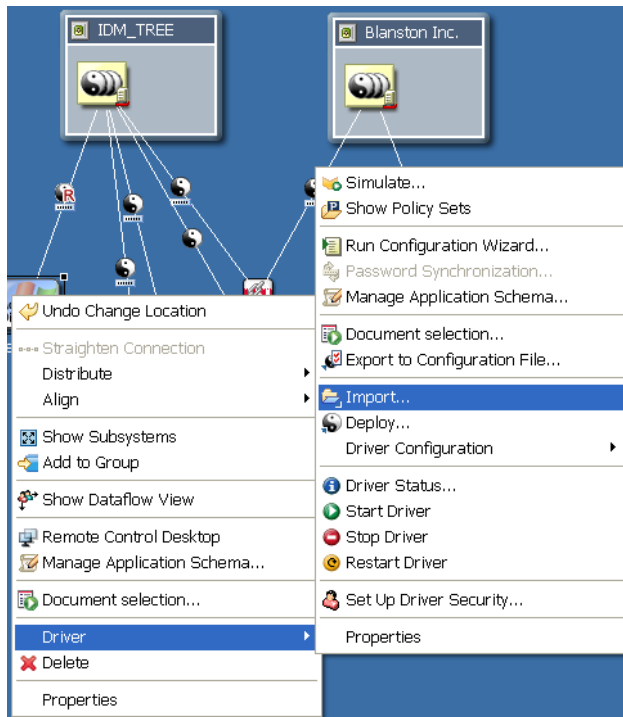
A channel is a combination of rules and policies, and Designer allows you to import a channel instead of the entire driver. The Subscriber and Publisher channels describe the direction in which the information flows. The Subscriber channel takes the event from the Identity Vault and sends that event to whatever the receiving system is (application, database, CSV file, etc.) The Publisher channel takes the event from the application, database, CSV file, etc., and sends that event to the Identity Vault. The Subscriber and Publisher channels act independently; actions in one are not affected by what happens in the other.

- ◆ [Section 15.4.1, “Importing a Channel,” on page 282](#)
- ◆ [Section 15.4.2, “Importing a Policy,” on page 285](#)
- ◆ [Section 15.4.3, “Importing a Schema,” on page 288](#)

15.4.1 Importing a Channel

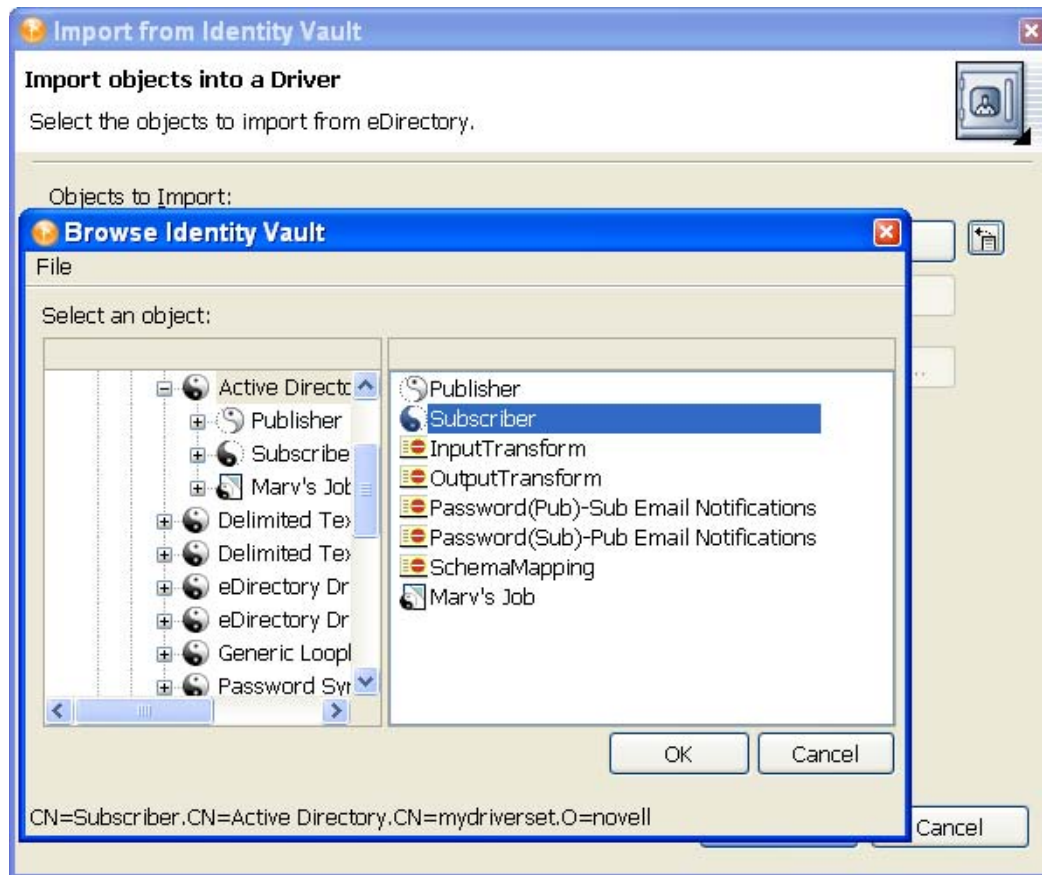
To import an Identity Manager channel (a Subscriber channel or a Publisher channel) object (and all contained policies) into a driver:

- 1 Select either a Driver object or an Application object in the Modeler view. The Driver object is represented by the line between the Identity Vault and the Application object and has a circle icon to represent it . The Application object connects to the Identity Vault through the Driver object.
 - 2 Right-click a Driver object, then click *Live > Import*.
- or
- Right-click an Application object and click *Driver > Import*.



If Designer can't authenticate to the eDirectory tree specified in the Identity Vault, you see the Identity Vault Credentials window if you have not previously specified the authentication credentials or if you do not save the password. Fill in the appropriate information and click *OK*.

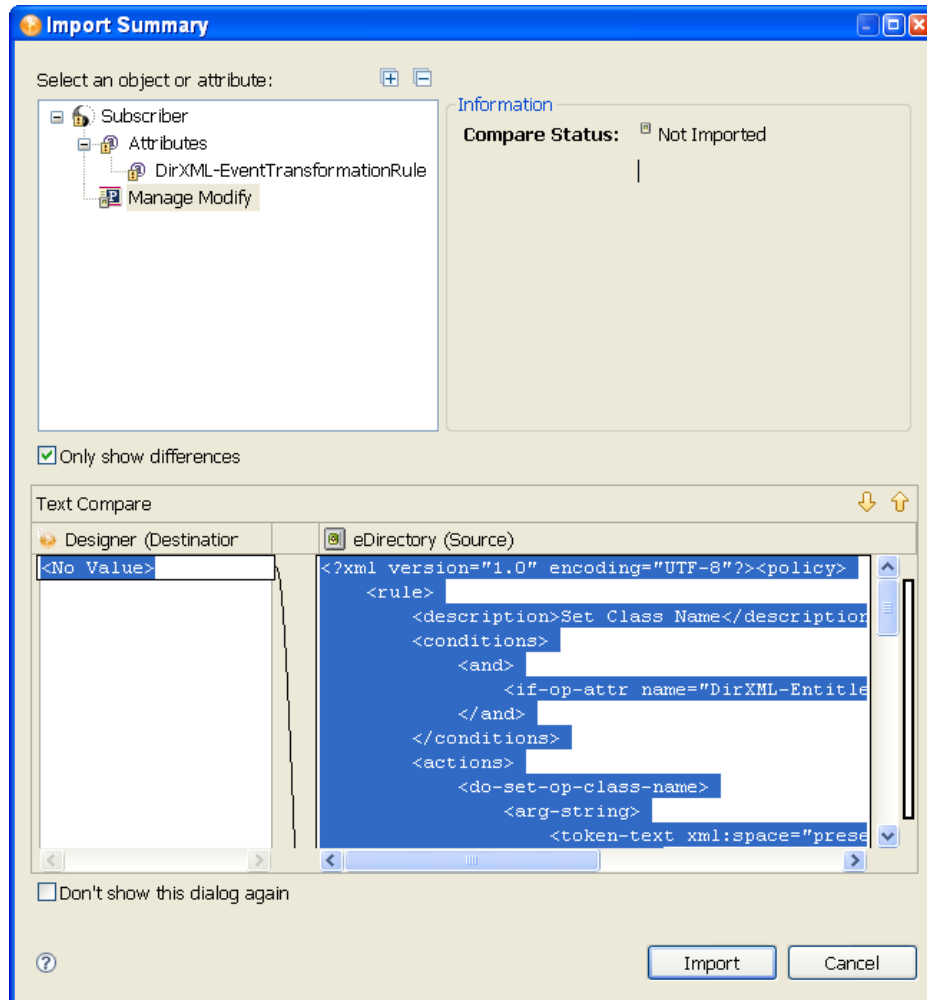
- 3 In the Import from Identity Vault window, browse to and select either a Publisher or a Subscriber Channel object from the eDirectory tree under the corresponding driver.



You can import more than one channel at a time; if you want to import both channels, select one channel, click *OK*, then browse to the next channel, select it, and click *OK*.

- 4 Click *Continue*.

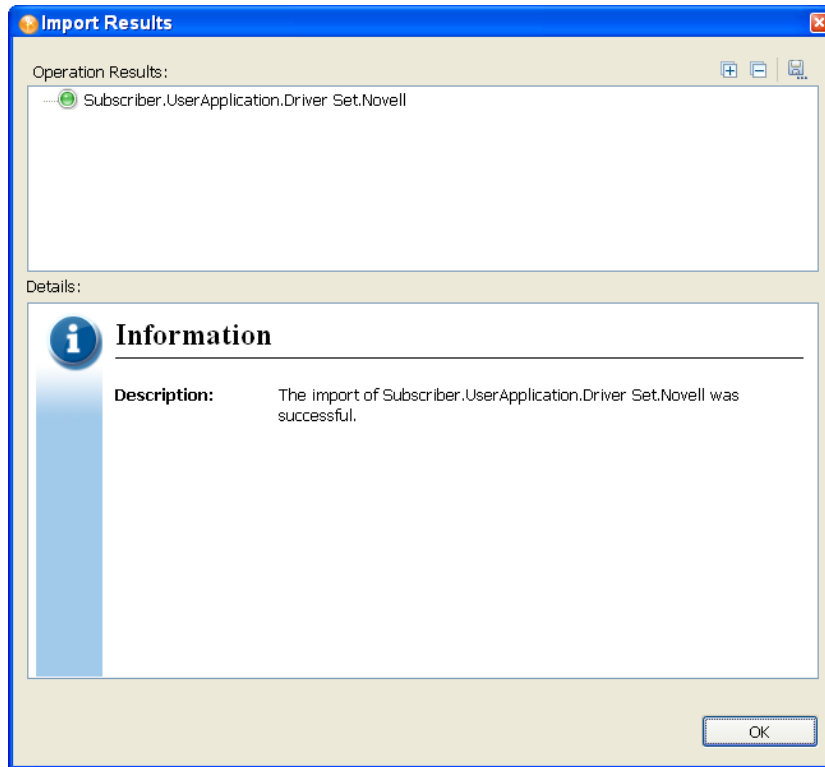
- 5 As the channel imports, you see the Import Summary window showing you the differences between eDirectory (the source of the import) and Designer (the destination).



You can click the different objects in the channel view differences between the two drivers. All channel information is overwritten by the import procedure; however, the rest of the driver is unaffected.

Click *Import*.

For more information on Compare, see [Section 15.5, “Using the Compare Feature When Importing,”](#) on page 291.



If there are any problems with the import procedure, they are displayed with a red icon in the Import Results window and you see an error description that is related to the operation results. If you have multiple errors, selecting an error displays the error’s description in the *Details > Description* field. See [“Identity Vault Configuration Errors”](#) on page 499 for further information.

- 6 Click *OK* to finish the import process.

15.4.2 Importing a Policy

A policy is a collection of rules and arguments that allows you to transform the data that an application sends to and receives from eDirectory. You use policies to manipulate the data you receive from eDirectory or from the connected system so they can synchronize the information in their databases. Each driver connects to a different system, and policies tell the driver how to synchronize the data on that connected system to the Identity Vault.

You might use the Import feature for policies more than anything else. For example, you can set up a policy to allow users with the title “Manager” to be placed in a specific container, no matter which application the information is coming from, and you can place this information in multiple connected systems. However, because each application is different, you need to modify the arguments and rules within policies to reflect those differences. For more information about policies, see [Understanding Policies for Identity Manager 3.5.1](#) and [Policies in Designer 2.1](#).

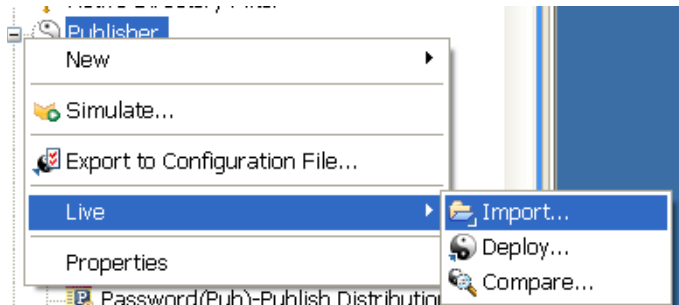
To import an eDirectory Policy object (for example, a rule or a style sheet) into a driver or channel (Subscriber or Publisher):

- 1 Select a driver in the Modeler view.

or

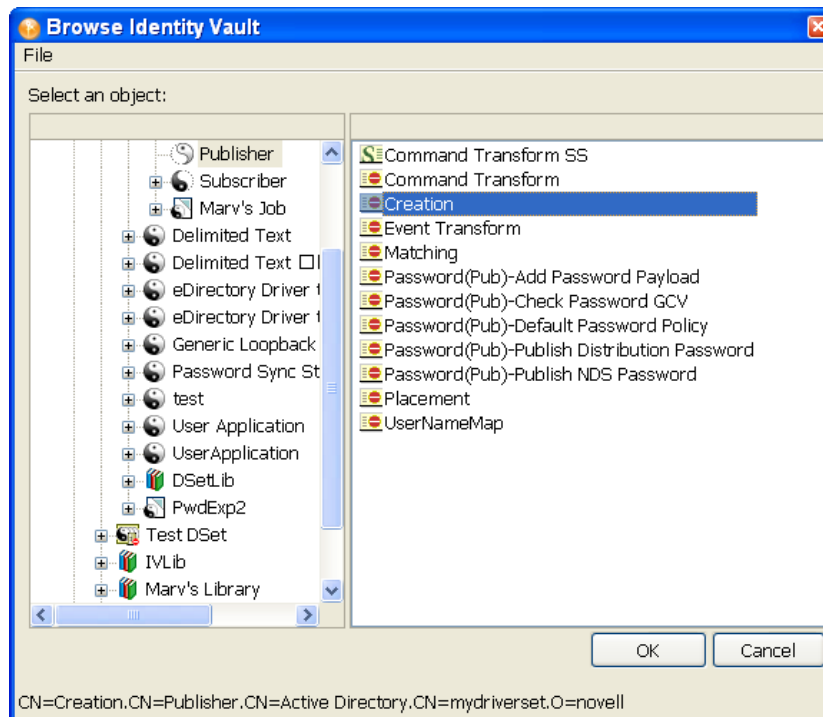
Click the *Outline* tab and select a Driver or Channel object from the Outline view.

- 2 Verify that the authentication credentials in the Properties view for the selected Identity Vault are correct.
- 3 Right-click the Driver or Channel object, then click *Live > Import*.



If the application can't authenticate to the eDirectory tree, you see the Identity Vault Credentials window asking for the host name, user name, and password if you have not previously specified the authentication credentials or if you do not save the password. Fill in the appropriate information and click *OK*.

- 4 In the Import from Identity Vault window, click *Browse*, then select a policy object from the channel you specified when you started the import process.



Policies are found under either the Publisher or Subscriber channel of a selected driver or under the driver itself. Be sure to match the proper policy to the proper channel or driver object.

- 5 Click *OK*, then click *Continue* to import the policy.

You see the Import Summary window showing you the differences between eDirectory (the source of the import) and Designer (the destination). You can click the different objects in the policy to see what is different between the two policies. All selected policy information is overwritten by the import procedure; however, the rest of the driver is unaffected.

- 6 Click *Import*.

If the importing policy contains the same values as the policy in Designer, you are not allowed to import the policy. (See [Section 15.5, “Using the Compare Feature When Importing,”](#) on [page 291](#) for more information on the Compare feature.)

Clicking *Import* brings up the Import Results window. If there are any problems with the import procedure, they are displayed with a red icon, and you see an Error description that is related to the operation results. If you have multiple errors, selecting the different errors displays the error’s description in the *Details > Description* field. See [Section 24.4.3, “Error Messages and Solutions,”](#) on [page 498](#) for further information.

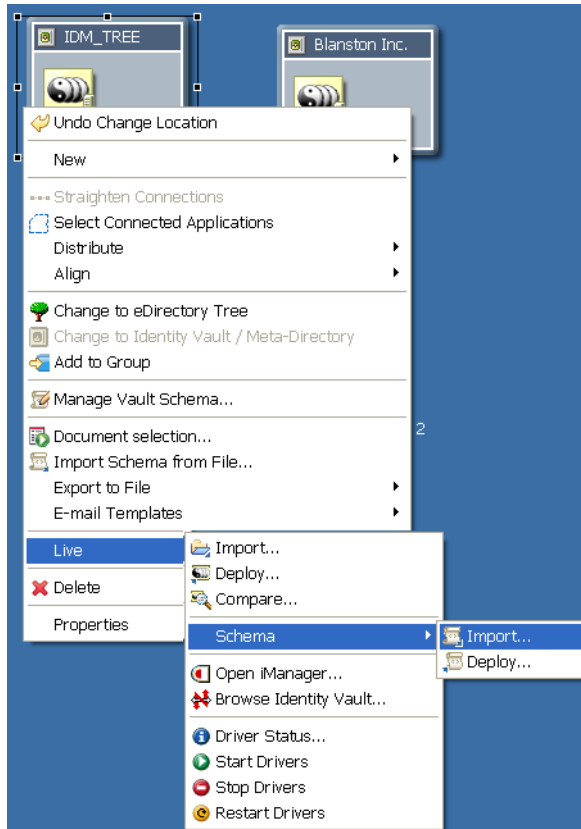
- 7 Click *OK* to finish the import process.

For policy design, see the Policy Builder and Policy Management Help topics within the Designer utility. Also see [Understanding Policies for Identity Manager 3.5.1](#) and [Policies in Designer 2.1](#).

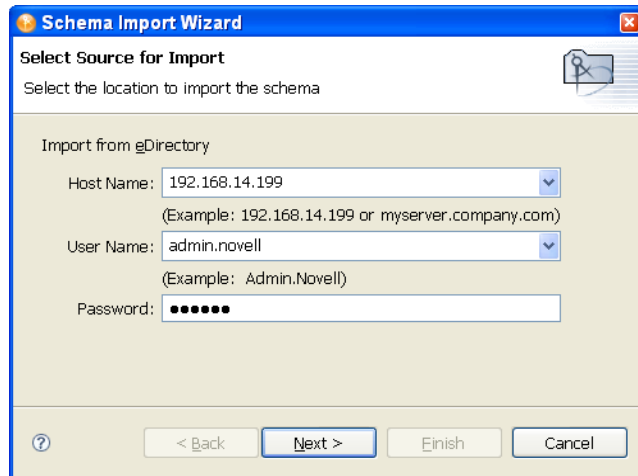
15.4.3 Importing a Schema

You can import a schema from the Identity Vault or from a .sch file into your project. When you import a schema, you can select the whole Identity Vault schema (not recommended) or just the schema differences between the Identity Vault and your project.

- 1 Bring up the project in Designer's Modeler view. Right-click the Identity Vault and select *Live* > *Schema* > *Import*.



- 2 On the Select Source for Import page, select *Import from eDirectory* if you are connected to an actual Identity Vault.



The screenshot shows a dialog box titled "Schema Import Wizard" with a sub-header "Select Source for Import". Below the sub-header is the instruction "Select the location to import the schema". The main area is titled "Import from eDirectory" and contains three input fields: "Host Name" with a dropdown menu showing "192.168.14.199" and a note "(Example: 192.168.14.199 or myserver.company.com)"; "User Name" with a dropdown menu showing "admin.novell" and a note "(Example: Admin.Novell)"; and "Password" with a masked field of six dots. At the bottom, there are four buttons: a help icon (?), "< Back", "Next >", "Finish", and "Cancel".

- 3 In the Import from eDirectory section, specify the host name, user name and password connection information.

The Host Name and User Name entries have drop-down menus storing the last information you typed into these fields, which you can use for filling in these entries.

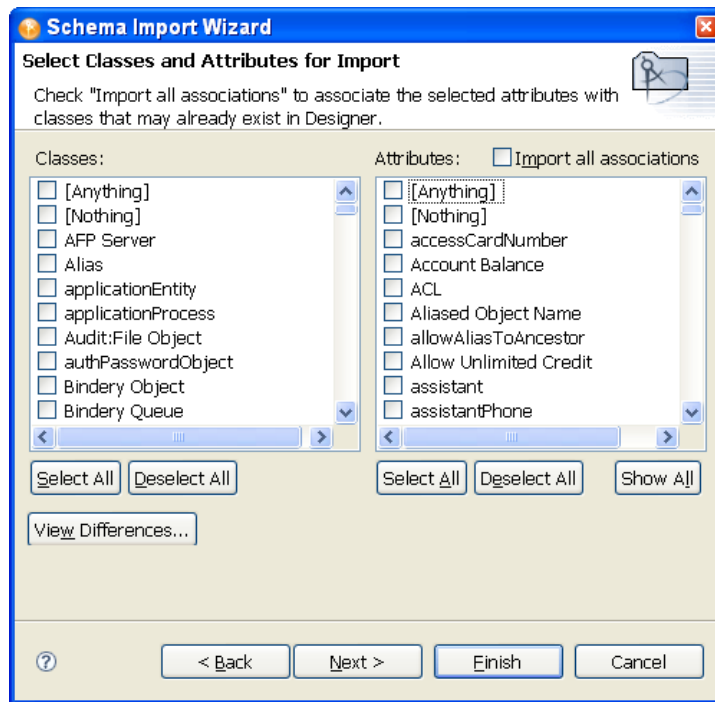
- 4 Click *Next*.

- 5 Decide which classes and attributes to import.

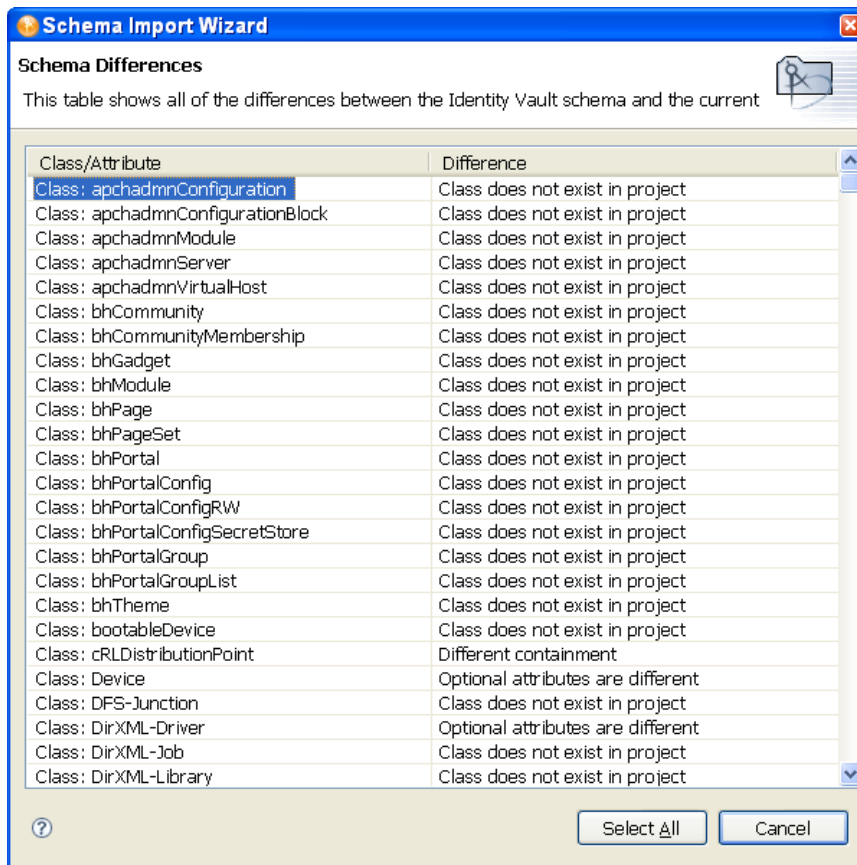
On the Select Classes and Attributes for Import page, you can select all of Identity Vault's scheme, including classes and attributes. However, this creates very large documents when you document the project (600 pages or more: see [Section 24.5, "Issues When Generating Documents,"](#) on page 509 for more information on this matter).

- 6 If you want to import all the classes and attributes, click *Select All*, click *Finish*, then skip to Step 8.

Select only the classes and attributes that you want to import. If you only want to import the schema differences between Identity Vault and your project, click *View Differences*, then continue with **Step 8**.



- 7 On the Schema Differences page, you see the schema differences between Identity Vault and your project. Click *Select All* if you only want to import schema differences. Otherwise, click *Cancel*.



- 8 Selecting *Select All* brings you back to the Select Classes and Attributes for Import page with the schema differences now selected under the Classes and Attributes headings. By default, the *Import All Associations* box is also selected. Leave it selected and click *Finish*. If you selected *Cancel* on the Check Differences page, make your schema selections here and click *Finish*.
- 9 On the Import Messages page of the Schema Import Wizard, click *OK*.
If you want to save the differences to a log file, click *Save to Log*. This brings up the Save As dialog box, where you can choose a filename and directory to store the file in.
- 10 Click *Save*, then click *OK*.

15.5 Using the Compare Feature When Importing

Designer's Compare feature allows you to see differences between the driver sets, drivers, channels, and policies that are stored in projects and those that are running in deployed systems, and reconcile any differences to either Designer or Identity Vault. Previous versions of Designer only provided conflict resolution when importing a Driver. While importing, you could select which policies of the driver you wanted to update, but you could not view any differences between existing and new values.

Designer provides conflict resolution on an object-by-object basis and allows you to view the differences between existing and new values when importing and deploying driver sets, drivers, channels and policies. For example, before importing a driver object in Designer to a driver object that already exists in the Identity Vault, you can run Compare. Compare shows whether the driver objects are equal (no action is necessary) or unequal. If unequal, you can choose not to reconcile the driver objects, choose to update the driver object in Designer, or choose to update the driver object in the Identity Vault.

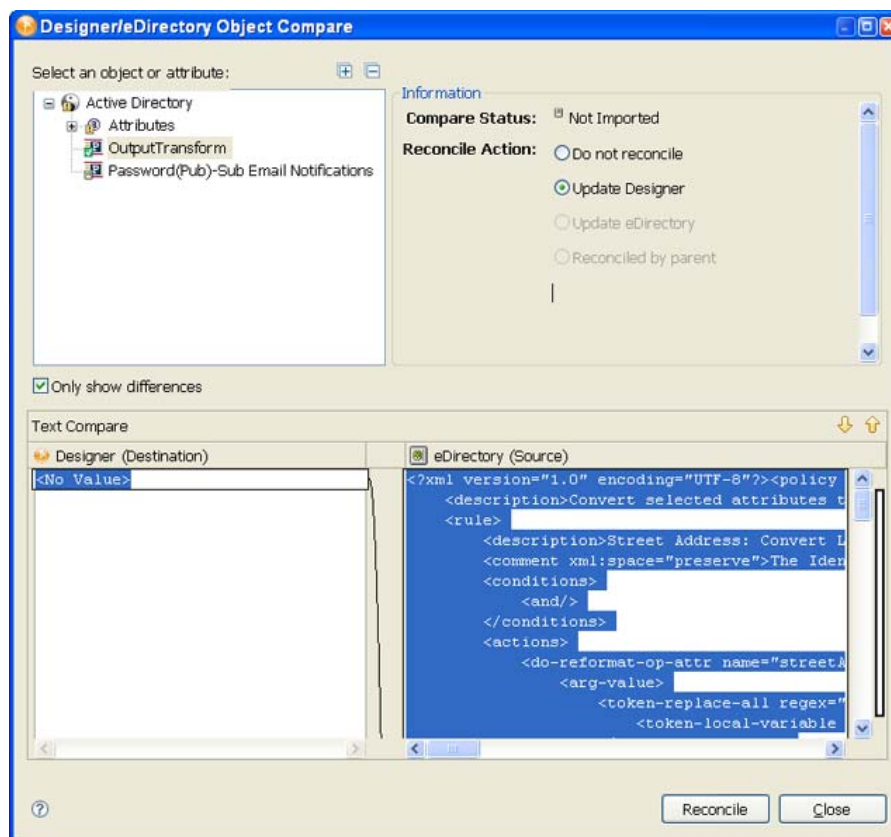
You can run the Compare feature at any time. If you choose to reconcile the differences between drivers objects in Designer and eDirectory™ while in Compare, you won't need to run Import or Deploy.

- ◆ [Section 15.5.1, “Using Compare when Importing a Driver Object,” on page 292](#)
- ◆ [Section 15.5.2, “Using Compare On a Channel Object,” on page 295](#)
- ◆ [Section 15.5.3, “Using Compare On a Policy,” on page 295](#)
- ◆ [Section 15.5.4, “Matching Attributes with Designer Properties,” on page 295](#)

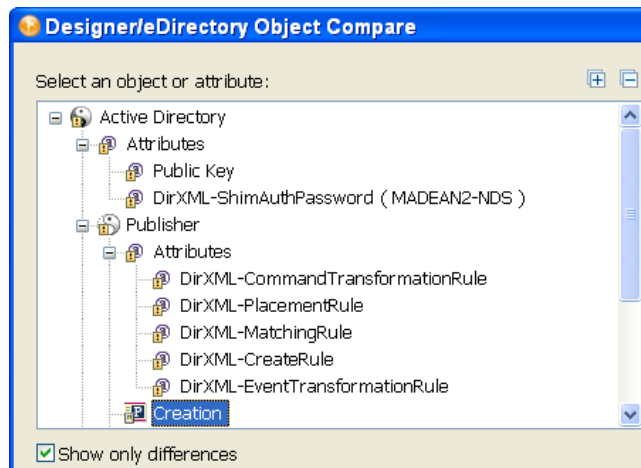
15.5.1 Using Compare when Importing a Driver Object

Use this procedure if you want to import a Driver object from the Identity Vault and the same driver already exists in Designer.

- 1 Right-click the driver object in either the Modeler view or in the Outline view, then click *Live* > *Compare* to bring up the Designer/eDirectory Object Compare window.



Under the *Select an object or attribute*, you see the selected object with the differences between Designer's and eDirectory's driver object. You can select the attributes and child objects to see the actual differences displayed in the *Text Compare* area.

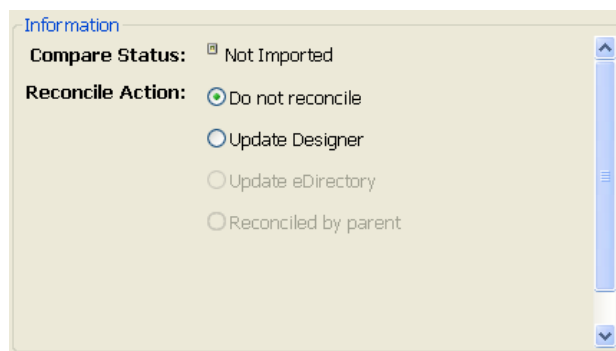


The plus icon at the right side of the *Select an object or attribute* allows you to expand all elements in the parent object, and the minus icon collapses all of the elements. The “?” button in the bottom left portion of the window displays the Summary/Compare dialog box help.

Server-specific attributes are attributes that have a value for each server that is associated with a driver set. These attributes are displayed in the *Attributes* list with the server name in parentheses to the right of the attribute name.

- By default, the Compare window only displays values that are different between Identity Vault and Designer. To view all of the object values, deselect *Show only differences*.

Values that are equal are shown as *Equal* on the *Compare Status* line under *Information*.



The following table describes what you see in the Compare Status line:

Compare Status	Description
Equal	The selected attribute's value or all attributes of the selected object are the same in eDirectory and Designer.
Unequal	The value of the selected attribute, or one or more attributes of the selected object are different in eDirectory and Designer.

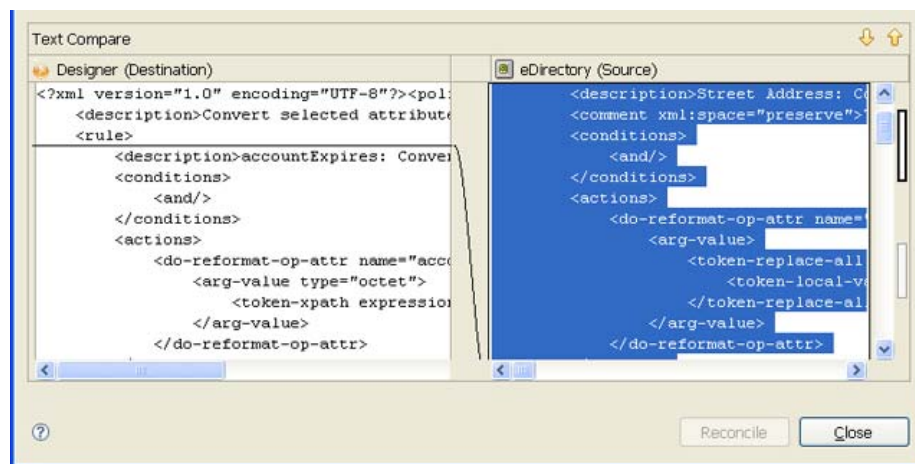
Compare Status	Description
Not Deployed	The selected object or the object containing the selected attribute is not deployed to eDirectory.
Not Imported	The selected object or object containing the selected attribute does not exist in Designer.

The overlay image displayed in the object and attribute tree identifies objects or attributes that need reconciliation. You can also see an *Attribute Note* if you select an attribute.

- In the *Information* portion of the Compare window, select how you want to reconcile the differences between the Source and Destination. If *Compare Status* shows *Unequal*, you have three choices:
 - ◆ **Do not reconcile.** To do nothing, keep the default value of *Do Not Reconcile*.
 - ◆ **Update Designer.** To update the driver in Designer so that it contains the same information as the driver in the Identity Vault, select *Update Designer*.
 - ◆ **Update eDirectory.** To update the driver in eDirectory to reflect the changes you have just made to the driver in Designer, select *Update eDirectory*.

If you select the parent object to perform the update, then all of the child objects under the parent reflect that choice and you see the *Reconciled By Parent* button selected. If you do not choose a parent object, you can reconcile each child object individually.

- View the differences displayed in the Text Compare area.



The Text Compare values displayed in the bottom portion of the Designer/eDirectory Object Compare window vary, depending on the object being compared. For instance, Compare shows you changes down to the policy level. The Text Compare dialog box uses the Eclipse Compare editor to compare attributes that contain XML data, such as policy data, driver filters, or configuration data. The differences in the code are highlighted in blue.

- After you view the differences, click *Reconcile* to perform the reconciliation actions for each object in the tree, or click *Close* to close the Designer/eDirectory Object Compare screen. After reconciliation, the object matches both locations and has been imported or deployed through the action.

15.5.2 Using Compare On a Channel Object

Use this procedure if you want to import a channel object from the Identity Vault and the same channel already exists in Designer. You can view the differences and decide whether to reconcile them.

- 1 Right-click the channel object in the Outline view. Click *Live > Compare* to bring up the Designer/eDirectory Object Compare window.

All Compare windows behave as described under the [Section 15.5.1, “Using Compare when Importing a Driver Object,” on page 292](#).

- 2 After reconciliation, the Channel object matches both locations and has been imported or deployed through the action.

15.5.3 Using Compare On a Policy

Use this procedure if you want to import a policy object from the Identity Vault and the same channel already exists in Designer. You can view the differences and decide whether to reconcile them.

- 1 Right-click the policy object in the Outline view. Select *Live > Compare* to bring up the Designer/eDirectory Object Compare window.

All Compare windows behave as described under the [Section 15.5.1, “Using Compare when Importing a Driver Object,” on page 292](#).

- 2 After reconciliation, the policy object matches both locations and has been imported or deployed through the action.

15.5.4 Matching Attributes with Designer Properties

The attributes of the object are displayed in the single list. Selecting an attribute displays its value below the attribute list with the Designer value on the left and the eDirectory value on the right. The name displayed in the list is the eDirectory attribute name.

The following tables map the eDirectory attribute to the Designer property page or control where you can change or set the attribute (you can’t make changes inside the Compare window).

- ♦ [Table 15-1 on page 295](#) shows Driver Set eDirectory attributes
- ♦ [Table 15-2 on page 296](#) shows Driver eDirectory attributes
- ♦ [Table 15-3 on page 297](#) shows Channel eDirectory attributes
- ♦ [Table 15-4 on page 297](#) shows the Job eDirectory attributes
- ♦ [Table 15-5 on page 297](#) shows the Resource eDirectory attributes

Table 15-1 *Driver Set eDirectory Attributes*

Driver Set eDirectory Attribute	Designer Property
DirXML-DriverTraceLevel	Driver Set Properties->Trace->Driver Trace Level
DirXML-XSLTraceLevel	Driver Set Properties->Trace->XSL Trace Level

Driver Set eDirectory Attribute	Designer Property
DirXML-JavaDebugPort	Driver Set Properties->Trace->Java Debug Port
DirXML-JavaTraceFile	Driver Set Properties->Trace->Java Trace File
DirXML-TraceSizeLimit	Driver Set Properties->Trace->Trace File Size Limit
DirXML-LogLimit	Driver Set Properties->Driver Set Log Level->Log Limit
DirXML-LogEvents	Driver Set Properties->Driver Set Log Level->Log Specific Events
DirXML-ConfigValues	Driver Set Properties->Global Configuration Values

Table 15-2 *Driver eDirectory Attributes*

Driver eDirectory Attribute	Designer Property or View
DirXML-InputTransform	Policy Set View->Input Transformation
DirXML-OutputTransform	Policy Set View->Output Transformation
DirXML-MappingRule	Policy Set View->Schema Mapping
DirXML-Driver Filter	Policy Set View->Driver Filter
DirXML-ConfigValues	Driver Properties->Global Configuration Values
DirXML-DriverTraceLevel	Driver Properties->Driver Log Level->Driver Log Level
DirXML-LogEvents	Driver Properties->Driver Log Level->Log Specific Events
DirXML-LogLimit	Driver Properties->Driver Log Level->Log Limit
DirXML-ConfigManifest	Driver Properties->Driver Manifest
DirXML-JavaModule	Driver Properties->Driver Configuration->Driver Module: Java
DirXML-NativeModule	Driver Properties->Driver Configuration->Driver Module: Native
DirXML-DriverImage	Driver Properties->iManager Icon
DirXML-TraceLevel	Driver Properties->Trace->Trace Level
DirXML-TraceFile	Driver Properties->Trace->Trace File
DirXML-TraceSizeLimit	Driver Properties->Trace->Trace File Size Limit
DirXML-TraceName	Driver Properties->Trace->Trace Name
DirXML-DriverCacheLimit	Driver Properties->Driver Configuration->Authentication->Driver Cache Limit
DirXML-ShimAuthID	Driver Properties->Driver Configuration->Authentication->User ID
DirXML-ShimAuthServer	Driver Properties->Driver Configuration->Authentication->Connection Information
DirXML-ShimAuthPassword	Driver Properties->Driver Configuration->Authentication->Set Password
DirXML-ShimConfigInfo	Driver Properties->Driver Configuration->Driver Configuration->Driver Parameters

Driver eDirectory Attribute	Designer Property or View
DirXML-DriverStartOption	Driver Properties->Driver Configuration->Startup Option
DirXML-ECMAScript	Driver Properties->Driver Configuration->ECMAScript

Table 15-3 Channel eDirectory Attributes

Channel eDirectory Attribute	Designer View
DirXML-EventTransformationRule	Policy Set View->Event Transformation
DirXML-MatchingRule	Policy Set View->Matching
DirXML-CreateRule	Policy Set View->Creation
DirXML-PlacementRule	Policy Set View->Placement
DirXML-CommandTransformation	Policy Set View->Command Transformation

Table 15-4 Job eDirectory Attributes

Job eDirectory Attribute	Designer View
XmlData	Job Editor, XML cannot be edited directly only through Job Editor UI
DirXML-ServerList	Job Editor
DirXML-Scope	Job Editor
DirXML-EMailTemplates	Job Editor
DirXML-EMailServer	Job Editor
DirXML-NamedPasswords	Job Editor
DirXML-TraceName	Job Properties -> Trace
DirXML-TraceFile	Job Properties -> Trace
DirXML-TraceSizeLimit	Job Properties -> Trace
DirXML-TraceLevel	Job Properties -> Trace

Table 15-5 Resource eDirectory Attributes

Resource eDirectory Attribute	Designer View
DirXML-ContentType	Read only, cannot be edited set at creation time of object
DirXML-DirXMLData	Resource Editor
DirXML-NamedPasswords	Resource Editor

15.6 Error Messages and Solutions

To view error messages along with their possible solutions associated with importing and deploying files, see [Section 24.4, “Deploying Identity Manager Objects,”](#) on page 495.

When you create a project, it's vital to keep track of how the driver works and how it's implemented into a network. The Document Generator helps you quickly generate customized documentation for your Designer projects. These documents can save you weeks or months of gathering and writing driver specifications and their implementations. To generate a document, choose a document style (it can be the default style that comes with Designer or one that you customize) and a Designer project or portion of a project. The Document feature combines the information and structure of the selected style with the project information in order to generate customized project documentation.

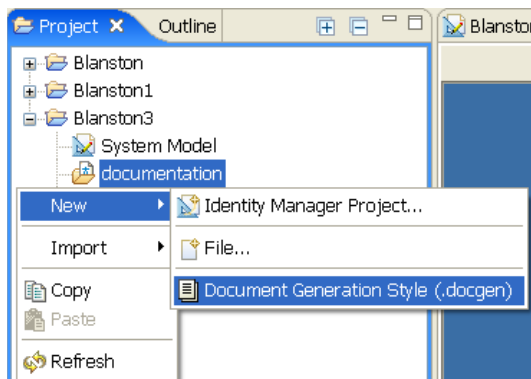
Designer comes with a default document style so you won't have to create a document from scratch. This default style contains everything that you have placed in a project through Designer. You must first use this default style to create your own document style for the project you are working on, then you can either use it as it is or customize it to meet your particular needs, including or excluding information as needed. After you have edited the style to your liking, you can also use it to document your other projects. There is an advanced editing feature that allows you to create your own sections for adding information that you did not create in Designer.

- ◆ [Section 16.1, “Creating a Document Style,” on page 299](#)
- ◆ [Section 16.2, “Editing a Document Style for Your Needs,” on page 300](#)
- ◆ [Section 16.3, “Generating a Document,” on page 302](#)
- ◆ [Section 16.4, “Using Your Template Style for Other Projects,” on page 306](#)
- ◆ [Section 16.5, “Customizing Styles to Include or Exclude Information,” on page 311](#)
- ◆ [Section 16.6, “Advanced Editing of a Document Style,” on page 320](#)
- ◆ [Section 16.7, “Troubleshooting Issues When Generating Documents,” on page 337](#)

16.1 Creating a Document Style

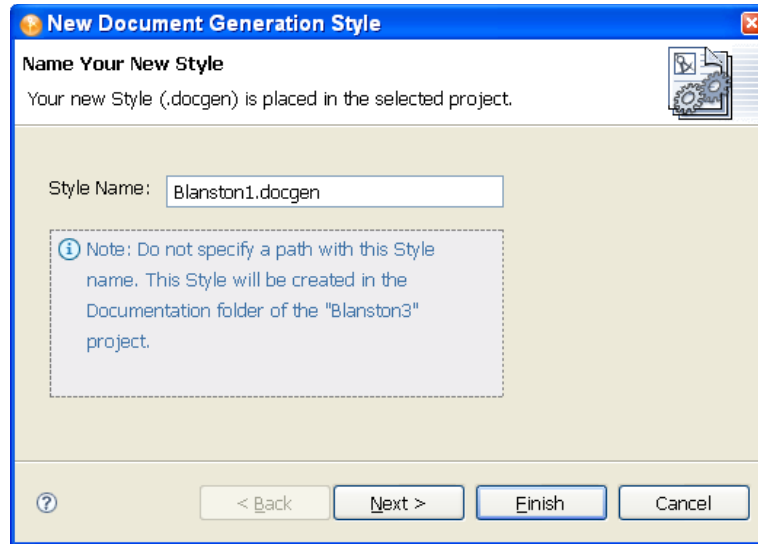
A document style allows you to define how you want your project information to look. In a matter of minutes, you can generate a document that contains all elements that you have placed in a project and define a document style to designate how the information looks, as well as what information you want in a document.

- 1 Select a project in the Project view, then right-click the *Documentation* icon.
- 2 Click *New > Document Generation Style (.docgen)*.



The Document Generation Style (.docgen) is the default style that is provided with Designer. You use this as the template for your own .docgen style.

- 3 Specify a name for the style, with a .docgen extension, or use the default name of the project you are presently in, then click *Finish*.



A .docgen file is created that you can use as the basis for designing your own style template. The .docgen template is placed in the Style Editor view for your modification (see [Section 16.2, “Editing a Document Style for Your Needs,” on page 300](#)). You can use the defined elements in your new style template again and again.

- 4 Use this basic document style to generate documentation for a project, or customize it for your needs.
 - ♦ [Section 16.3, “Generating a Document,” on page 302](#)
 - ♦ [Section 16.2, “Editing a Document Style for Your Needs,” on page 300](#)
 - ♦ [Section 16.4, “Using Your Template Style for Other Projects,” on page 306](#)
 - ♦ [Section 16.5, “Customizing Styles to Include or Exclude Information,” on page 311](#)
 - ♦ [Section 16.6, “Advanced Editing of a Document Style,” on page 320](#)

16.2 Editing a Document Style for Your Needs

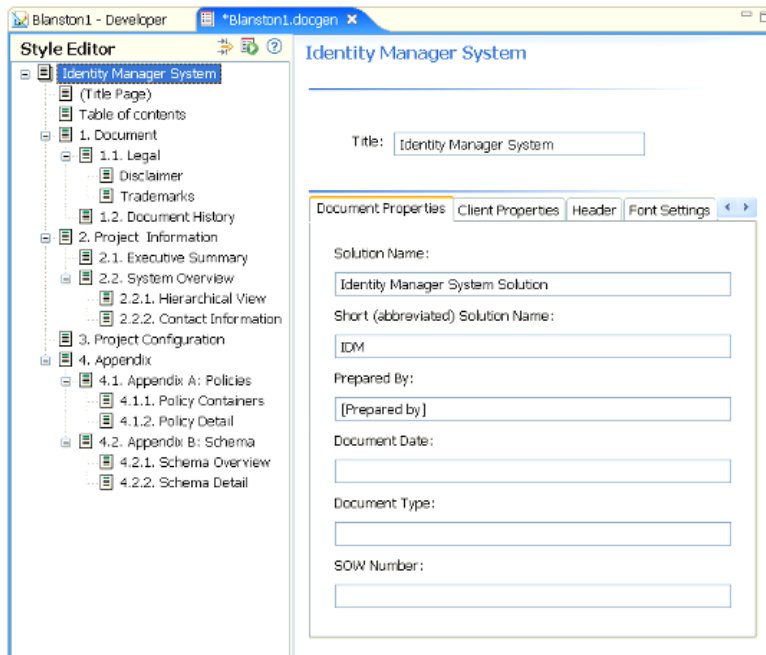
Now that you have created a .docgen template, you can edit the file to meet your documentation needs. The .docgen file appears under the documentation directory in the Project view. Double-click the .docgen file to bring up the template in the Style Editor.

You can add or modify information in the style template, which in turn affects the documents that you generate. The information that you can customize through the Style Editor appears in the Style Editor view. For more detailed editing, see [Section 16.6, “Advanced Editing of a Document Style,” on page 320](#).

- ♦ [Section 16.2.1, “Editing a Style Template,” on page 301](#)
- ♦ [Section 16.2.2, “Editing Sections of a Style,” on page 301](#)

16.2.1 Editing a Style Template

- 1 Select a project in the Project view, then right-click the *Documentation* icon.
- 2 Double-click the `.docgen` file. The file appears in the Style Editor view.



- 3 Use the Style editor to edit sections of a style or to modify the style according to your needs. The Style editor is divided into two parts, beginning with the Identity Manager and working through the Appendixes. When you click an item under the *Style Editor* section, the right portion of the view changes to display the information associated with the heading. For example, clicking *Disclaimer* under the *Document > Legal* heading allows you to edit the disclaimer content.
- 4 Save your changes. Your changes are saved when you close the Style Editor, or when you click *Save*.

16.2.2 Editing Sections of a Style

- 1 Select an item (for example, Executive Summary) in the Style Editor view.
- 2 Enter data or make changes in the left pane.
- 3 Select other items as appropriate and make changes. The information in the left pane varies, depending on items that you select.

The main areas that you need to pay attention to are the information found under Identity Manager System (Title Page and Table of Contents), Document, Legal, Disclaimer, Trademarks, and Executive Summary.

- 4 Save your changes. Your changes are saved when you close the Style Editor, or when you click *Save*.

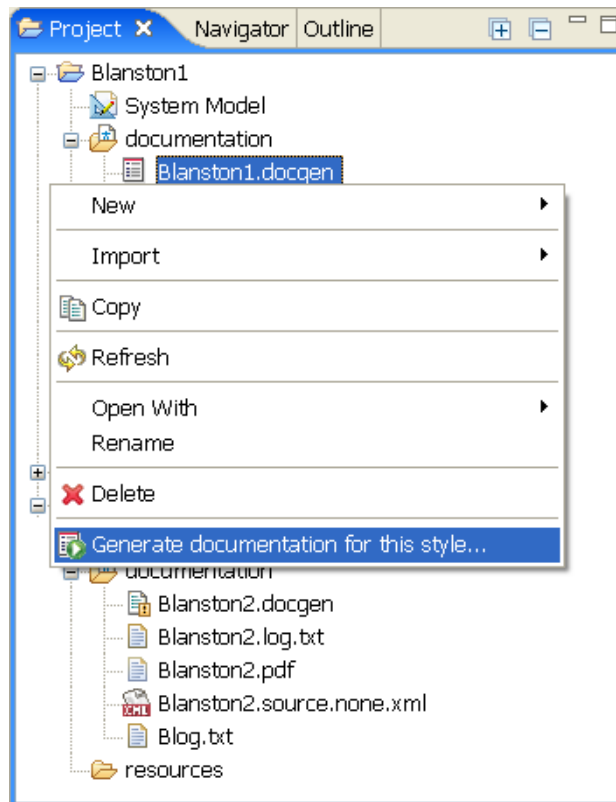
- 5 Use this document style to generate documentation for a project, or continue to customize it for specific documentation needs.
 - ♦ [Section 16.3, “Generating a Document,” on page 302](#)
 - ♦ [Section 16.4, “Using Your Template Style for Other Projects,” on page 306](#)
 - ♦ [Section 16.5, “Customizing Styles to Include or Exclude Information,” on page 311](#)
 - ♦ [Section 16.6, “Advanced Editing of a Document Style,” on page 320](#)

16.3 Generating a Document

- 1 (Conditional) If you haven’t yet created a Designer Project, create one.
 - 1a Select *File > New > Identity Manager Project*.
 - 1b Provide a project name, then click *Finish*.
- 2 (Conditional) If you haven’t yet created a document style that you want to use as a template for your documentation, create one. See [“Creating a Document Style” on page 299](#) and [“Editing a Document Style for Your Needs” on page 300](#).
- 3 In the Project view, select and expand a project, then right-click the .docgen icon and select *Generate Documentation for This Style*.

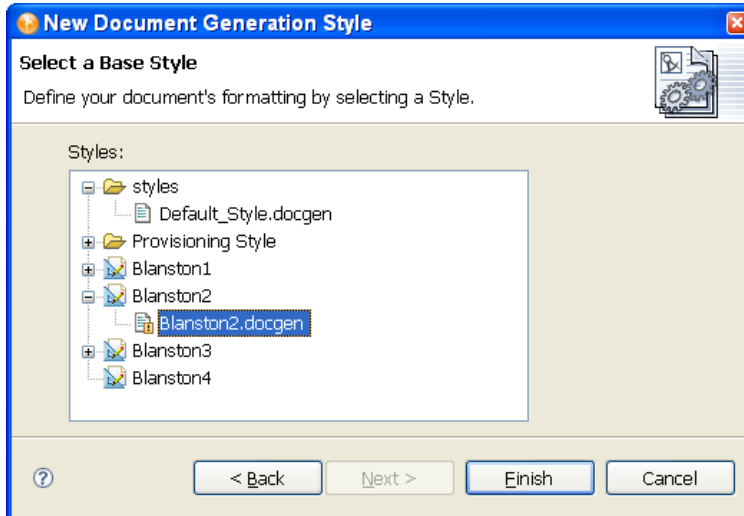
or

Select *Project > Documentation > Generate Documentation*.

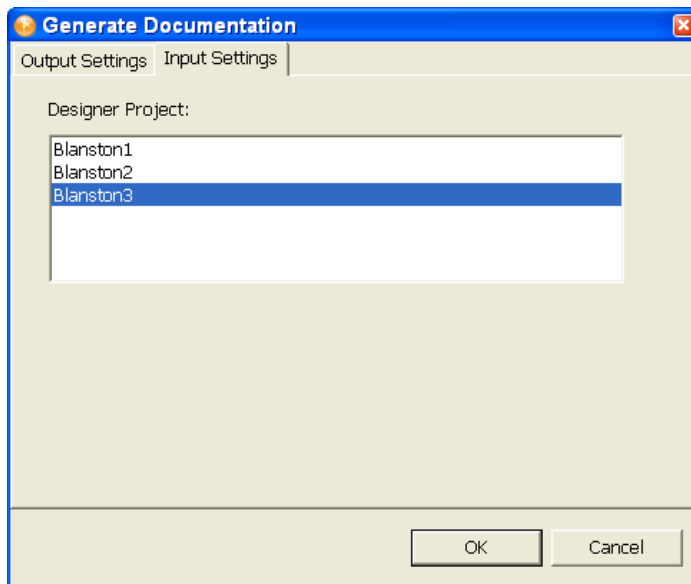


You can also expand the Documentation folder and click the .docgen file to open the file in the Style Editor, then click the *Document Generation* icon to the right of the Style Editor heading.

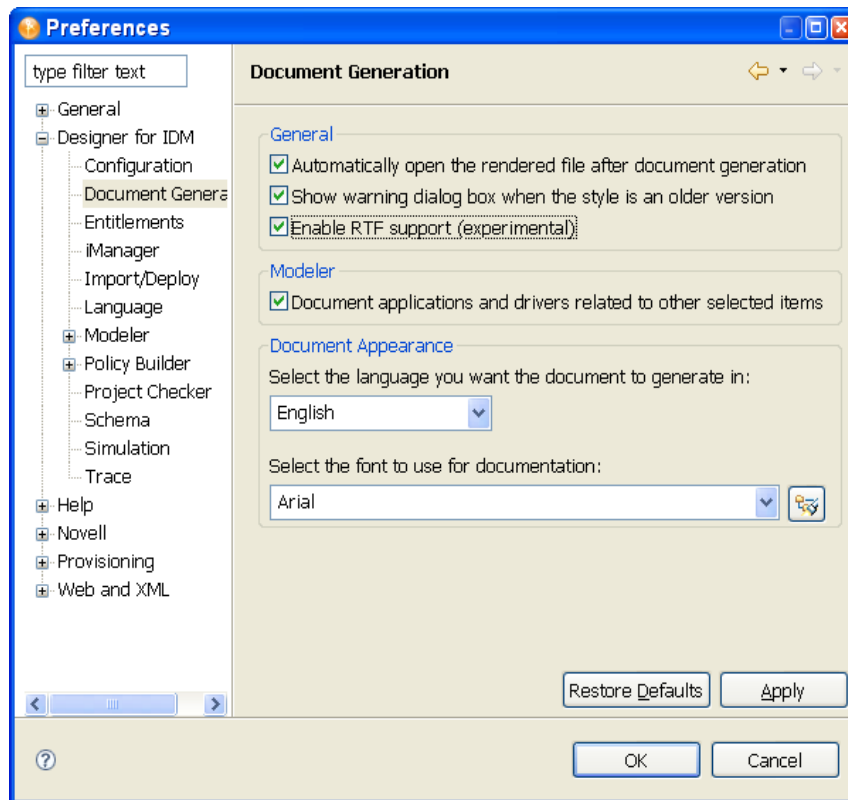
- 4 (Conditional) If the Documentation folder you selected does not contain a .docgen file, you are asked to select a Base Style. Select a .docgen style, then click *Next*.



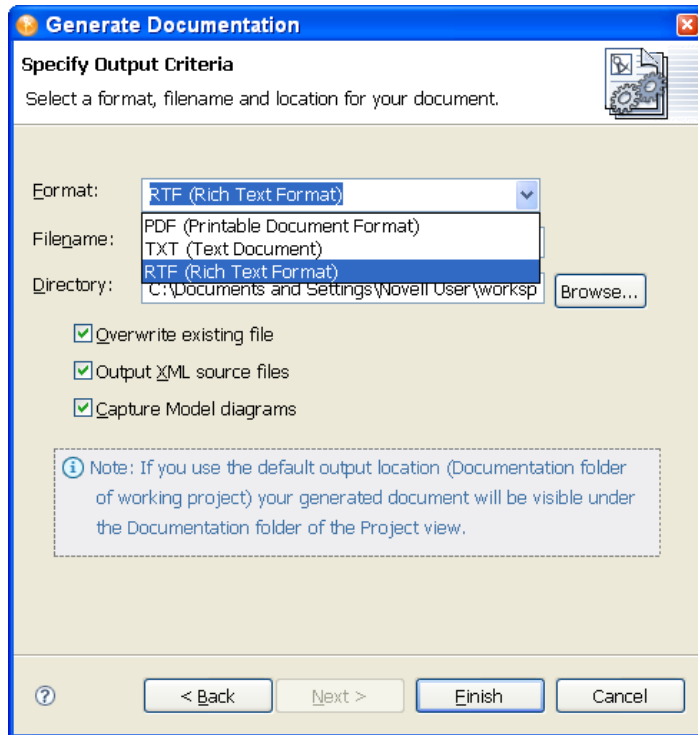
- 5 In the Generate Documentation window, click the *Input Settings* tab, select the project you want to create documentation for using this style template, then click the *Output Settings* tab.



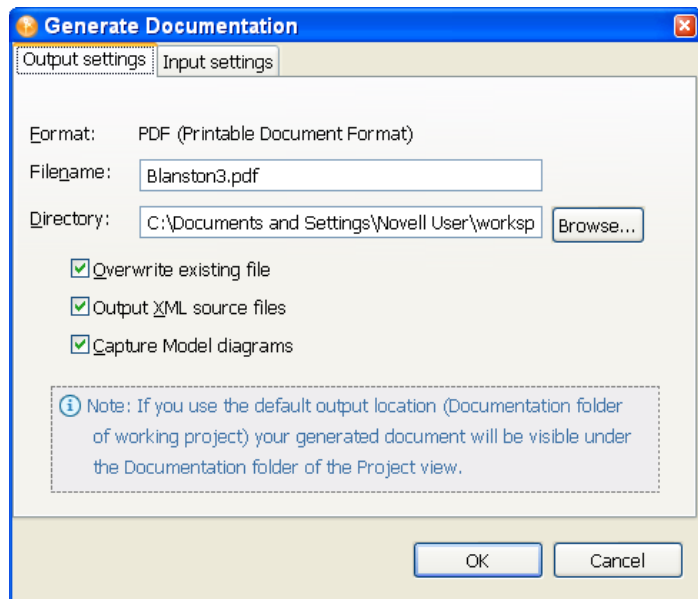
- 6 (Conditional) This release of Designer includes the ability to generate documents to RTF (Rich Text Format). If you want to enable this functionality, click *Window > Preferences* to bring up the Preferences window. Then, under *Designer for IDM*, select *Document Generation*.



By selecting *Enable RTF support (experimental)*, you can select the RTF format when creating documents.



7 On the Output Settings page, fill in the needed information.



- ♦ Select the format for the file. If you have enabled rich text formatting, you can select *PDF (Printable Document Format)*, *TXT (Text Document)*, or *RTF (Rich Text Format)*.
- ♦ Select the name you want to call the file, or keep the default name. If you are generating the whole document, the default name often suffices. If you are generating a document for

a section, such as an Executive Summary, name the file to reflect the section you documented.

- ♦ Select the directory where you want to store the document. If you use the default output location that appears in the Directory field, your generated document is visible under the Documentation folder of the Project View.
 - ♦ To overwrite an existing .pdf file, select the *Overwrite existing file* option (selected by default).
 - ♦ To see the XML source files, select the *Output XML source files* option (selected by default).
 - ♦ To include the model diagrams in the PDF file, select the *Capture Model diagrams* option (selected by default).
- 8 Click *Finish* to generate the document. The document appears in the current Project > Documentation folder unless otherwise specified.

PDF files must be viewed through a PDF viewer, such as Adobe Acrobat*. If Adobe Acrobat is presently installed on your workstation, Designer launches the document in Acrobat. RTF files can be viewed in any word processor that can handle Rich Text Formatted files.

- 9 If you receive an “Out of Memory” error message while generating documentation, see [“Issues When Generating Documents” on page 509](#).

16.4 Using Your Template Style for Other Projects

To generate documentation for any project, you can use the default style provided with Designer or you can use your own customized styles.

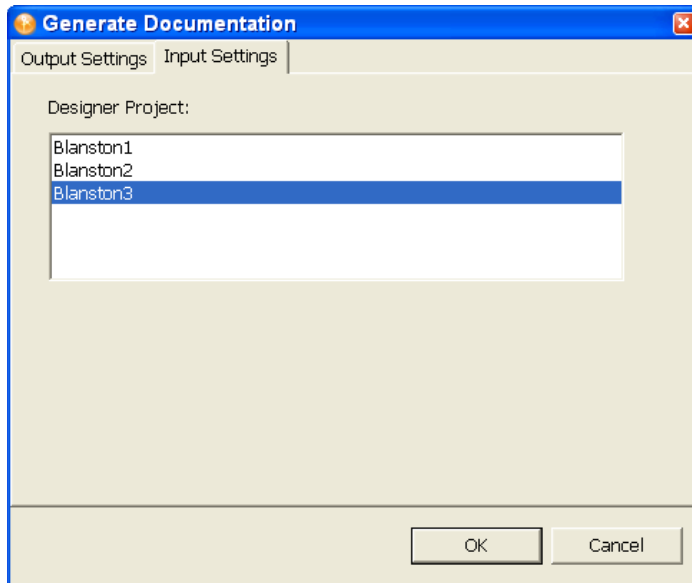
For example, suppose you have created a Blanston2 style. You customize the Blanston2 style to fit your needs, and you save it in the Documentation folder as Blanston2.docgen as outlined in [Section 16.1, “Creating a Document Style,” on page 299](#). You can use this style to create documentation for the Blanston2 project or for other projects.

- ♦ [Section 16.4.1, “Start with the Style and Choose the Project,” on page 306](#)
- ♦ [Section 16.4.2, “Start with the Project and Choose the Style,” on page 307](#)
- ♦ [Section 16.4.3, “Document a Section of the Style,” on page 308](#)
- ♦ [Section 16.4.4, “Document Multiple Sections of the Style,” on page 310](#)

16.4.1 Start with the Style and Choose the Project

- 1 With the Blanston2 project open in the Project view, right-click the Blanston2.docgen file in the Documentation folder.
- 2 Click *Generate Documentation for This Style*.

- 3 In the Generate Documentation window, click the *Input Settings* tab, then select the project you want to create documentation for with this style template (in this example, Blanston3).

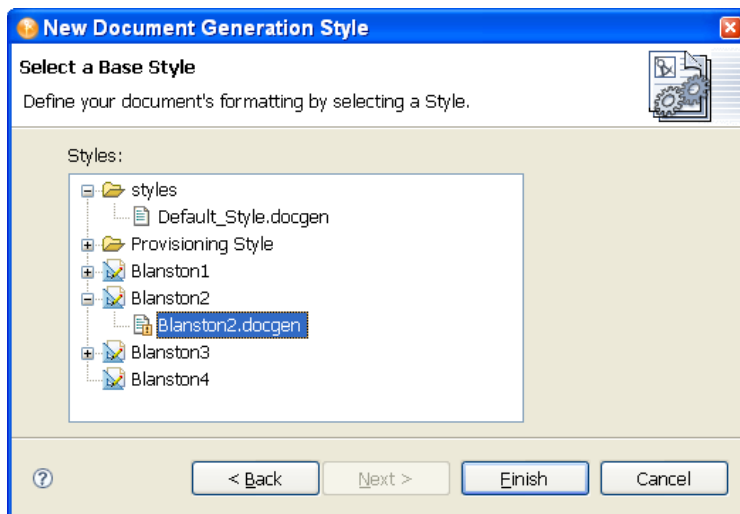


- 4 Select the *Output Settings* tab to open the Specify Output Criteria window, fill in the necessary information, then click *OK*.

For information on using this style to generate documentation, see [“Generating a Document” on page 302](#).

16.4.2 Start with the Project and Choose the Style

- 1 With two projects open (for example, Blanston1 and Blanston2), go to the Project view and click *Documentation > New > Documentation Generation Style (.docgen)*.
- 2 Name the new *.docgen* file (this example is *Blanston2.docgen*), then click *Next*.
- 3 Select a base style (for this example, *Blanston.docgen*), then click *Finish*.



The `Blanston2.docgen` file opens in the Style Editor. Because a style usually contains project-specific information, such as the company name, use the above information to see if the template transferred. If you often use the `Blanston2.docgen` style with the `Blanston1` project, you might want to copy the `Blanston2` style to the `Blanston1 Documentation` folder, or copy the `.docgen` file to the styles template library. For more information, see [Section 16.6.5, “Saving a Style File in the Styles Template Library,” on page 337.](#)

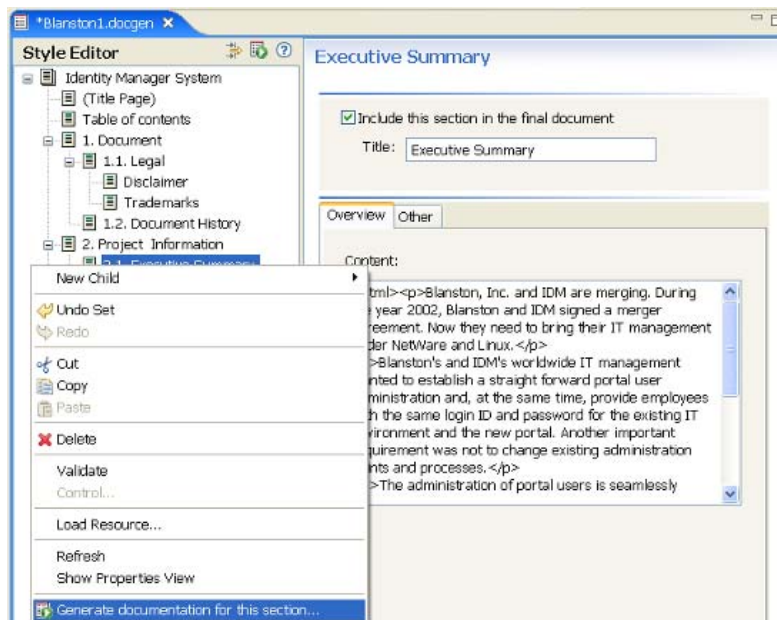
- 4 You can click the *Generate Documentation* icon in the Style Editor to generate a document. See [Section 16.3, “Generating a Document,” on page 302.](#)

NOTE: If you see an exclamation mark in the `.docgen` file icon, it means the file does not contain new features and needs to be updated. To do this, create a new `.docgen` file, copy any changes to the new file from the older `.docgen` file, then delete the older `.docgen` file.

16.4.3 Document a Section of the Style

Instead of generating a document for the entire project, you can generate a document for a selected section of a project.

- 1 With the project's `.docgen` file open in the Style Editor pane, right-click a section of the style.
- 2 Select *Generate Documentation for This Section.*
- 3 In the Generate Documentation window, type a different project name in the *Filename* field (for example, `ExecutivSumofMerger`), then click *OK.*

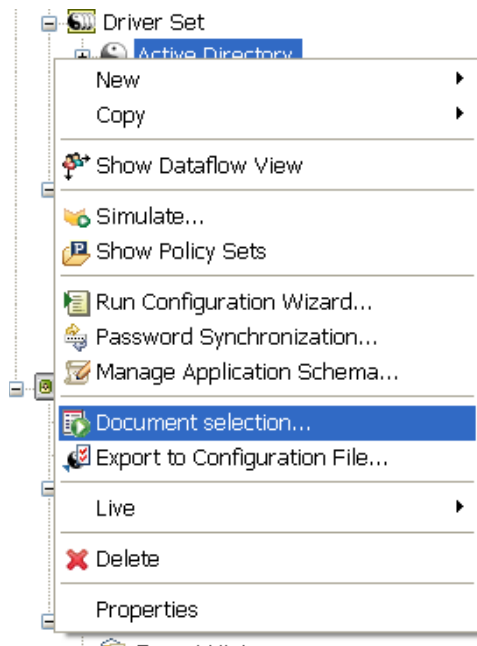


Specify which portion or portions of the project you want to include in the generated document.

You can document domains, Identity Vaults, driver sets, drivers, and applications using the Modeler view or the Outline view (use the `Ctrl` key to select multiple items). Document generation also ties in with schema notes, classes and attributes. You can find out more about this in [Chapter 11, “Managing the Schema,” on page 187.](#)

For example:

- 1 To document a specific driver in a project, right-click the driver in the Modeler or Outline view and select *Document Selection*.

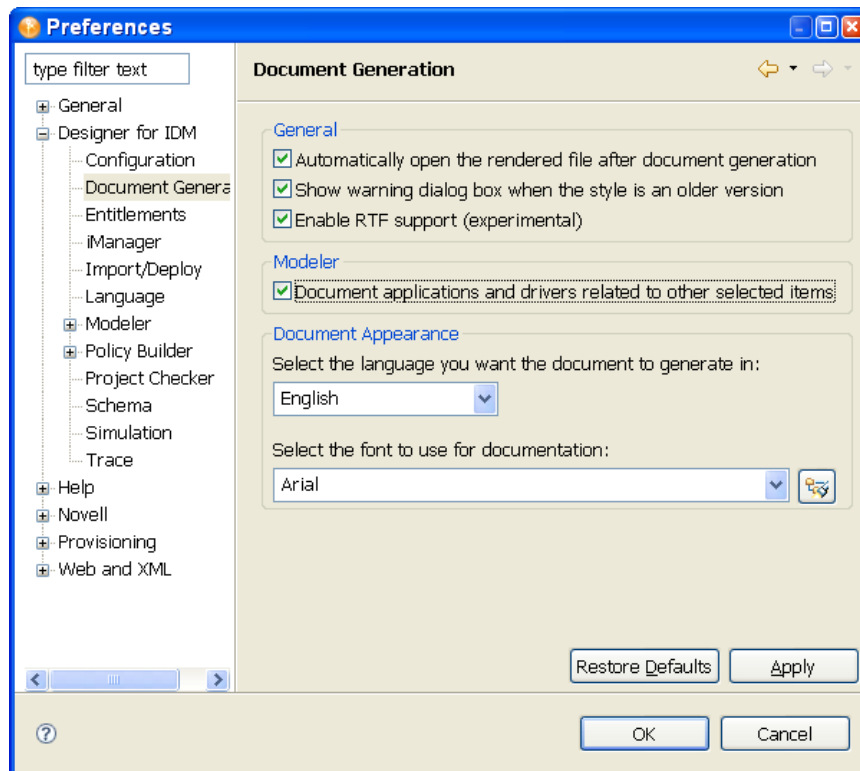


- 2 Select the `.docgen` style sheet for the document, give the document a filename, and generate the driver document.

By default, documenting an application includes the connected driver (and its related objects). Likewise, documenting a specific driver includes its connected application. However, you can change this behavior in the Documentation Generation's Preferences page.

- 1 Click *Window > Preferences* to bring up the Preferences window.

2 Under *Designer for IDM*, select *Document Generation*.



Under the *Modeler* heading, the *Document applications and drivers related to other selected items* option is selected by default, which means that directly related items are included in the documentation. For example, by default, documenting a driver set includes the direct children (the applications) as well as some information of the direct parents (the Identity Vault and domain) to give context to the driver set. Deselecting this option excludes direct children of the selected item.

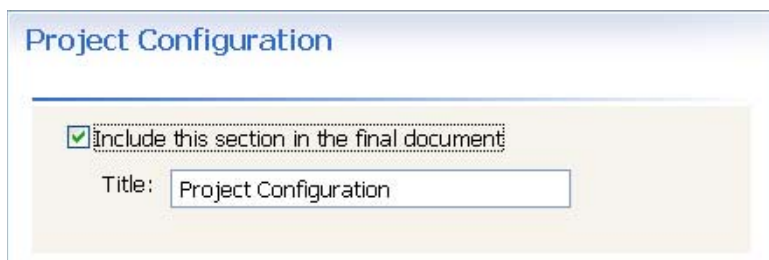
3 Select or deselect the options you want, then click *OK*.

16.4.4 Document Multiple Sections of the Style

If necessary, you can generate only selected sections so that peers can help you with information in the selected sections.

1 If you have not already done so, double-click the `.docgen` file to bring up the template in the Style Editor.

- 2 Select or deselect section headings. Each section and child section has a check box entitled *Include this section in the final document*. By default the box is checked, as shown below.



To limit the sections you want to document, deselect the check boxes in the sections you don't want to generate.

- 3 Click the Generate Document icon to the right of the Style Editor heading and generate your document.
Give the document a unique name to reflect the type of information it includes.

16.5 Customizing Styles to Include or Exclude Information

If you import a project with the entire Identity Vault schema and run document generation on that project, you can have an extra 600 pages or more of schema information. This happens because you have all of the Identity Vault's base schema along with the added schemas are included in the document.

Novell does not recommend that you document all Identity Vault schema unless you need to. The main problem in doing this is that your workstation can run out of memory. If you do have memory issues when generating documentation, see [Section 24.5, "Issues When Generating Documents," on page 509](#).

These are the general ways you can control the amount of information you generate:

- ♦ [Section 16.5.1, "Limiting Documentation for the Identity Vault Schema and Application Schema," on page 311](#)
- ♦ [Section 16.5.2, "Deleting Extra Identity Vault Schema from Your Generated Report," on page 315](#)
- ♦ [Section 16.5.3, "Using Project Configuration to Limit Information," on page 318](#)

16.5.1 Limiting Documentation for the Identity Vault Schema and Application Schema

You can limit what you document for the Identity Vault and application schema that you have imported. This is accomplished by selecting the Appendix B: Schema heading in the Style Editor view. This brings up the Appendix B section template in the right-hand side of the Style Editor view.

Figure 16-1 The Appendix B: Schema Section Template

Appendix B: Schema

Include this section in the final document
Title:

i Identity Vault schema is NOT available to render unless marked for documentation using the schema tool, or schema has been imported or manually created in Designer.

Document all custom Identity Vault Schema even if not marked for documentation

i Application schema will NOT be documented unless individual classes have been marked to be documented or the override option below is used.

Document all Application Schema even if not marked for documentation

The Appendix B: Schema section has three boxes:

- ◆ *Include this section in the final document*
- ◆ *Document all custom Identity Vault Schema even if not marked for documentation*
- ◆ *Document all Application Schema even if not marked for documentation*

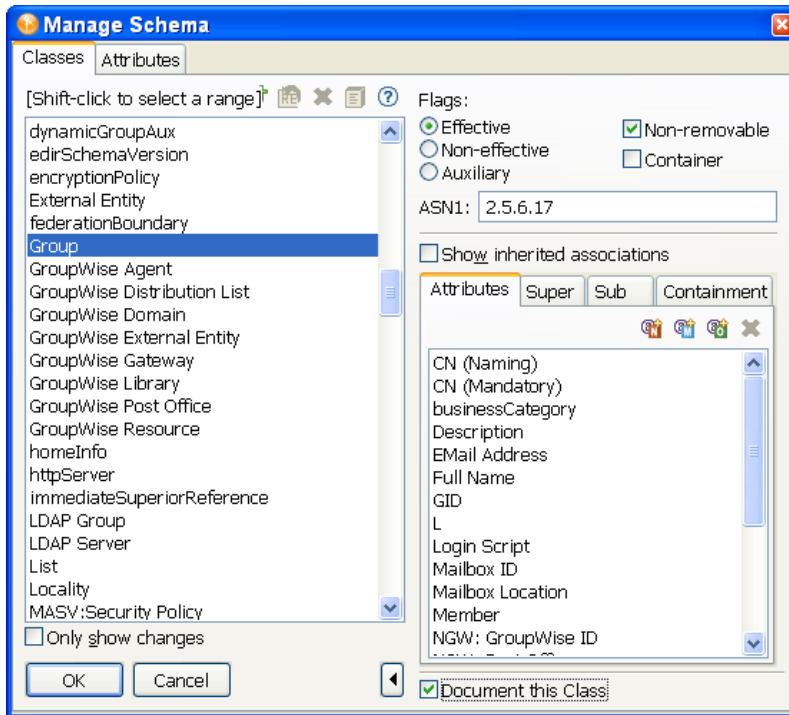
Include this section. The *Include this section in the final documentation* check box allows you to include or not include Appendix B information in the documentation. The default is checked to include this information. Deselect the check box if you do not want to include application or Identity Vault schema in the document.

Document all custom Identity Vault Schema. The *Document all custom Identity Vault Schema even if not marked for documentation* check box allows you to control documentation on Identity Vault schema that you import from the Identity Vault or that you create.

To select the Identity Vault schema you want to document:

- 1** Right-click the Identity Vault icon in the Modeler view and select *Manage Vault Schema*.
- 2** In the Manage Schema window, select the classes that you want to document, then click *Document this Class*.

You can select multiple classes by using the Ctrl or the Shift key.



3 Click *OK*.

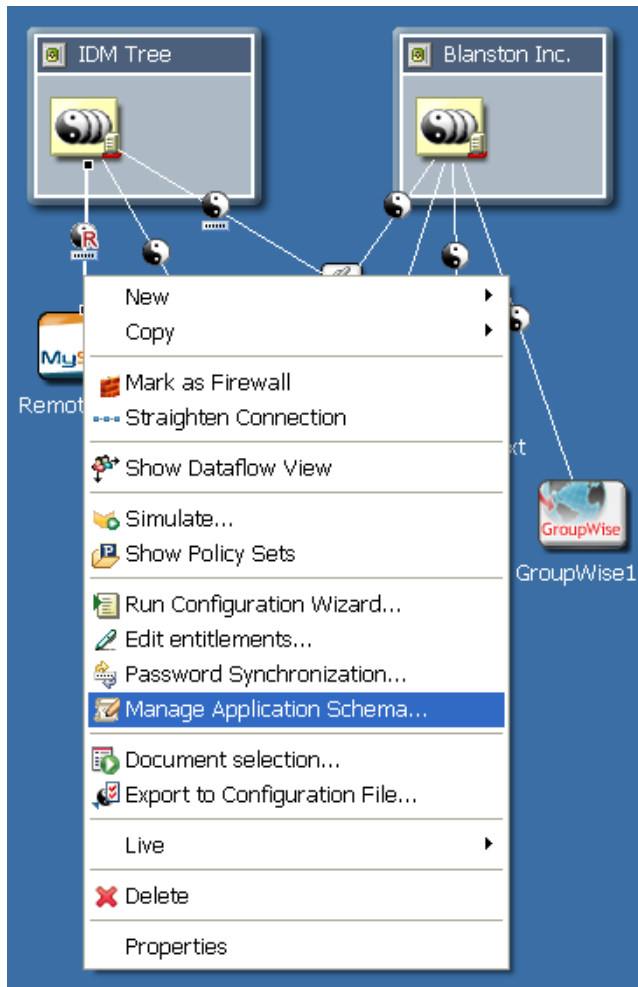
If you select the *Document all custom Identity Vault Schema even if not marked for documentation* box in the Appendix B: Schema section, it effectively overrides any Identity Vault schema selections you make through the Schema Import Wizard. To summarize:

- ◆ If you select the *Document all custom Identity Vault Schema* box in the Appendix B: Schema section, the entire Identity Vault schema is documented.
- ◆ If you deselect the *Document all custom Identity Vault Schema* box in the Appendix B: Schema section, only the Identity Vault schema that you have selected through the *Manage Vault Schema* option is documented.
- ◆ If you do not select the *Document all custom Identity Vault Schema* box in the Appendix B: Schema section and you do not select any Identity Vault schema through the Manage Schema window, then no schema is documented.

Document all Application Schema. The third check box on the Appendix B: Schema page, *Document all Application Schema even if not marked for documentation*, refers to your ability to choose which driver's application classes you want to document. If you select the *Document all Application Schema* box in the Appendix B: Schema section, then all application schema is documented. You can also select the application schema that you want to document instead of documenting the entire schema. Use the following example for more information.

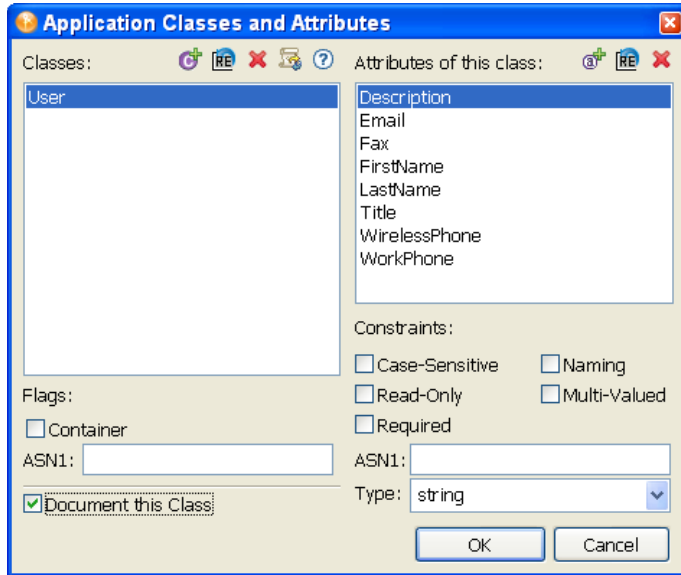
- 1** To select a driver's application classes, bring up a project in the Modeler.

2 Right-click the driver icon and select *Manage Application Schema*.



3 The Application Classes dialog box appears with a list of classes that you can add to or delete. The right side shows the attributes that already have been created for the class. If you do not see

any classes in the Application Classes window, it means that there are presently no classes created for this driver.



- 4 In the lower right corner is a check box entitled *Document this Class*. When you select this box, you can choose which application classes you want to document through the Document Generator instead of documenting the entire schema.

If you select the *Document all Application Schema even if not marked for documentation* box in the Appendix B: Schema section, it effectively overrides any driver application schema selections you make through the Application Classes window. To summarize:

- ♦ If you select the *Document all Application Schema* box in the Appendix B: Schema section, the entire application schema is documented.
- ♦ If you deselect the *Document all Application Schema* box in the Appendix B: Schema section, only the application schema that you have selected through the Application Classes window is documented.
- ♦ If you do not select the *Document all Application Schema* box in the Appendix B: Schema section and you do not select any application schema through the Application Classes window, then no schema is documented.

16.5.2 Deleting Extra Identity Vault Schema from Your Generated Report

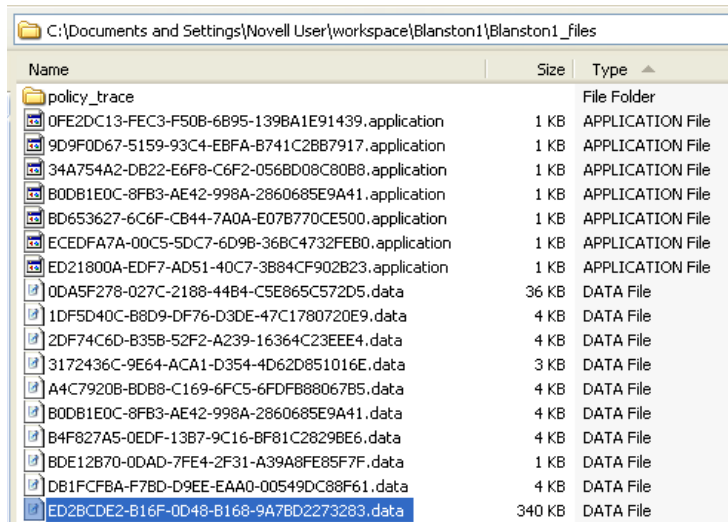
If you import a project with the entire Identity Vault schema and run document generation on that project, it can result in an extra 600 pages or more of schema information. This happens because all of the Identity Vault's base schema is included in the document.

Follow these steps if you do not want to document all of the Identity Vault's base schemas:

- 1 Go into the workspace folder for this project and locate the `.data` file.

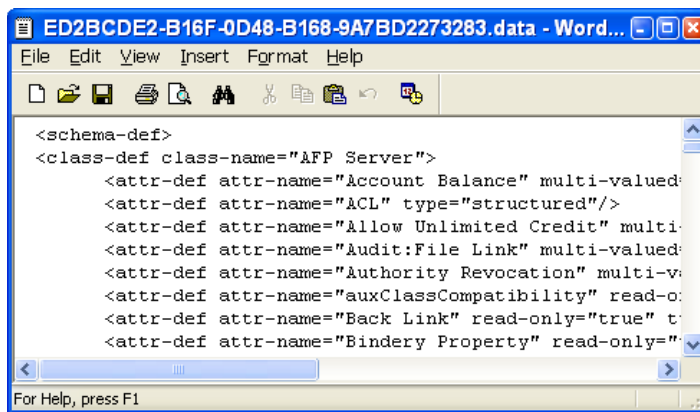
For example, on a Windows workstation, the `Blanston1_files` directory is found at `C:\Documents and Settings\Novell`

User\workspace\Blanston1\Blanston1_files. Click *Type* to see all of the .data files shown together. The largest .data file holds the base Identity Vault schema.



If you have more than one Identity Vault in your project, you will see more than one large .data file in the project's folder.

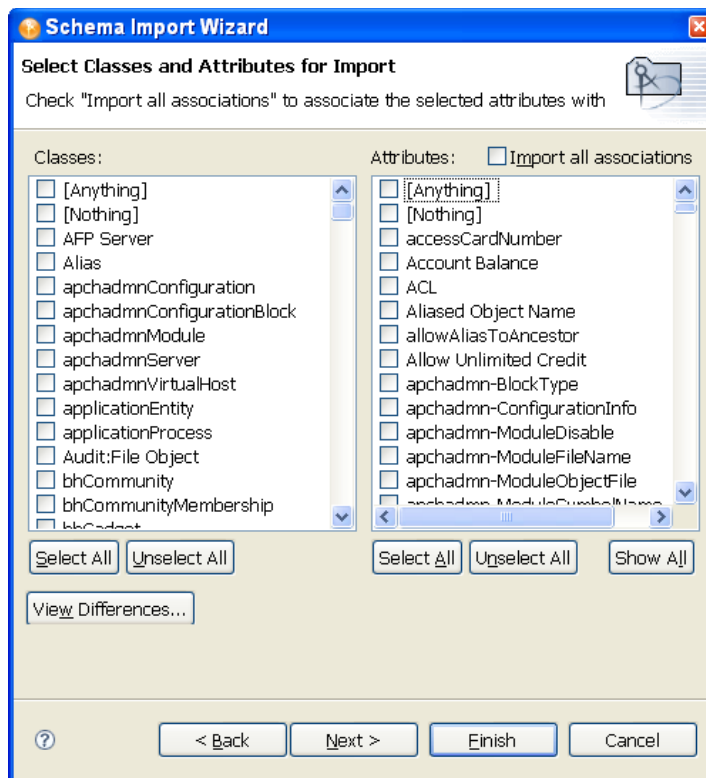
- Open the .data file in a text-based editor (in this instance, Wordpad) to make sure you have the schema file. Schema files have `<schema-def>` at the top of the file.



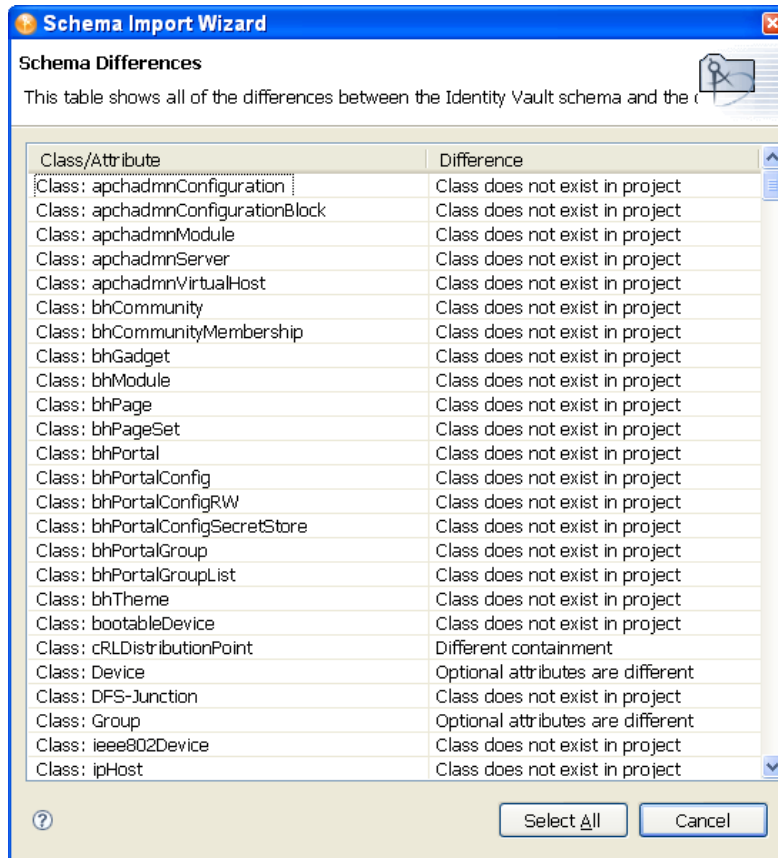
Now suppose you only want to import the Identity Vault schema differences between your project and the Identity Vault.

- To get the schema differences, delete any large .data files,
- Re-import the project's schema:
 - Bring up the project in Designer's Modeler view. Right-click the Identity Vault and select *Live Operations > Import Schema*.
 - At the Select Source for Import page, select *Import from eDirectory* if you are connected to an actual Identity Vault, or select *Import from .sch File* if you have saved the schema files locally.
 - In the Import from eDirectory dialog box, type the host name, username, and password connection information, then click *Next*.

4d On the Select Classes and Attributes for Import page, click *View Differences*.



On the Schema Differences page, you see the schema differences between Identity Vault and your project.



4e Click *Select All*. This brings up the Select Classes and Attributes for Import page with the differences now selected under the Classes and Attributes headings. By default, the *Import All Associations* box is also selected. Leave it selected, then click *Finish*.

4f On the Import Summary page, click *OK*.

5 Bring up the Manage Schema window and select the classes and attributes you want to document, as described in [“Limiting Documentation for the Identity Vault Schema and Application Schema”](#) on page 311.

16.5.3 Using Project Configuration to Limit Information

The Project Configuration heading allows you to include or deselect information on:

- ◆ Identity Vault
- ◆ Driver Sets
- ◆ Drivers
- ◆ General

The following table shows what type of information can be included or excluded in these areas.

Table 16-1 *Project Configuration*

Identity Vault	Information to Select or Deselect
Selected	Include host information
Selected	Include username information
Deselected	Include deployment context information
Selected	Include driver set names
Selected	Include policy library on Identity Vaults
Deselected	Include e-mail notification templates
Deselected	Include XML source while documenting policies under the policy library
Deselected	Include XML source while documenting credential provisioning objects under the policy library
Deselected	Include XML source while documenting mapping table objects under the policy library
Driver Set	Information to Select or Deselect
Selected	Include server information associated with the driver set
Selected	Include driver set Global Configuration Value (GCV)
Selected	Include the policy library on driver sets
Selected	Include job objects on driver sets
Deselected	Include XML source for policies under the policy library
Deselected	Include XML source for credential provisioning objects under the policy library
Deselected	Include XML source for mapping table objects under the policy library
Deselected	Include XML source for job objects
Driver	Information to Select or Deselect
Selected	Include the driver filter policy
Selected	Include policy set
Selected	Include server-specific information for this driver
Selected	Include Remote Loader configuration
Selected	Include entitlements
Selected	Include credential provisioning
Selected	Include mapping table
Selected	Include ECMA script resource object
Selected	Include job objects
Selected	Include XML source when documenting entitlement objects
Deselected	Include XML source when documenting credential provisioning objects

Identity Vault	Information to Select or Deselect
Deselected	Include XML source when documenting mapping table objects
Deselected	Include XML source when documenting job objects
General	Information to Select or Deselect
Deselected	Include passwords
Selected	Page break after this section

WARNING: Credential provisioning the XML source might contain passwords that will be displayed in clear text. If selected, passwords will be displayed in clear text and documentation will include all passwords in the project.

16.6 Advanced Editing of a Document Style

In addition to selecting and deselecting the content of a document style, you can also change the layout and usability of your document style. You do this by editing the attributes that are associated with certain sections. You can also create additional sections for your documents as you see fit.

- ◆ [Section 16.6.1, “What’s In the Advanced Editing Mode,” on page 320](#)
- ◆ [Section 16.6.2, “A Walk-through Tutorial,” on page 327](#)
- ◆ [Section 16.6.3, “Selecting a Language for Generated Documents,” on page 334](#)
- ◆ [Section 16.6.4, “Double-Byte Font Support,” on page 335](#)
- ◆ [Section 16.6.5, “Saving a Style File in the Styles Template Library,” on page 337](#)

16.6.1 What’s In the Advanced Editing Mode


The Advanced Editing icon  lets you toggle between simple editing and advanced editing modes. By using the advanced editing mode, you can define information and a structure that is different from the default template already attached to a predefined style. In this example, the Title Page template is shown in its XSL format, which you must maintain.

Figure 16-2 Viewing a Predefined Template

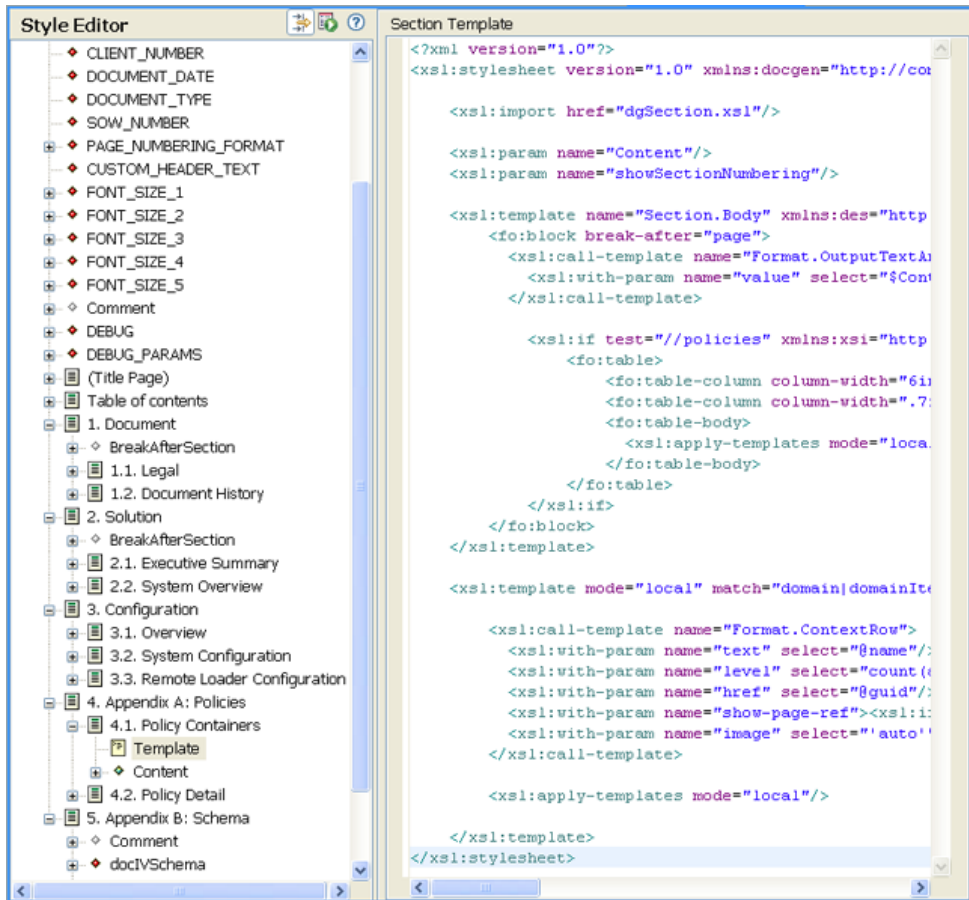










Table 16-2 Style Editor Legend

Icon	Name	Description
	Green Page	A green page means it's a titled section. The title appears when you generate the document.
	Grey Page	A grey page means it's not a titled selection. These pages are also in parentheses; for example, (Title Page).
	White Page	A white page means this section is disabled and will not be included when you generate the document.
	Template	A yellow template page gives specific format and styles that are included when you generate a document.
	Global Attribute	A global (red) attribute means it is passed down to every section below it (all children sections).

Icon	Name	Description
	Local Attribute	A local (green) attribute means it is only used by the section in which it appears.
	Grey Attribute	A grey attribute is used for comments.
	Control Icon	A Control type defines the functionality that you can give to Attributes. Each Control type has a different functionality.

Advanced editing mode allows you to add the following:

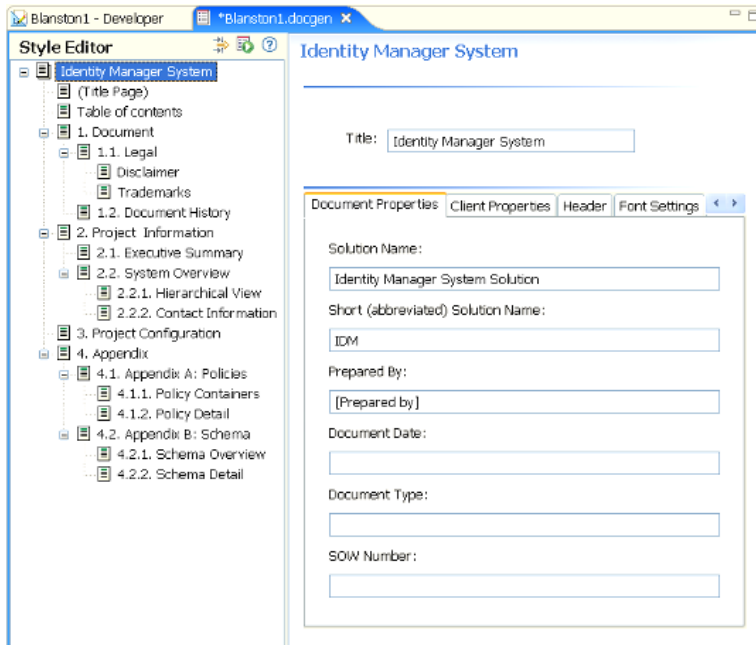
- ◆ [“Sections” on page 322](#)
- ◆ [“Viewing or Editing Properties of a Section” on page 324](#)
- ◆ [“Templates” on page 325](#)
- ◆ [“Attributes” on page 325](#)
- ◆ [“Controls and Parameters” on page 326](#)

You can have multiple sections in a document, but only one template per section. The template defines the section’s layout; however, you can use the default template for newly created sections. You can also have multiple attributes defining how the section looks, as well as multiple controls. You use parameters (such as names and values) to specify options for a Control type. A Parameter is a general name for a child item of a Control. The name of the Control denotes the type of control and what you can perform by using that type.

Sections

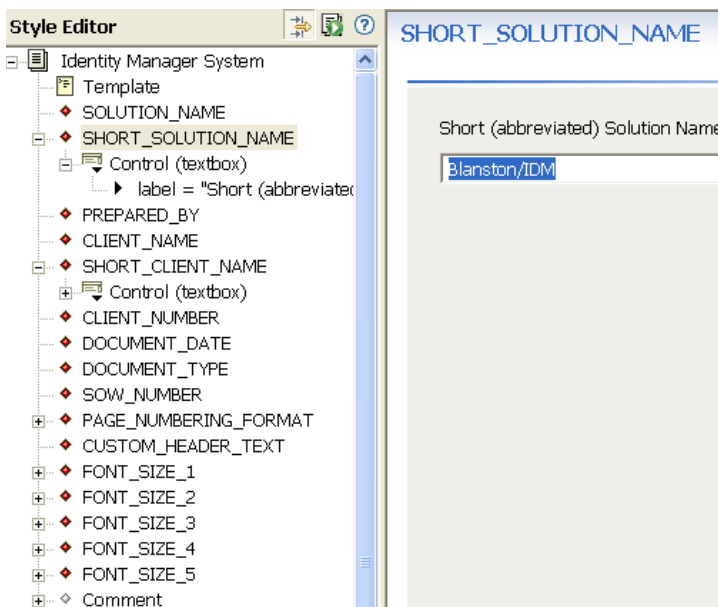
Sections are blocks of the documentation composed of attributes, parameters, templates, and controls through XSL programming. Section content includes a Title, Body, and children or subordinate information. The following example shows the Section Properties page of the Identity Manager System as seen through the simple edit mode.

Figure 16-3 Section Properties Page



The Identity Manager System section contains a Section Title (this can be changed), along with a number of tabs (attributes): *Document Properties*, *Client Properties*, *Header*, *Font Settings*, and *Other*. Each of these tabs contain fields that are editable; for instance, you can give the Section Title a different name. When you click the *Advanced Editing* icon, you see that the Identity Manager System section contains one template but several attributes, controls, and parameters underneath the heading.

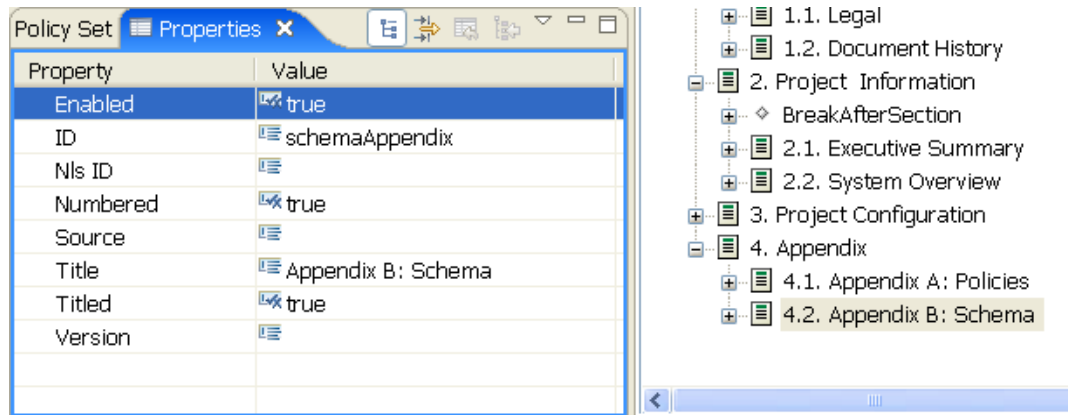
Figure 16-4 Advanced Editing Mode



Viewing or Editing Properties of a Section

If you click a section within the Style Editor and look in the Properties view (by default the lower left corner of Designer), you see the values associated with the selection. (If you do not see the Properties view by default, right-click a Section and select Show Properties View.)

Figure 16-5 The Properties View of the Appendix B: Schema



These values are edited in the Properties view, and for the Section heading the values are listed in the following table:

Table 16-3 Values of a Section

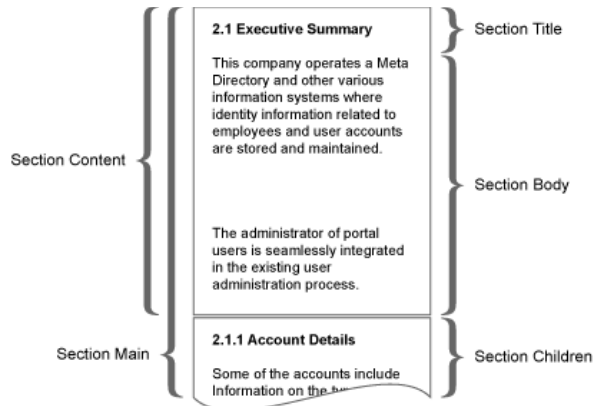
Property Name	Description
Enabled	(True/False) Indicates whether this section is enabled. You can change this setting using the section Style Editor, which is the editor shown to the right of the hierarchical view.
ID	Used for reference. Most the time, ID is left empty. However, you can specify an ID for convenience in finding this section during the transformation process.
NLS ID	Used for reference. Most the time, NLS ID is left empty.
Numbered	(True/False) Indicates whether this section should be included in the numbering and placed in the Table of Contents.
Source	Data source used to transform the template. For example, designer, style, and none.
Title	The value to be displayed as the title. You can change this setting using the section Style editor, which is the editor shown to the right of the hierarchical view.
Titled	(True/False) Indicates whether the title value should be shown in the generated document. Otherwise, it is used only in the GUI for context.
Version	The version of the section.

NOTE: Values change, depending on what you select under the Style Editor view. For example, an attribute shows different properties than a section or a template.

Templates

A template is the XML source that defines the overall layout of a generated page. For instance, the Title Page contains a template, as well as a number of other headings. The following figure illustrates the parts of the Executive Summary template. For more information about templates, see [“A Walk-through Tutorial” on page 327](#).

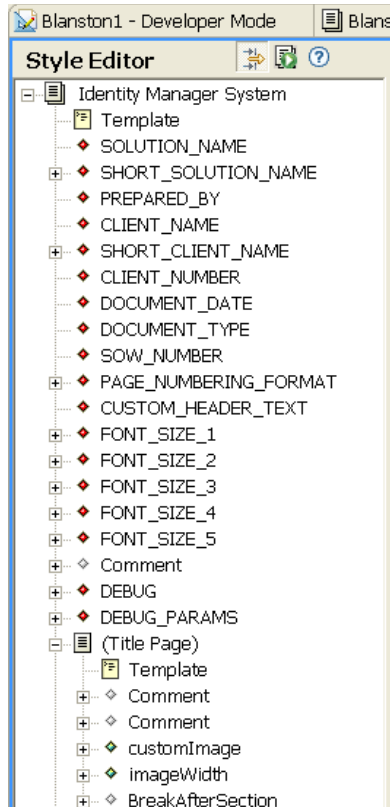
Figure 16-6 *Parts of the Executive Summary Template*



Attributes

Attributes are the child elements of a Section. For example, clicking the Advanced Editor mode while selecting the section title Identity Manager System reveals the following attributes in red (global), green (local), and grey (comment):

Figure 16-7 *Attributes*



Controls and Parameters

You can add parameters to control how things appear. In the Advanced Editing mode, the structure of the Short (abbreviated) Solution Name entry is a global attribute that contains a control and a label. For example, the control type known as textbox allows anyone to type a name that appears in the generated document. Use the Properties view to edit controls.

Designer’s supported parameters or values for controls include the following:

Table 16-4 *Supported Parameters*

Control	Parameter or Value	Description
Table	columns="3" header="date" width="30" label="show this" addrows removerows	Number of columns to show in the Control. Column header text. Column width for each column. Explains what you see in this control. Displays a button to perform this function. Displays a button to perform this function.
File	extensions=".jpg;*.gif" label="show this"	Supported extensions separated by a semicolon (*.jpg;*.jpeg;*.gif). (One file only.) The label explains what to perform with this control.

Control	Parameter or Value	Description
Select (Identity Manager System/ font Settings)	option="font 1" option="font 2" label="show this"	Parameters allow font point selection, such as option = "20pt" and option = "24pt" The label indicates what you can do with this control.
Checkbox	label="show this"	The label explains what you can perform with this control. It includes a check box.
Textbox or Textarea	label="show this"	The name gives a title to the text box or text area control. You edit these controls through the Properties view.
Comment	label="show this"	Allows you to add comments to help guide users. You edit this control through the Properties view.

16.6.2 A Walk-through Tutorial

Now that you better understand what components are necessary in order to add advanced functionality to your template, use this section to create a new section, add an attribute, and view the source.

- ♦ [“Creating a New Section in a Style” on page 327](#)
- ♦ [“Adding an Attribute to a Style” on page 327](#)
- ♦ [“Getting Documents to Recognize Your Additions” on page 328](#)
- ♦ [“Viewing the Source” on page 333](#)

Creating a New Section in a Style

To insert an additional section into the Style Editor:

- 1 Create or open an existing `.docgen` file in the Style Editor.
- 2 Click the *Enable Advanced Editing* icon. 

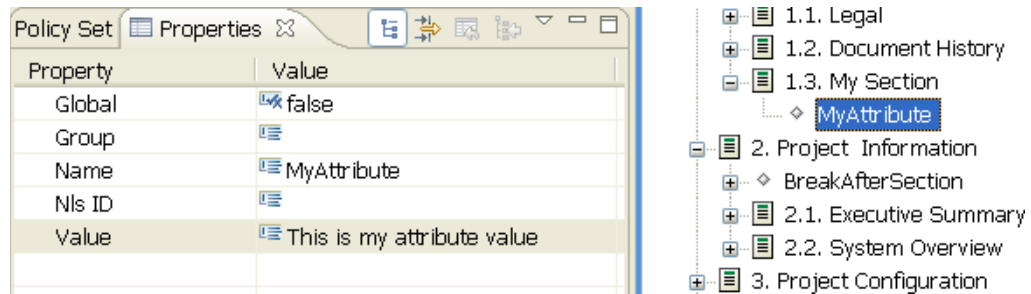
The tree view of the document outline expands to include additional objects, (such as attributes, template, and parameters).

- 3 Right-click the parent section where you want to add your new section, then select *New Child > Section*. Specify a new section name; for this example, call it “My Section.”
- 4 (Optional) Reorder the section by dragging the section object to a different location in the navigation tree. You can also copy and paste within this style or other styles.
- 5 Click the *Save* icon, then continue with the next section.

Adding an Attribute to a Style

- 1 Right-click a section under the Style Editor view. Select *New Child > Attribute*.
- 2 Specify the attribute name in the Attribute Name window. For example, MyAttribute with no spaces. Click *OK*.

- 3 Specify a value under the Value property in the Properties view. For example, *This is my attribute value*.

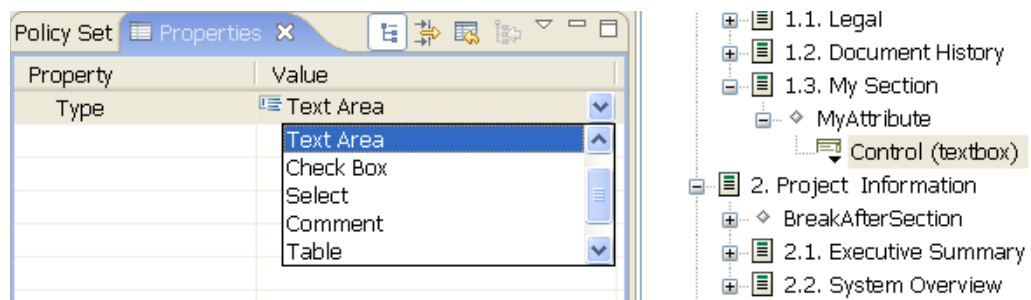


The Properties view shows the following values of Attributes (attributes are defined through controls and parameters).

- ♦ **Global:** passed down to subsections
- ♦ **Group:** used to group attributes together--these appear as part of a tab in Style Editor's simple mode
- ♦ **Name:** the attribute's name
- ♦ **NLS ID:** the attribute's NLS identification
- ♦ **Value:** the attribute's value

You can also show your attribute with another control type. The following example first creates a control, then changes the control type from check box (the default) to something else, such as a text area.

- 1 Right-click *MyAttribute* and select *New Child > Control*.
- 2 With your cursor on the control you just created, change the control type value to *Text Area* in the Properties view.
- 3 Click your section to see the changes take place.



Getting Documents to Recognize Your Additions

After you have added attributes, your generated document doesn't include information from these attributes until you do one of the following:

- ♦ Make sure your section is a leaf section (does not contain child sections).
- ♦ Create a template that uses the attribute explicitly. This is usually the preferred method because you can display the attribute values exactly the way you want by describing the layout using the template.

Method 1: Set the Section Source to “Style”

If you create a section without a template (and the template is a leaf section containing no section children), the default template generates the attribute values with the document. You do not need to do anything. You can generate a document for just that section by right-clicking the section head and selecting *Generate Documentation for This Section*. Or you can click the *Generate Document* icon at the top to generate the whole document.

NOTE: Text boxes, text areas, and tables are the only attributes that are generated through the default template (check boxes, selects, and comments are not generated).

Method 2: Add a Template for a Custom Layout

Complete the following tasks:

- ◆ [“Creating a Template” on page 329](#)
- ◆ [“Creating Another Section and Template” on page 331](#)

Creating a Template

- 1 Right-click your newly created section.

2 Select *New Child > Template*.



```
Section Template
<?xml version="1.0" encoding="UTF-8"?><xsl:stylesheet version="1.0" xmlns:fo="http://t
<xsl:import href="dgSection.xsl"/>

<xsl:param name="DOCUMENT_DATE"/>
<xsl:param name="SOLUTION_NAME"/>
<xsl:param name="SHORT_SOLUTION_NAME"/>
<xsl:param name="DOCUMENT_TYPE"/>
<xsl:param name="CLIENT_NAME"/>
<xsl:param name="SHORT_CLIENT_NAME"/>
<xsl:param name="CLIENT_NUMBER"/>
<xsl:param name="PREPARED_BY"/>
<xsl:param name="SOW_NUMBER"/>

<xsl:param name="customImage"/>
<xsl:param name="imageWidth"/>

<xsl:template name="Section.Content">
  <fo:block>
    <xsl:if test="$SOLUTION_NAME != '' or $SHORT_SOLUTION_NAME != ''">
      <fo:block space-after="({$FONT_SIZE_4})">
        Solution: <xsl:value-of select="$SOLUTION_NAME"/>
        <xsl:if test="$SHORT_SOLUTION_NAME != ''">
          (<xsl:value-of select="$SHORT_SOLUTION_NAME"/>)
        </xsl:if>
      </fo:block>
    </xsl:if>

    <xsl:if test="$PREPARED_BY != ''">
      <fo:block space-after="({$FONT_SIZE_4})">
        Prepared by: <xsl:value-of select="$PREPARED_BY"/>
      </fo:block>
    </xsl:if>

    <xsl:if test="$CLIENT_NAME != '' or $SHORT_CLIENT_NAME != ''">
      <fo:block space-after="({$FONT_SIZE_4})">
        For Client: <xsl:value-of select="$CLIENT_NAME"/>
        <xsl:if test="$SHORT_CLIENT_NAME != ''">
          (<xsl:value-of select="$SHORT_CLIENT_NAME"/>)
        </xsl:if>
      </fo:block>
    </xsl:if>

    <xsl:if test="$CLIENT_NUMBER != ''">
      <fo:block space-after="({$FONT_SIZE_4})">
        Client number: <xsl:value-of select="$CLIENT_NUMBER"/>
      </fo:block>
    </xsl:if>

    <xsl:if test="$DOCUMENT_DATE != ''">
      <fo:block space-after="({$FONT_SIZE_4})">
        Document Date: <xsl:value-of select="$DOCUMENT_DATE"/>
      </fo:block>
    </xsl:if>
  </fo:block>
</xsl:template>
```

The template has some default content that consists mostly of comments, which helps you get started on your first template. This is shown in the next task; for now, replace the comments in this template with the following XSL commands:

```
<xsl:param name="MyAttribute"/>

<xsl:template name="Section.Body">
  MyAttribute:
  <xsl:call-template name="Format.OutputTextArea">
    <xsl:with-param name="value" select="$MyAttribute"/>
  </xsl:call-template>
</xsl:template>
```

There is a `Format.OutputTextArea` call in the XSL that is a helper function included with the DocGen Core Support Templates. Because HTML code is allowed in text areas, this ensures that it's

interpreted and escaped properly. If you want to see the core XSL library calls for documentation generation, see [“Document Generator Core Support Templates” on page 525](#).

Your template should look like this:

Figure 16-8 Example Template

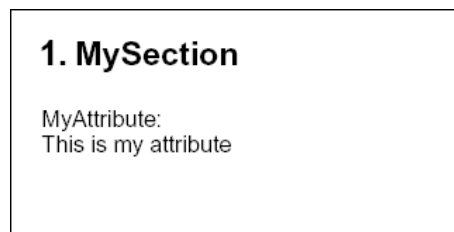
```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1:
  <xsl:import href="dgSection.xsl"/>

  <xsl:param name="MyAttribute"/>

  <xsl:template name="Section.Body">
    MyAttribute:
    <xsl:call-template name="Format.OutputTextArea">
      <xsl:with-param name="value" select="$MyAttribute"/>
    </xsl:call-template>
  </xsl:template>
</xsl:stylesheet>
```

Generating a document for this section should give you something like this:

Figure 16-9 Sample Section



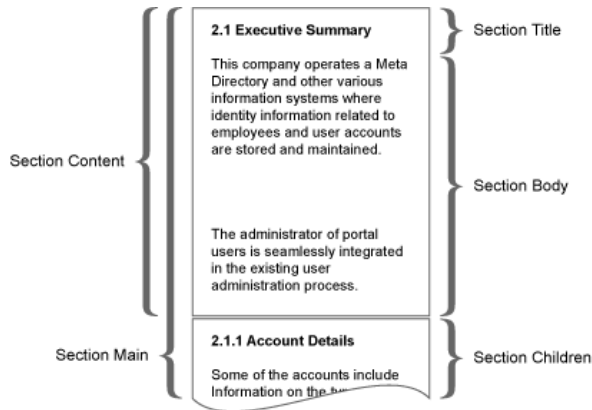
Creating Another Section and Template

- 1 Right-click your newly created section and select *New Child > Section*. In the Section Name window, name the new section Table of Contents and click *OK*.
- 2 Right-click this new section and select *New Child > Template*. Carefully read through the comments in this template. These details are important.

When you create a custom section, you are inserting some information into the document. As the comments mention, developers usually override one of the following template functions:

- ♦ Section Body (most common)
- ♦ Section Content
- ♦ Section Title

The following image illustrates which section is being defined. As a developer of the style, you write this template to overwrite the area that is of interest.



For this example, we are going to overwrite the Section.Body because we don't want to change the default behavior of the Title, nor do we want to change the way other sections are related to this one. (You can use the hierarchical view to control this with the default template anyway).

- 3 To overwrite the Section Body, we simply uncomment the sample function that is shown in the default template. This is shown below:

```

Section Template
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/
  <xsl:import href="dgSection.xsl"/>

<!-- /**
  * Table of Contents example (remove commenting below
  *
  * This template assumes you have your section source
  * to 'style' (the default value), and demonstrates h
  * render data taken from an XML source.
  *
  */ -->
<xsl:template name="Section.Body">
  <fo:block>Here is my Table of Contents:</fo:block>
  <xsl:apply-templates/>
</xsl:template>

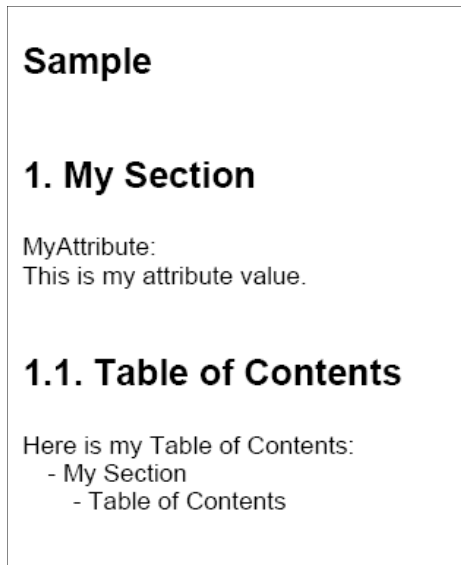
<xsl:template match="children">
  <fo:block margin-left="{count(ancestor::*)}em" - <xsl:
  <xsl:apply-templates/>
</xsl:template>
</xsl:stylesheet>

```

If you render your document at this point, you get no content in your Table of Contents (other than the surrounding text). This is because this template assumes that the style source has been specified for this Section. To specify the source:

- 1 Click the Table of Contents section.
- 2 From the Properties view, set the source to *style*.

3 Right-click *MySection* and select *Generate Documentation for This Section*.



Viewing the Source

When you generate your document, you'll notice there is an option to *Output XML Source Files*. Click the box next to this option to turn it on. You'll see `.xml` source files appear where you are generating the document. These source files are the XML data that is used in your template when you set the source (for example, to "style"). Designer 1.1 and above include the following sources:

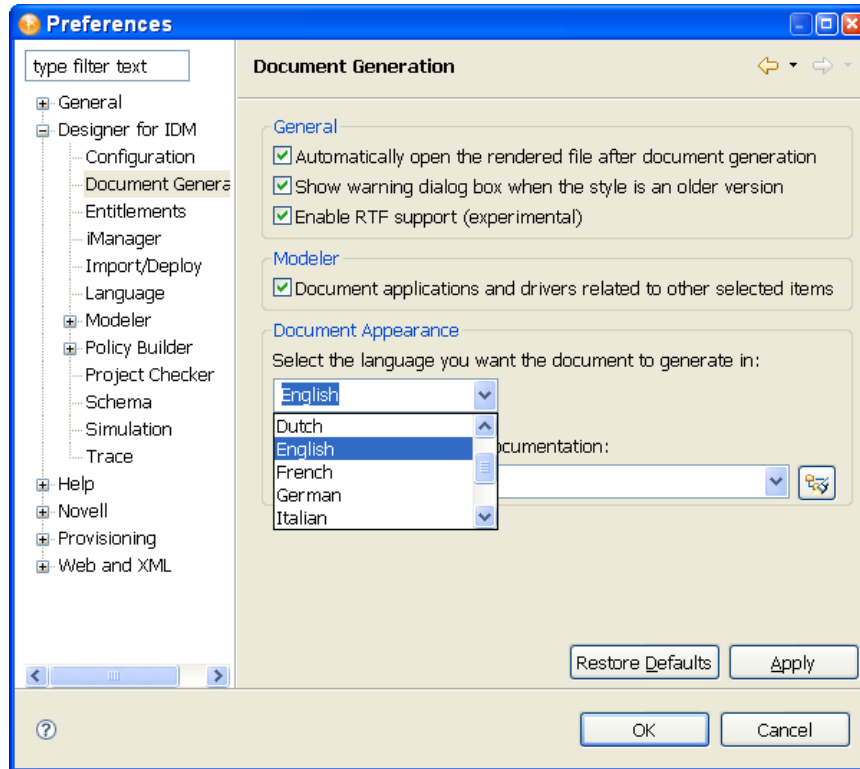
Table 16-5 Sources

Source Key	Description
none	An empty source, used when no source is specified or when "none" is specified.
style	The XML source of the style, used to build things like the Table of Contents.
designer	A source that has been defined by an extension point for the Designer model. This contains all information about the configuration of your Designer project.

16.6.3 Selecting a Language for Generated Documents

You can also select the language you want to print the document in.

- 1 Click *Window > Preference > Designer for IDM > Document Generation*. Under the *Document Language* heading, select the language you want to use for document generation.



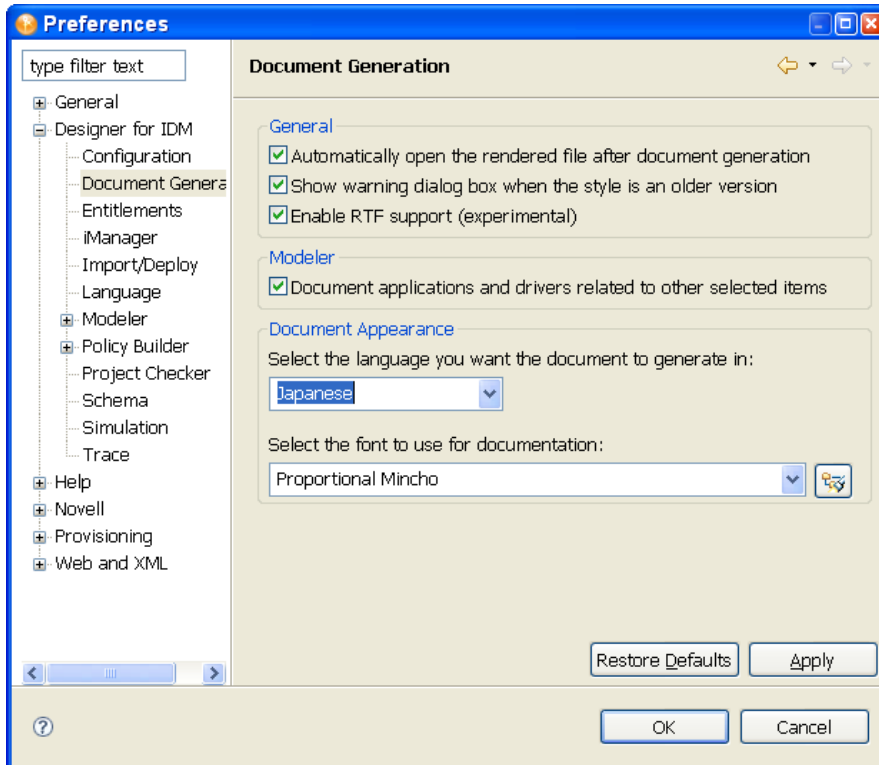
Present languages include:

- ♦ Chinese simplified
 - ♦ Chinese traditional
 - ♦ Dutch
 - ♦ English (default)
 - ♦ French
 - ♦ German
 - ♦ Italian
 - ♦ Japanese
 - ♦ Portuguese Brazil
 - ♦ Spanish
- 2 Once you select a language, click *Apply*.
 - 3 Click *OK* to close the Preference page.

16.6.4 Double-Byte Font Support

Designer now has double-byte font support for the Document Generation feature. If you select a language that uses double-byte fonts, such as Chinese Simplified, Chinese Traditional, and Japanese, Designer automatically installs the font `Proportional Mincho`.

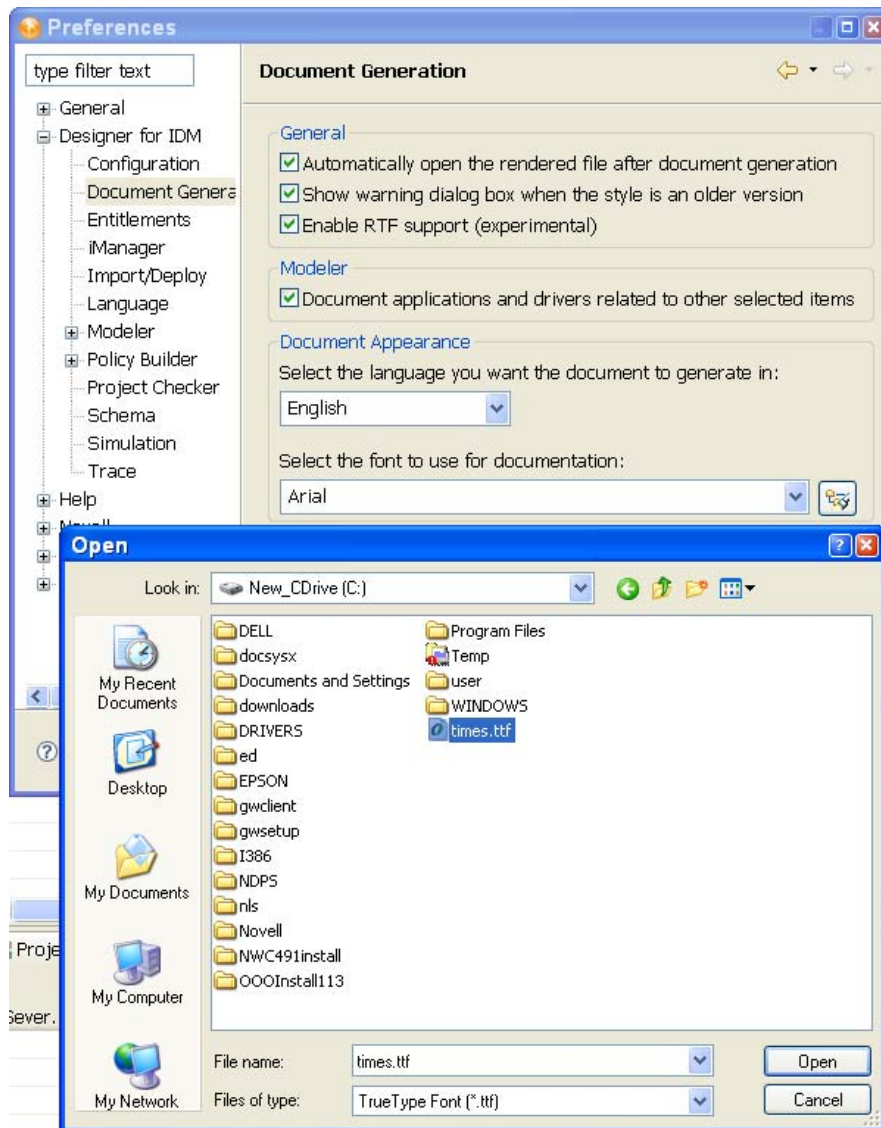
Figure 16-10 *Selecting a Double-Byte Font Language*



For English and other languages, the default font is Arial.

To add a font for your specific language:

- 1 Click *Window > Preference > Designer for IDM > Document Generation*. Under *Font Settings*, select the font you want to use.



To change the font on a Windows workstation, you must first copy the font file from the `C:\Windows\Fonts` directory to another directory. You can then use the *Browse* button to select the font.

To change the font on a Linux workstation, browse to the `usr/share/fonts/truetype` directory, or to another directory containing the fonts you want.

- 2 Click the *Browse* button to bring up the Open window, then change to the directory where you placed the font, then click *Open*.

You can also type the directory and font file name into *Font Settings* field, or use the pull-down menu to select a font that you have previously selected.

- 3 Click *Apply*, then click *OK*.

Using the above steps will change the font globally in the generated document, as well as add double-byte font support for such selections.

16.6.5 Saving a Style File in the Styles Template Library

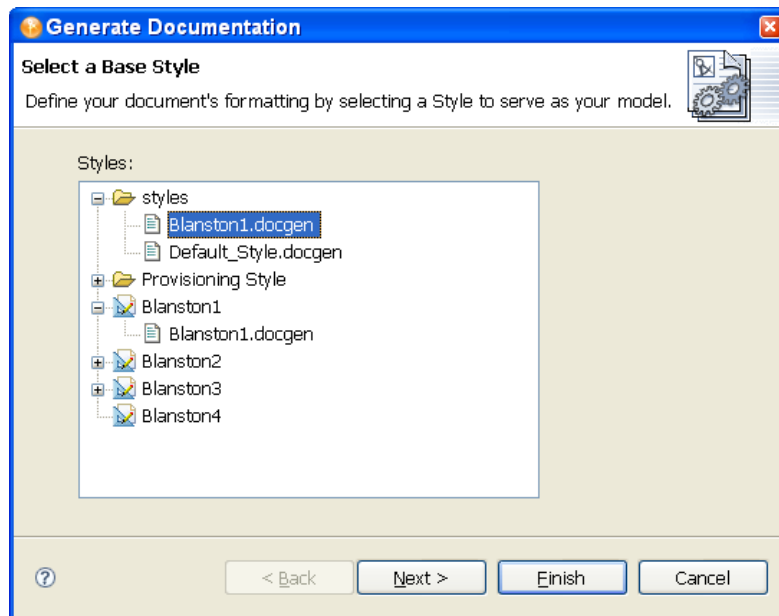
Designer provides one or more default styles that you can use to generate documentation for your project. The `Default_Style.docgen` file is stored in the `Novell\designer\eclipse\plugins\com.novell.designer.idm.docgen.editor_2.0\content\styles` folder and is displayed as an option when you choose a new style.

You can add your own style files to the `...\content\styles` folder so that you can easily find them when choosing a basis for a new style.

- 1 Locate the `.docgen` file you want to add.

By default, the file is in a directory inside the project where you saved it. For example, the `Blanston2.docgen` file was created in the `workspace\Blanston2\documentation` directory.

- 2 Copy the `.docgen` file into the Styles Library (`Novell\designer\eclipse\plugins\com.novell.designer.idm.docgen.editor_2.0\content\styles`). The `.docgen` file appears in the `styles` folder the next time you open Designer and generate a document.



16.7 Troubleshooting Issues When Generating Documents

For information on troubleshooting issues that you encounter when generating documents, see [Section 24.5, "Issues When Generating Documents," on page 509](#).

Identity Manager allows you to synchronize data between connected systems. Entitlements allow you to set up criteria for a person or group that, once met, initiate an event to grant or revoke access to business resources within the connected system. This gives you one more level of control and automation for granting and revoking resources.

Use Designer to create entitlements and deploy them into existing Identity Manager drivers. Designer allows you to create entitlements through the Entitlement Wizard, which gives you a graphical interface where you can create the entitlement and steps you through the process. Because of its graphical interface, we recommend using Designer for creating and editing entitlements.

There are four aspects to making entitlements work effectively: design, creation, editing, and management.

- ◆ [Section 17.1, “Designing Entitlements,” on page 339](#)
- ◆ [Section 17.2, “Creating Entitlements through the Entitlement Wizard,” on page 342](#)
- ◆ [Section 17.3, “Editing and Viewing Entitlements,” on page 354](#)
- ◆ [Section 17.4, “Managing Entitlements,” on page 359](#)

17.1 Designing Entitlements

You must know beforehand what you want to accomplish with entitlements. Entitlements work from the functionality you build into Identity Manager drivers through policies. These driver policies implement rules and process the events between the Identity Vault and the connected system. If the policies in the Identity Manager driver do not specify what you want to do, entitlements cannot work. For example, if you don't specify the action section of the Check User Modify for Group Membership rule in the Command policy, attempts to grant or revoke a group membership entitlement are ignored.

When you know what you want to accomplish with Identity Manager, you can correctly design granting and revoking capabilities for any connected system resources. The following four-step procedure can help you plan to create and use entitlements:

1. Know what you want to accomplish in your business situation. You can design and implement many business solutions through Identity Manager, but you need to know what you want to do before implementing something that isn't defined. Make a numbered list of what you want to do.
2. Define an entitlement that represents one item from your numbered list. You can create valueless and valued entitlements. Valued entitlements can get their values from an external query, they can be administrator-defined, or they can be free form. There are examples in [Section 17.2, “Creating Entitlements through the Entitlement Wizard,” on page 342](#).
3. Add policies to the Identity Manager Driver to implement the designed entitlement. To create a policy for an Identity Manager driver, you need to be conversant with XSLT or DirXML[®] script, in the way the connected system handles and receives information, and with the way Novell[®] eDirectory[™] stores information. Unless you are a good DirXML programmer, this is a job for consultants.

4. Set up a managing agent to grant or revoke the entitlement. If you want an automated process, use Role-Based Entitlements; if you want a manual process, use the User Application’s workflow-based provisioning feature. See “[Identity Manager Installation Programs and Services](#)” in the *Identity Manager 3.5.1 Installation Guide*.
 - ◆ [Section 17.1.1, “Terminology,” on page 340](#)
 - ◆ [Section 17.1.2, “Entitlement Prerequisites,” on page 340](#)
 - ◆ [Section 17.1.3, “Identity Manager Drivers with Preconfigurations that Support Entitlements,” on page 341](#)
 - ◆ [Section 17.1.4, “Enabling Entitlements On Identity Manager Drivers,” on page 341](#)

17.1.1 Terminology

Following are some terms that are used throughout this section.

Table 17-1 Terminology

Terms	Explanation
Entitlement	An Identity Vault object that represents a business resource in a connected system.
Entitlements Service Driver	Grants and revokes entitlements. For Role-Based entitlements, the agent is the Entitlements Service driver that must be initiated for entitlements to work.
Grant or Revoke	Granting or revoking an entitlement is controlled by Global Configuration Variables (GCVs) on an Identity Manager driver.
Entitlement Consumer	Anything that uses entitlement-related information. Entitlement consumers include iManager, the User Application, and Identity Manager policies.

17.1.2 Entitlement Prerequisites

- ❑ eDirectory 8.7.3.3 or later with the latest Support Pack (SP8 or later)
- ❑ Identity Manager 2 or 3
- ❑ An Entitlements Service driver

You must have an Entitlements Service driver in each driver set where you want to use entitlements. This requires a very simple, two-step setup for each driver set. To do this, see “[Creating an Entitlements Service Driver Object](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

- ❑ A driver configuration that supports entitlements

Before you can use entitlements with a connected system, do one of the following:

- ◆ Import the Identity Manager driver configuration for the driver and specify that the driver has entitlements enabled.
- ◆ Enable your driver to support entitlements. To do this:
 - a. Create entitlements using Designer.

- b. Add the DirXML-EntitlementRef attribute to your driver filter as described in [Section 17.1.4, “Enabling Entitlements On Identity Manager Drivers,” on page 341.](#)
- c. Write policies to implement the entitlements you create in Step 1 under [Section 17.1, “Designing Entitlements,” on page 339.](#)

17.1.3 Identity Manager Drivers with Preconfigurations that Support Entitlements

Identity Manager comes with several preconfigured drivers that already contain entitlements, policies to implement the entitlements, and the driver enabled to listen for entitlement activities. You must enable entitlements when you initially install preconfigured drivers in order to add the preconfigured elements to the driver’s structure. The following drivers have preconfigurations that support entitlements:

- ◆ Driver for Active Directory: Grant and revoke accounts, group membership, Exchange Mailbox
- ◆ Driver for Exchange 5.5: Grant and revoke mailbox and group membership
- ◆ Driver for GroupWise: Grant and revoke accounts, grant and revoke members of distribution lists
- ◆ Driver for LDAP: Grant and revoke user accounts
- ◆ Driver for Linux and UNIX: Grant and revoke accounts
- ◆ Driver for Lotus Notes: Grant and revoke user accounts and group memberships
- ◆ Driver for NT Domain: Grant and revoke user accounts and group membership
- ◆ Driver for RACF: Grant and revoke group accounts and group memberships

These preconfigured drivers fulfill the first three of the four steps outlined in [Section 17.1, “Designing Entitlements,” on page 339.](#) They contain example entitlements and policies that you can use as is if they meet your needs; you can also change them to better meet your needs, or you can use them as examples and make your own entitlements through Designer.

17.1.4 Enabling Entitlements On Identity Manager Drivers

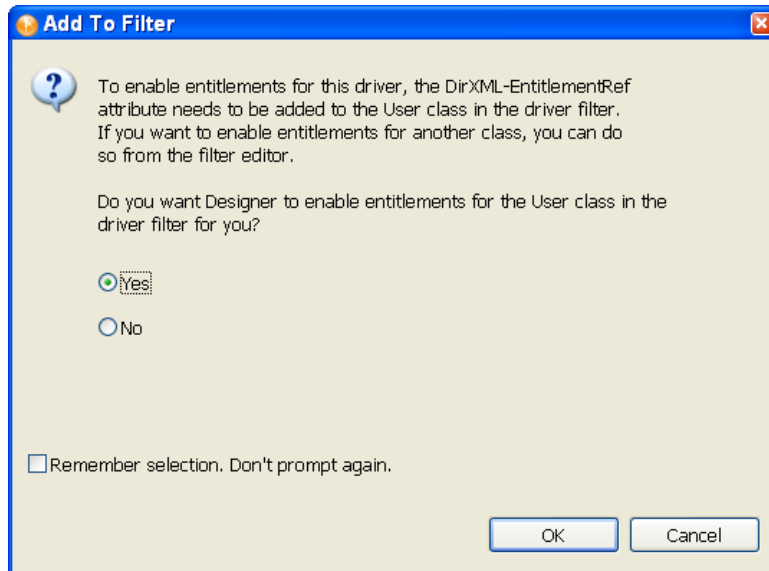
Before you can use entitlements, you must first ensure that your driver has entitlements enabled. You can do this through the Entitlements Wizard as you finish creating entitlements; this applies to both preconfigured and non-preconfigured drivers.

However, if you want to use the preconfigured driver’s entitlements and the infrastructure that supports them, you must enable entitlements when you initially create a driver in Designer or iManager; the preconfigured policies and rules that support the preconfigured entitlements cannot be added later without re-creating the driver. If you import a driver that has entitlements enabled into Designer from an Identity Vault, the imported driver also has entitlements enabled. If you deploy a driver that has entitlements enabled into an Identity Vault, the deployed driver also has entitlements enabled.

You can see if your preconfigured drivers have entitlements enabled by clicking the Outline view, then clicking the Subscriber channel of your selected driver. If entitlements are enabled, you should see the preconfigured entitlements appear under the Subscriber Channel. If entitlements do not appear under the Subscriber Channel in the Outline view, entitlements were not enabled when the driver was initially installed.

However, you can still use entitlements on preconfigured Identity Manager drivers that do not have entitlement preconfigurations enabled. To do this, run the Entitlement Wizard. The last page in the Entitlement Wizard asks if you want to add the DirXML-EntitlementRef attribute to the driver filter, with *Yes* selected. Click *OK*. However, because the policies and rules are not in place on the driver, you won't be able to use their preconfigured entitlements without adding those supporting policies and rules yourself.

Figure 17-1 Enabling Entitlements



You can also use entitlements on Identity Manager drivers that do not contain entitlement preconfigurations. To enable your driver to support entitlements, add the DirXML-EntitlementRef attribute to your driver filter. Run the Entitlement Wizard as described above to add the DirXML-EntitlementRef attribute to the driver filter.

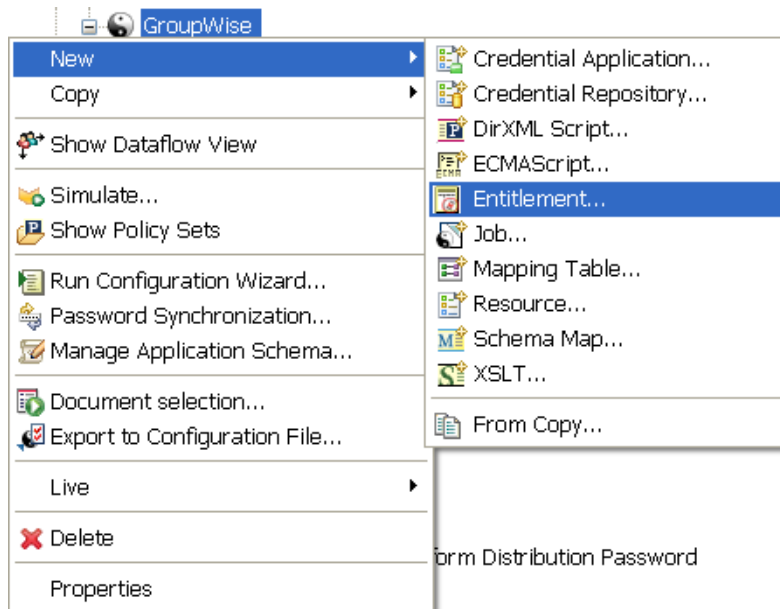
17.2 Creating Entitlements through the Entitlement Wizard

Designer comes with an Entitlement Wizard. This wizard steps you through the creation of entitlements by asking a series of questions about how the entitlement will be used in the enterprise. Use one of the following methods to access the Entitlement Wizard:

From the Outline View

To access the Entitlement Wizard from the Outline view, right-click a Driver object, then click *New > Entitlement*.

Figure 17-2 Adding an Entitlement through the Outline View



From the Modeler View

To access the Entitlement Wizard from the Modeler view, right-click the driver icon, then click *New > Entitlement*.

There are two types of entitlements that you can create: valueless and valued. The type you use depends on whether you need to pass additional information to the policies.

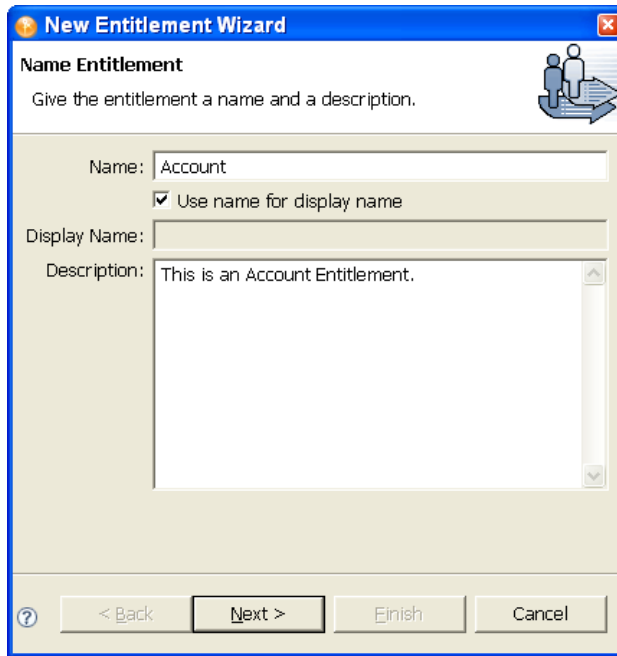
- ◆ [Section 17.2.1, “Valueless Entitlements,” on page 343](#)
- ◆ [Section 17.2.2, “Valued Entitlement that Queries an External Application,” on page 345](#)
- ◆ [Section 17.2.3, “Administrator-Defined Entitlements with Lists,” on page 351](#)
- ◆ [Section 17.2.4, “Administrator-Defined Entitlements without Lists,” on page 353](#)

17.2.1 Valueless Entitlements

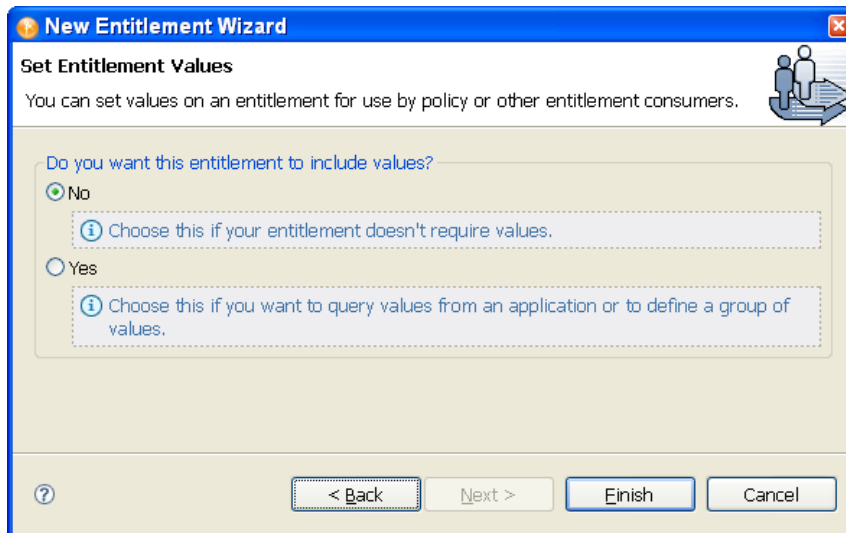
A valueless entitlement has no values to go with it. An example is the Account Entitlement for Active Directory, which is used to turn on account capabilities. You use valueless entitlements if you don't need to pass any extra information to driver policies.

To create a valueless entitlement:

- 1 Right-click the driver icon in the Outline view or in the Modeler view, then click *New > Entitlement*.

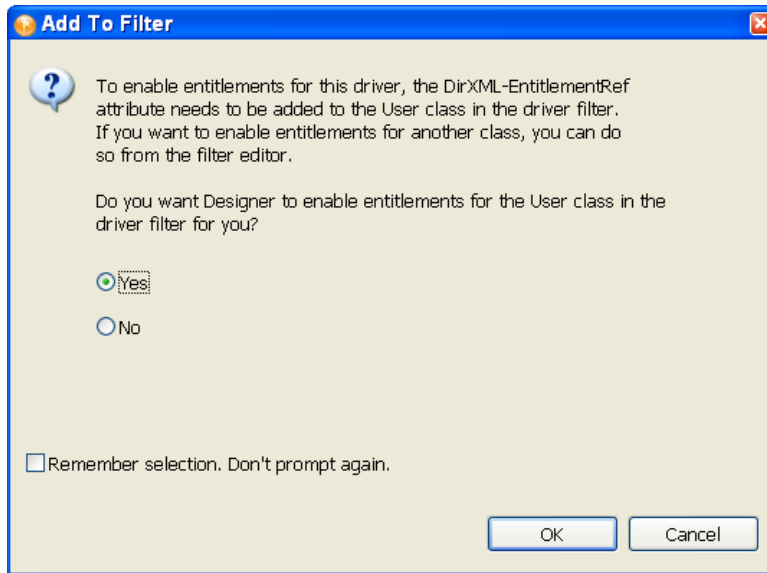


- 2 Type the name and description information. For this example, the entitlement is named Account, with a description of “This is an Account Entitlement.” Click *Next*.



- 3 Because this first example is valueless, select *No* to *Do you want this entitlement to include values?*
- 4 Click *Finish*.

- 5 In the Add To Filter dialog box, answer *Yes* if you want the driver to listen for this entitlement. This enables entitlements for the driver.



The DirXML-EntitlementRef attribute allows the driver filter to listen for entitlement activities. This is necessary in order to use the entitlements you are creating.

If you don't want to see the Add To Filter window on every entitlement you are creating for any driver in Designer, select *Remember Selection - Don't Prompt Again*, then click *OK*. However, after the attribute is added to this driver filter, you won't see the Add To Filter window again.

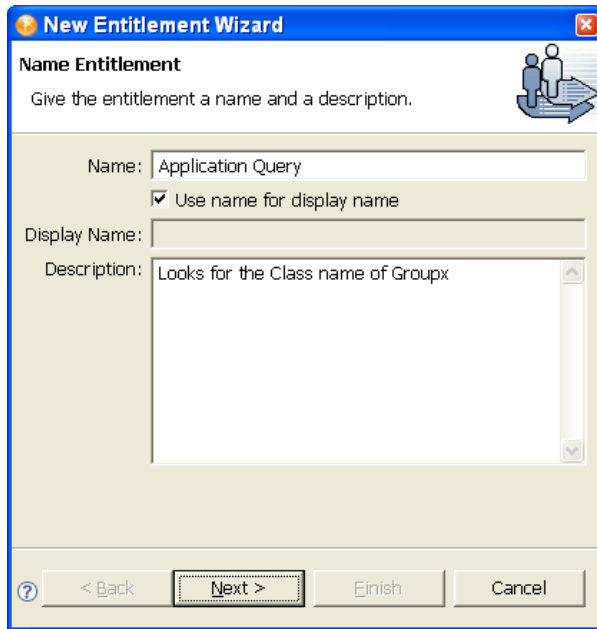
17.2.2 Valued Entitlement that Queries an External Application

Values are a way of passing data that you might need to use in policies. Valued entitlements can get their values from an external query; they can be administrator-defined, or they can be free-form. Below are four examples of the entitlements that you can create.

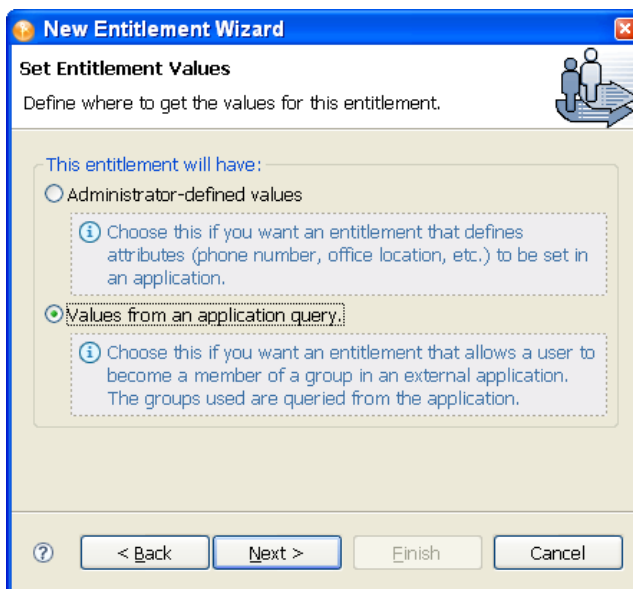
The second example is an application query entitlement. Use this entitlement when you need values from a connected system in order to implement entitlements in that system.

- 1 Right-click the driver icon in the Outline view or in the Modeler view, then click *New > Entitlement*.

- 2 Give the entitlement a name. This example uses Application Query, with the *Use Name for Display Name* option selected. In the Description box, type Looks for the Class name of Groupx, then click *Next*.

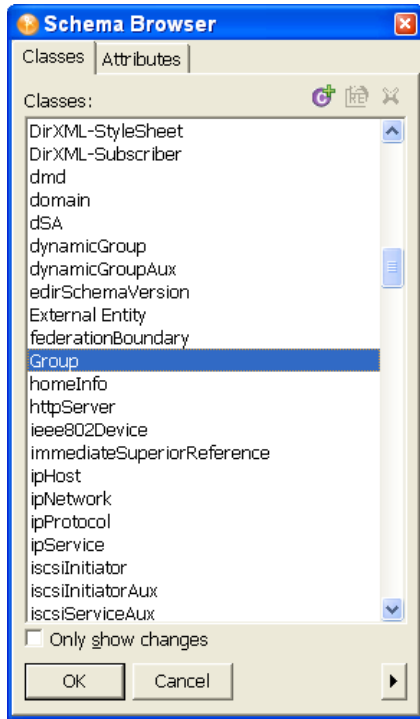


- 3 On the Set Entitlements Values page, select *Yes* so you can query values from an application or define a group of values, then click *Next*.
- 4 The next Set Entitlements Values page allows you to define where you get the values for this entitlement. Valued entitlements can get their values from an external query, or they can be administrator-defined. For this example, select the *Values from Application Query* option, then click *Next*.



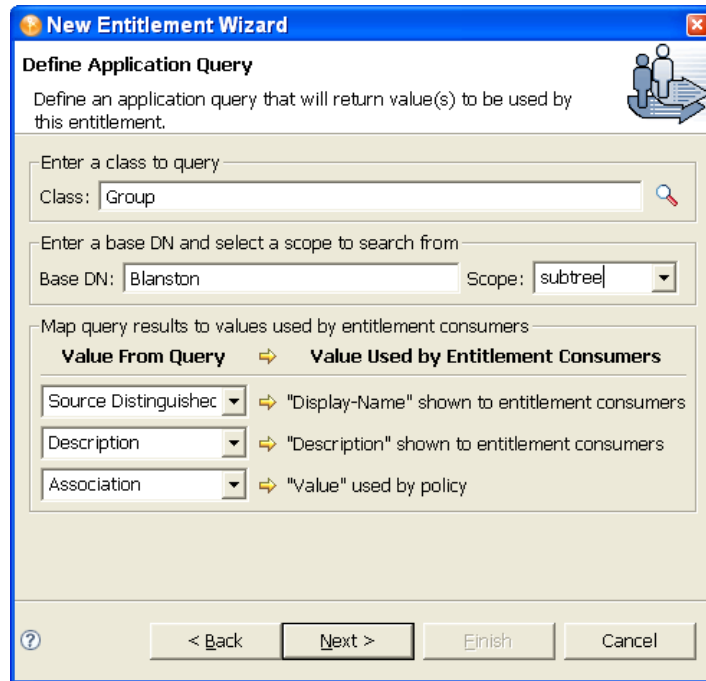
The Define Application Query window combines two steps: defining the query and mapping the query results. To define the query, fill in the Class, the Base DN (Distinguished Name), and the Scope (subtree, entry, or subordinates). Then map the query results to values that are used by the entitlement consumers (either iManager for Role-Based Entitlement policies, or the User Application for workflow entitlements).

- 5 To fill in the Class Query, click the *Schema Browser* button on the right side of the Class entry.



- 6 The Schema Browser shows you the Classes in the eDirectory namespace that are available to choose from (the Attributes tab shows you which attributes are part of the Class). If you know the name of the Class type you want to query, click to select a selection in the Classes tab, then

start typing the Class name. The browser jumps to the alphabetical order of what you type. Select the Class name, then click *OK*.



- 7 Type the base distinguished name (DN) and the scope. For this example, select the Class Group, at the Base Distinguished Name of Blanston, with the Scope of subtree (choices are subtree, entry, and subordinates).

Our example maps the query results from the connected system to certain values that entitlement consumers can use. At present, the consumers are iManager managing Role-Based Entitlement policies and the User Application managing workflow-based provisioning entitlements. The Value From Query information pre-populates the consumer's user interface with the following:

- ♦ **Display Name:** Defines the attribute that displays in the list of values. The example selects Source Distinguished Name for the display name. Click the drop-down button on the *Display Name shown to entitlement consumers list* to see a list of Attributes associated with the class you selected through the Schema Browser. The list includes both the Attributes and the Inherited Attributes for the selected class.
- ♦ **Description:** Defines the attribute that displays as a description for that value. For the description, select *Description* from the Value drop-down list to map the query results from the connected system to the entitlement.
- ♦ **Value:** Defines the attribute or token that is the actual value. The *Value* entry is not seen in the entitlement consumer, but it is the value that is assigned when the entitlement is granted or revoked. In this case, choose *Association*.

If you do not use the *Schema Browser* button when selecting the class, you see only two selections in the Value From Query lists: *Association* and *Source Distinguished Name*. If these attributes suit your needs, use them. You can also type the attribute name into the text field. However, if you want to select the attributes from the lists, use the *Schema Browser* button when selecting a class for the query. You see the attributes and inherited attributes for the selected class.

- 8 When the values are filled out, select *Next*.
- 9 In the Assign Multiple Values window, select *Yes* if you want the entitlements to be granted more than once and with different values. If you select *No*, the entitlement can only be granted once. For our example, click *Yes*, then click *Next*.

It makes sense to assign group entitlements with multiple values, but it would not make sense to assign an account entitlement more than once.

- 10 You are asked if this entitlement is intended to be used by Role-Based Entitlement policies through iManager. If you want this entitlement to be granted or revoked automatically, select *Yes* to the Role-Based Entitlements question, click *Next*, then continue with Step 11.

Or

If you want the granting or revoking of this entitlement to be a manual process (approved by someone), select *No* to use the User Application, then skip to Step 12.

We recommend that you have only one agent control an entitlement. If multiple agents are in control, you have the following consequences:

- ♦ Whoever comes last controls the entitlement results
- ♦ Results are unpredictable
- ♦ Using both agents to control an entitlement is not supported by Novell

- 11 (Conditional) If you select *Yes* to the Role-Based Entitlements question, you are asked if you want to use the Role-Based Entitlements priority to resolve any conflicts that might happen when this entitlement is assigned more than once with different values. You can resolve the conflict by either using Role-Based Entitlements priority, or by merging the values.

Merging the values merges the entitlements of all involved Role-Based Entitlement policies, so if one policy revokes an entitlement but another policy grants an entitlement, the entitlement is eventually granted. Solving conflicts by priority works if you need to ensure that only one policy is applied to this entitlement at any time. This example uses priority.

- 12 Click *Finish*.

For this example, the query values look for the Source Distinguished Name attribute of the Class name of Group, starting from the Base DN (Blanston) and checking through the subtree from that beginning point. The values that come back from the query are similar to the following:

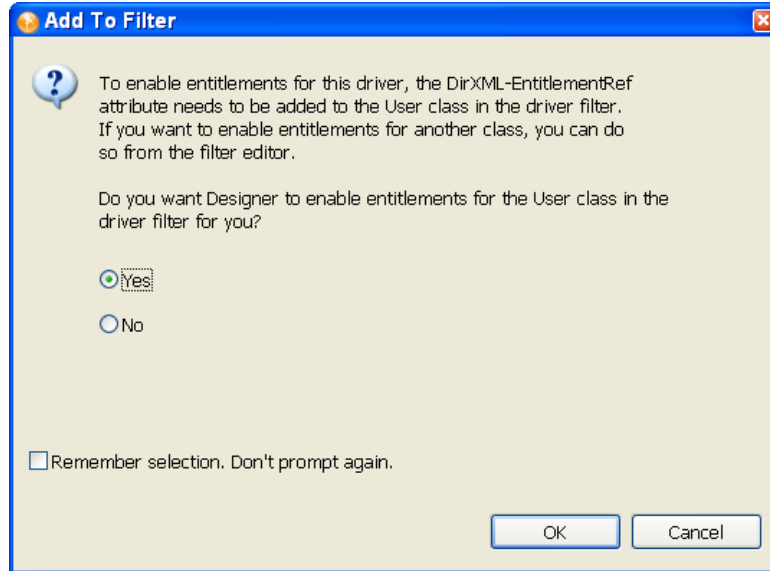
```
<instance class-name="Group" src-dn="o=Blanston,cn=group1">
  <association>o=Blanston,cn=group1</association>
  <attr attr-name="Description"> the description for group1</attr>
</instance>
<instance class-name="Group" src-dn="o=Blanston,cn=group2">
  <association>o=Blanston,cn=group2</association>
  <attr attr-name="Description"> the description for group2</attr>
</instance>
<instance class-name="Group" src-dn="o=Blanston,cn=group3">
  <association>o=Blanston, cn=group3</association>
  <attr attr-name="Description"> the description for group3</attr>
</instance>
<!-- ... -->
```

The information received from the query fills in the various fields. For instance, the `<display-name>` field receives `o=Blanston,cn=group1`. The `<description>` field receives the description for Group1, and the `<ent-value>` field receives

o=Blanston,cn=group1. Because more than one group exists and meets the query criteria, this information is also collected and shown as other instances of the query.

The association format value is unique for every external system, so the format and syntax are different for each external system queried.

- 13 At the Add To Filter window, click *Yes* if you want the driver to listen for this entitlement. This enables entitlements for the driver.



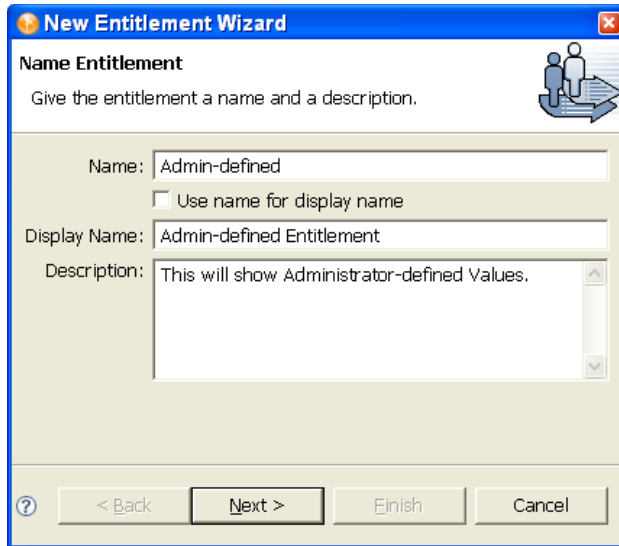
The DirXML-EntitlementRef attribute allows the driver filter to listen for entitlement activities. This is necessary in order to use the entitlements you are creating.

If you don't want to see the Add To Filter window on entitlements you are creating for any driver in Designer, select *Remember Selection - Don't Prompt Again*, then click *OK*. However, after the attribute is added to this driver filter, you won't see the Add To Filter window again.

17.2.3 Administrator-Defined Entitlements with Lists

The example in the following procedure is an administrator-defined entitlement that allows you to select a listed entry. This type of entitlement is best used through Workflow entitlements rather than Role-Based Entitlements.

- 1 Right-click the driver icon in the Outline view or the Modeler view, then click *New > Entitlement*.



In this example, the entitlement name is Admin-defined, but the defined display name is Admin-defined Entitlement. You need to define a display name only if you want the display name to be different from the name you called the entitlement; otherwise, you can just use the entitlement name as the display name. In this example, the *Description* field is defined as *This will show Administrator-defined Values*.

- 2 Click *Next*.
- 3 In the Set Entitlement Values window, select *Yes* to the question “Do you want this entitlement to include values?” Click *Next*.
- 4 In the next Set Entitlement Values window, select *Administrator Defined Values*, then click *Next*.

- 5 In the Define Values window, type the values you want to add to the Entitlement Value entry, click *Add* to add the value to the Defined List pane, then click *Next*.

New Entitlement Wizard

Define Values

Entitlement values can be taken from a defined list, or they can be entered by entitlement consumers.

Do you want to define a list of values?

Yes

No

Choose this if you want to define a list of values for the entitlement consumers to choose from.

Choose this if you want entitlement consumers to define their own values.

Entitlement Value:

Building D

Defined List:

Building A
Building B
Building C

In this example, the values are corporate buildings: Building A through Building D. Through an entitlement client, such as an iManager Role-Based Entitlement task or through the user application, users or defined-task managers can specify the building information, which is then included in an external application, such as Novell eDirectory.

- 6 In the Assign Multiple Values window, select *Yes* if you want the entitlements to be granted more than once and with different values. If you select *No*, the entitlement can only be granted once. For the example, click *No*, then click *Next*.

It makes sense to assign group entitlements with multiple values, but it does not make sense to assign building letters more than once.

- 7 You are asked if this entitlement is intended to be used by Role-Based Entitlement policies through iManager. If you want this entitlement to be granted or revoked automatically, select *Yes* to the Role-Based Entitlements question, click *Next*, then continue with Step 8.

Or

If you want the granting or revoking of this entitlement to be a manual process (approved by someone), select *No* to use the User Application, then skip to Step 9.

We recommend that you have only one agent control an entitlement. If multiple agents are in control, you have the following consequences:

- ◆ Whoever comes last controls the entitlement results
- ◆ Results are unpredictable

- ♦ Using both agents to control an entitlement is not supported by Novell

8 (Conditional) If you select *Yes* to the Role-Based Entitlements question, you are asked if you want to use the Role-Based Entitlements priority to resolve any conflicts that might happen when this entitlement is assigned by different Role-Based Entitlement Policies with different values. You can resolve the conflict by either using the Role-Based Entitlements priority, or by merging the values. This example merges the values.

Merging the values merges the entitlements of all involved Role-Based Entitlement policies, so if one policy revokes an entitlement but another policy grants an entitlement, the entitlement is eventually granted. Solving conflicts by priority works if you need to ensure that only one policy is applied to this entitlement at any time.

9 Click *Finish*.

10 If you see the Add To Filter window, answer *Yes* if you want the driver to listen for this entitlement. This enables entitlements for the driver. The DirXML-EntitlementRef attribute allows the driver filter to listen for entitlement activities, which is necessary in order to use the entitlements you are creating.

Or

If you don't want to see the Add To Filter window on entitlements you are creating for any driver in Designer, select *Remember Selection - Don't Prompt Again*, then click *OK*. However, after the attribute is added to this driver filter, you won't see the Add To Filter window again.

17.2.4 Administrator-Defined Entitlements without Lists

The example in the following procedure is an administrator-defined entitlement that forces the administrator to type a value. You can use this kind of entitlement if you do not have all of the information at the initial setup, so you cannot create a task list.

1 Right-click the driver icon in the Outline view or the Modeler view, then click *New > Entitlement*.



In this example, the entitlement name is *Admin-defined (no lists)*, and it uses the entitlement name as the displayed name because the *Use Name For Display Name* option is selected.

- 2 Click *Next*.
- 3 Select *Yes* on the Set Entitlement Values page, then click *Next*.
- 4 Select *Administrator Defined Values* on the second Set Entitlement Values page, then click *Next*.
- 5 Select *No* to the question “Do you want to define a list of values?” on the Define Values page, then click *Next*.

This displays a text field, where the administrator or users must type information. Using this option can be risky; if the information is wrong or is misspelled, the value is incorrect and the action in the entitlement fails.

- 6 Select *No* to the question “Allow this entitlement to be assigned multiple times with different values?” on the Assign Multiple Values page, then click *Next*.
- 7 You are asked if this entitlement is intended to be used by Role-Based Entitlement policies through iManager. If you want this entitlement to be granted or revoked automatically, select *Yes* to the Role-Based Entitlements question, click *Next*, then continue with Step 8.

Or

If you want the granting or revoking of this entitlement to be a manual process (approved by someone), select *No* to use the User Application, then skip to Step 9.

We recommend that you have only one agent control an entitlement. If multiple agents are in control, you have the following consequences:

- ♦ Whoever comes last controls the entitlement results
- ♦ Results are unpredictable
- ♦ Using both agents to control an entitlement is not supported by Novell

- 8 (Conditional) If you select *Yes* to the Role-Based Entitlements question, you are asked if you want to use the Role-Based Entitlements priority to resolve any conflicts that might happen when this entitlement is assigned by different Role-Based Entitlement Policies with different values. You can resolve the conflict by either using the Role-Based Entitlements priority, or by merging the values. This example uses priority.

Merging the values merges the entitlements of all involved Role-Based Entitlement policies, so if one policy revokes an entitlement but another policy grants an entitlement, the entitlement is eventually granted. Solving conflicts by priority works if you need to ensure that only one policy is applied to this entitlement at any time.

- 9 Click *Finish*.
- 10 If you see the Add To Filter window, answer *Yes* if you want the driver to listen for this entitlement. This enables entitlements for the driver. The DirXML-EntitlementRef attribute allows the driver filter to listen for entitlement activities, which is necessary in order to use the entitlements you are creating.

Or

If you don't want to see the Add To Filter window on entitlements you are creating for any driver in Designer, select *Remember Selection - Don't Prompt Again*, then click *OK*. However, after the attribute is added to this driver filter, you won't see the Add To Filter window again.

17.3 Editing and Viewing Entitlements

Designer comes with an Entitlement Wizard. This wizard steps you through the creation of entitlements by asking a series of questions about how the entitlement will be used in the enterprise.

However, before beginning this procedure, you should first see [Section 17.1, “Designing Entitlements,”](#) on page 339. For information on creating entitlements, see [Section 17.2, “Creating Entitlements through the Entitlement Wizard,”](#) on page 342.

- ◆ [Section 17.3.1, “Entitlement XML Source and XML Tree Views,”](#) on page 356
- ◆ [Section 17.3.2, “Using the Novell Entitlement DTD,”](#) on page 359

After you have created entitlements, you might need to edit them. You can also use the Edit mode to see the entitlements in their XML source code.

To edit an entitlement:

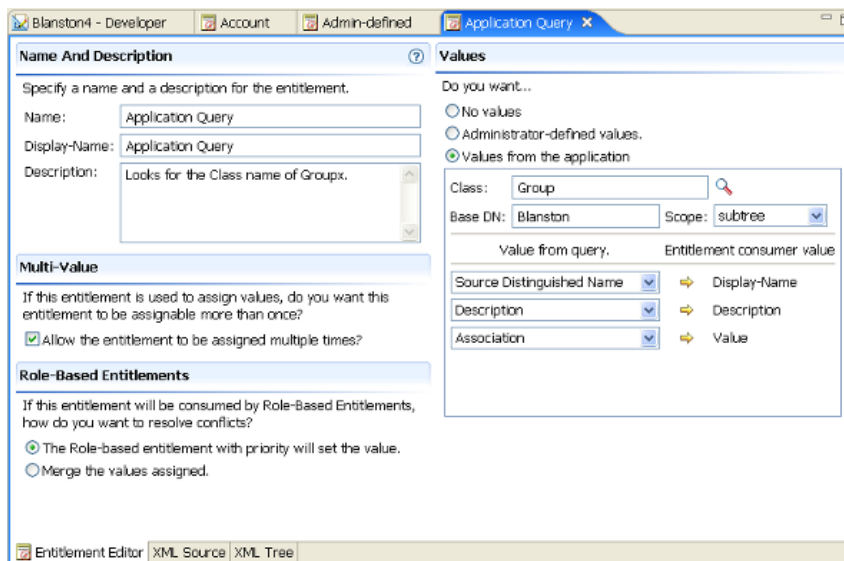
- 1 From the Outline view, right-click an entitlement that appears under the Subscriber channel of the selected driver, then click *Edit*.

or

Double-click the entitlement icon to bring up the entitlement in the Entitlement Editor.

You can also right-click the driver icon in the Modeler view, then select *Edit Entitlements*.

- 2 If you have more than one entitlement for the selected driver, you see the Edit Entitlements windows listing the available entitlements. Select an entitlement, then click *OK*. The entitlement appears in the Entitlement Editor.



The Entitlement Editor view shows you all of the pages and choices that you see in the Entitlement Wizard, but the information is on one page.

- ◆ **Name and Description:** Allows you to edit the name, the display name, and the description that you have given to this entitlement.
- ◆ **Multi-Value:** Allows you to select if you want an entitlement to be assigned multiple times.
- ◆ **Role-Based Entitlements:** Allows you to select conflict resolution for Role-Based Entitlements. If you do not select Role-Based Entitlements, *the Role-based entitlements with priority* button is the default.

- ♦ **Values:** Allows you to define how values are defined: no values, administrator defined values, or values from an application.

The information that appears in the Entitlement Editor depends on what you initially defined in the entitlement. If you choose to edit a valueless entitlement, the Values heading displays *No Values*. If you are editing a valued entitlement and you want to add values to a list, type the value in the *Value* field and click *Add*. If you want to remove a value, select the value in the Values list and click *Remove*.

If you don't want to select from a list, select *Administrator Defined Values* under the *Values* heading and leave the *Values* list blank. This gives you a blank text box in iManager or in the user application, and you can fill in the value there.

- 3 When you have made your changes to the entitlement, click the *Save* icon in the upper left corner of Designer, or click the *X* on the entitlement's tab to display a Save Resource window, allowing you to save changes (*Yes/No/Cancel*).

17.3.1 Entitlement XML Source and XML Tree Views

To view the entitlement in XML source code:

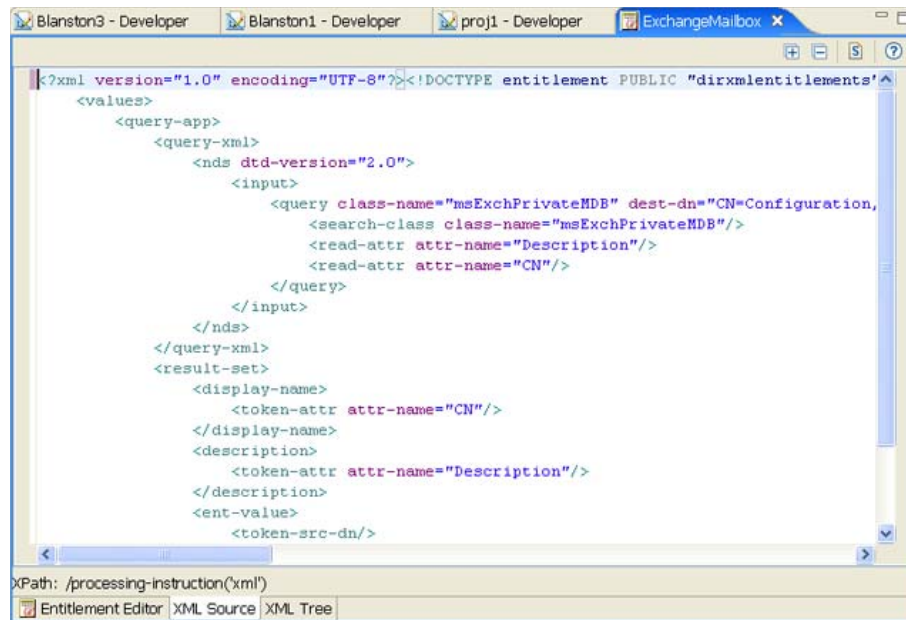
- 1 From the Outline view, right-click an entitlement that appears under the Subscriber channel of the selected driver, then click *Edit*.

or

Double-click the entitlement icon to bring up the entitlement in the Entitlement Editor.

You can also right-click the driver icon in the Modeler view, then select *Edit Entitlements*.

- 2 To see the XML Source view, click *XML Source* at the bottom of the Entitlement Editor view. The XML Source view shows the XML code in a formatted state.



The upper right corner of the XML Source view has the following selections:

Name	Description
<i>Expand All</i>	Allows you to see all items under the item that you have selected.
<i>Collapse All</i>	Allows you to collapse all items that you have selected.
<i>Attach XML Catalog Entry, XML Schema, or DTD</i>	Allows you to attach an XML Catalog entry, an XML schema file, or a DTD (Document Type Definition) file. For default Windows installation, the DTD for entitlements is found under C:\Program Files\Novell\Designer\eclipse\plugins\com.novell.designer.idm.entitlements_1.1.0\DTD\dirxmlentitlements.dtd.
Find/Replace (Ctrl+F)	Ctrl+F brings up the Find/Replace window, which allows you to query text, structure, and XPath searches in a forward or a backward direction. Other options include case sensitive, wrap search, whole word, incremental, and regular expressions search capabilities.
Help	Opens the Help view to the right of the XML Source view.

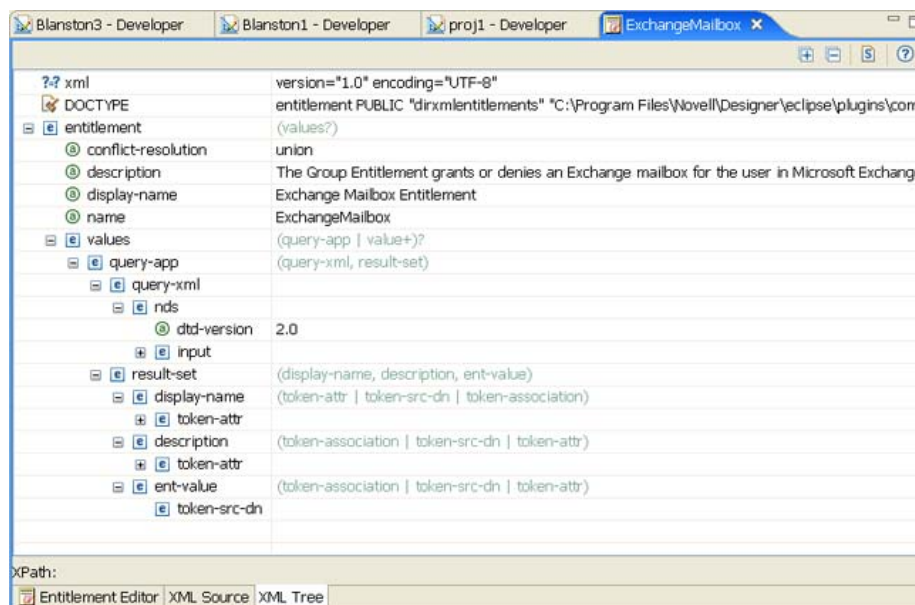
Right-clicking in the XML Source view brings up the following options:

- ◆ Undo Text Change (Ctrl+Z)
- ◆ Revert File
- ◆ Save
- ◆ Cut (Ctrl+X)
- ◆ Copy (Ctrl+C)
- ◆ Paste (Ctrl+V)
- ◆ Format the document or active elements
- ◆ Clear Validation Errors
- ◆ Validate
- ◆ Preferences

3 To see the XML Tree view, click *XML Tree* at the bottom of the Entitlement Editor view.

The XML Tree view is a tree control view of the XML source code. You can perform the same edits in this view as you can in the Entitlement Editor view or the XML Source view. To view

the entitlement in XML Tree view, select *XML Tree* at the bottom of the Entitlement Editor view.



The upper right corner of the XML Tree view menu contains the following selections:

Name	Description
<i>Expand All</i>	Allows you to see all items under the item that you have selected.
<i>Collapse All</i>	Allows you to collapse all items that you have selected.
<i>Attach XML Catalog Entry, XML Schema, or DTD</i>	Allows you to attach an XML Catalog entry, an XML schema file, or a DTD (Document Type Definition) file. For default Windows installation, the DTD for entitlements is found under C:\Program Files\Novell\Designer\eclipse\plugins\com.novell.designer.idm.entitlements_1.1.0\DTD\dirxmlentitlements.dtd.
Find/Replace (Ctrl+F)	Brings up the Find/Replace window, which allows you to query text, structure, and XPath searches in a forward or a backward direction. Other options include case sensitive, wrap search, whole word, incremental, and regular expressions search capabilities.
Help	Opens the Help view to the right of the XML Tree view.

Right-clicking in the XML Tree view can bring up a number of different options. For example, right-clicking the highlighted value on the right side presents the following options:

- ◆ Undo
- ◆ Cut
- ◆ Copy
- ◆ Paste
- ◆ Delete
- ◆ Select All

Right-clicking an attribute on the left side in the XML Tree view presents the following options:

- ◆ Remove
- ◆ Edit the Selected Attribute
- ◆ Replace with a value

Depending on what you select on the left side in the XML Tree view, you see different options. For example, right-clicking an element presents the following options:

- ◆ Remove Element
- ◆ Add New Attribute
- ◆ Add to a Child Element a Comment, a Processing Instruction, a PCDATA, a CDATA Section, a new Element
- ◆ Add Before a Comment, a Processing Instruction, a PCDATA, a CDATA Section, a new Element
- ◆ Add After a Comment, a Processing Instruction, a PCDATA, a CDATA Section, a new Element

17.3.2 Using the Novell Entitlement DTD

Some entitlements come predefined on drivers that have entitlements enabled. (For a list of these drivers with predefined entitlements see [Section 17.1.3, “Identity Manager Drivers with Preconfigurations that Support Entitlements,” on page 341.](#)) You can use these entitlements or you can create your own entitlements in iManager or Designer. To help you create your own entitlements, you can use the Novell Entitlement DTD as an example to create entitlements. For an example of the Novell Entitlement DTD and an explanation of its functionality, see [“Using Novell’s Entitlement Document Type Definition \(DTD\)”](#) and [“Explaining the Entitlement DTD”](#) sections in the Entitlements section of the *Novell Identity Manager 3.5.1 Administration Guide*.

17.4 Managing Entitlements

After you create entitlements (or use entitlements that come preconfigured with certain Identity Manager drivers), you need to manage them. Entitlements are tied into the eDirectory event system and granting and revoking are initiated through two agents:

- ◆ iManager through Role-Based entitlement policies
- ◆ The User Application as workflow entitlements

Role-Based Entitlements allow you to automatically grant or revoke business resources if the criteria are met. In order for workflow entitlements to work with the User Application, manual approval is first required.

For instance, you can specify that if user has A, B, and C, then through a Role-Based Entitlement, the user is made a member of Group H; but if the user has E and F qualifications, he or she is made a member of Group I. Through Role-Based entitlements, this action is done automatically, as long as the conditions are met. In order for this entitlement to work with workflow entitlements, the User object must first acquire approval, which you need to set up through the User Application. However, if you do not add to the driver the policies and rules to interpret the event in the designated system, granting and revoking entitlements has no effect.

Use either Role-Based Entitlements or workflow entitlements; it is a not good idea to mix them to manage the same resource. We recommend that you have only one agent control an entitlement. If multiple agents are in control, you have the following consequences:

- ◆ Whoever comes last controls the entitlement results
- ◆ Results are unpredictable
- ◆ Using both agents to control an entitlement is not supported by Novell

Scheduling Jobs

18

Designer has a job scheduling utility to schedule events. Through this utility, the system can be set to disable an account on a specific day, or to initiate a workflow to request an extension for a person's access to a corporate resource, such as:

- ♦ Create a Job object from an installed job definition.
- ♦ Define when a job is to run, which servers the job is to run on, the scope of the job in terms of eDirectory objects, and the a job reports intermediate and final results.
- ♦ Set values for the job's parameters, its description and display name.
- ♦ Enable/disable a job, manually start a job, stop a job that is running, and display a list of running jobs.

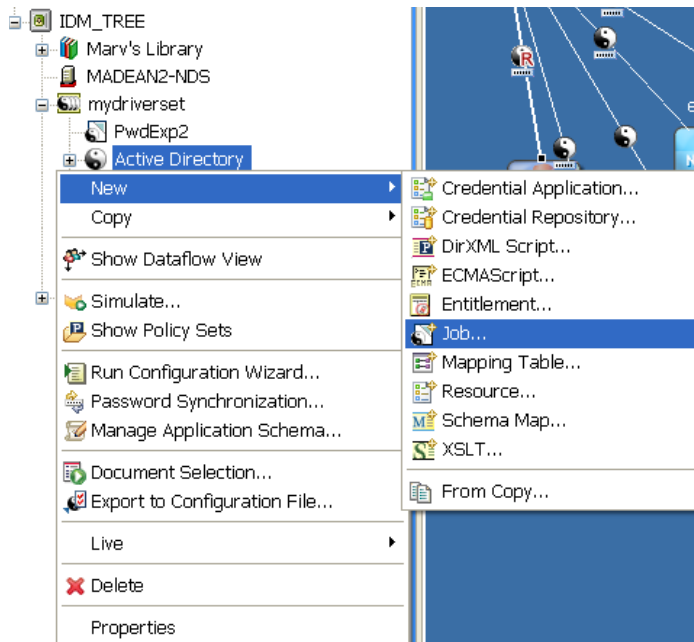
Designer's job scheduler contains the same functionality as the job scheduler found in iManager.

- ♦ [Section 18.1, "Creating a Job," on page 361](#)
- ♦ [Section 18.2, "Editing a Job," on page 363](#)

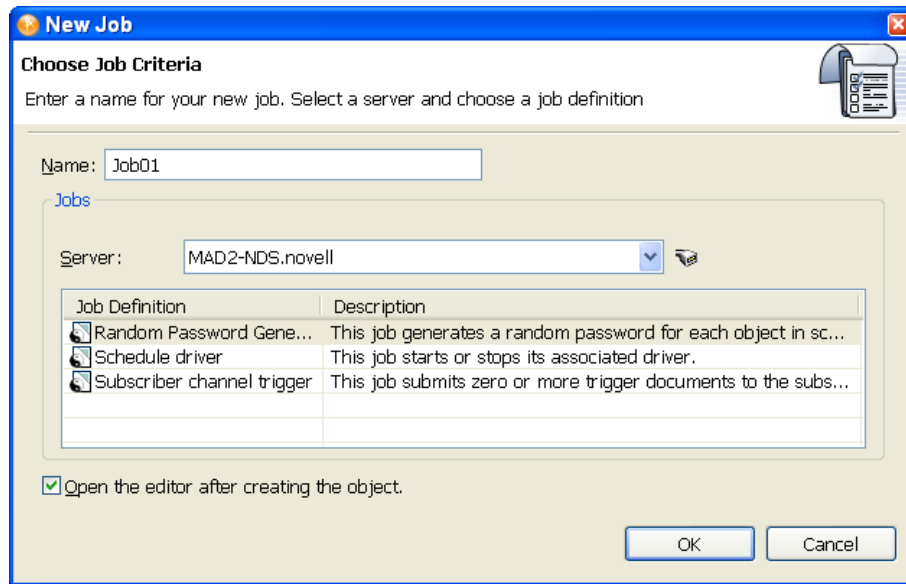
18.1 Creating a Job

To create a new job:

- 1 In the Outline view, right-click a driver and select *New > Job*.



- 2 In the New Job window, give the job a descriptive name, or use the name provided.



- 3 In the *Server* field, select the server to perform the job execution. You can click the drop-down item to select a different server than the one listed.
- 4 If the administrator has created and installed any custom job definitions on the selected server, click the *Update Job Definitions from Server* icon to the right of the server entry to read in a live list of available jobs from the server. Since Designer is an offline modeling tool, only the Identity Manager 3.5 job definitions show in the server's *Job Definition* list by default.
- 5 Choose a definition for the job. The New Job Wizard comes with three job definitions. Additional custom jobs might also be listed.
 - ♦ **Random Password Generator:** Generates a random password for each object in the job's scope. The password is generated by NMAS to match Password Policy object that the job references. Generally these Password Policy objects are not the same as those used for eDirectory user password policies.

The job submits the generated password one at a time to the driver's Subscriber channel. The Subscriber channel policies must do something useful with the passwords.
 - ♦ **Schedule Driver:** Starts or stops the associated driver. You can also toggle a driver to start the driver if it is stopped or to stop the driver if it is running.
 - ♦ **Subscriber Channel Trigger:** Submits zero or more trigger documents to the Subscriber channel. The submission can either be a document per object if a scope is defined, or it can be a single trigger event if no scope is defined.

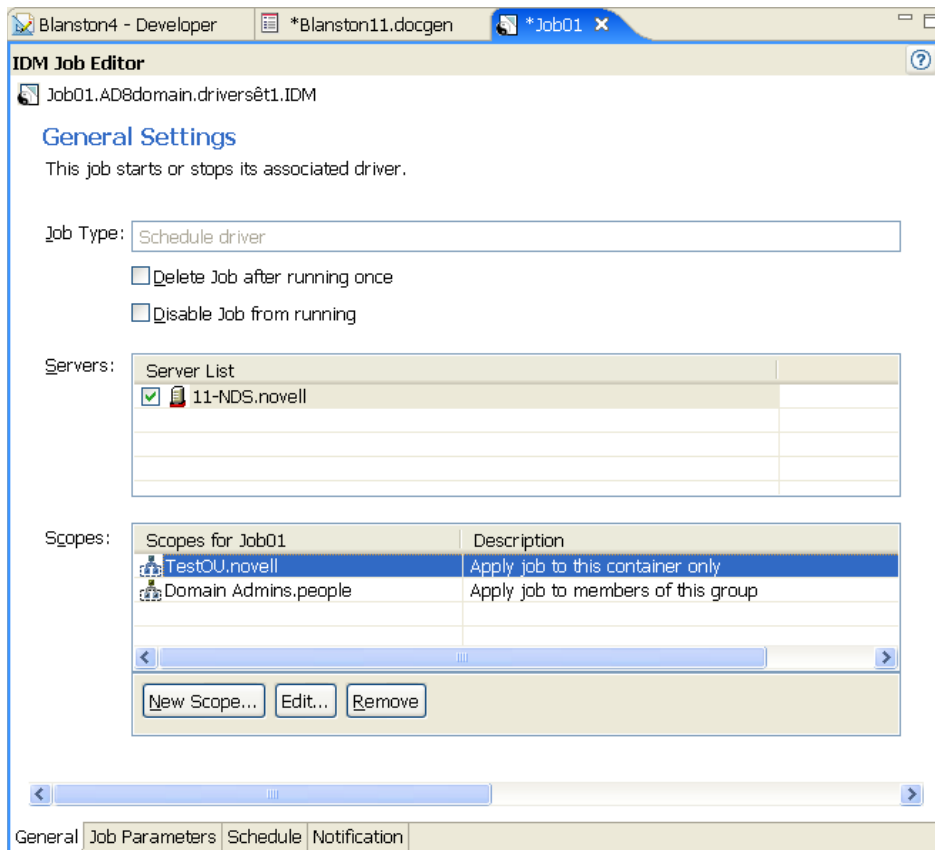
Trigger event documents identify the job and the scope object. A trigger event can bypass the cache and "go to the head of the queue" if desired. You will likely use trigger jobs the most and they allow you to use driver policies that you can customize for your personal requirements.
- 6 Decide if you want to edit the job. The *Open the editor after creating the object* option opens the newly created job in the IDM Job Editor window after it first saves the job object. If you do not want to open the editor at this time, deselect the editor.
- 7 Click *OK*.

- 8 The File Conflict window gives you the option to save the job object and continue. If you want to create the job object and continue with the IDM Job Editor, click *Yes*. Otherwise, click *No*.
- 9 Continue with [Section 18.2, “Editing a Job,” on page 363](#).

18.2 Editing a Job

After you create a job, you need to add the necessary information to make the job useful. To edit a job, double-click a newly created job in the Outline view to bring up the job in the IDM Job Editor view.

Figure 18-1 *The IDM Job Editor View*



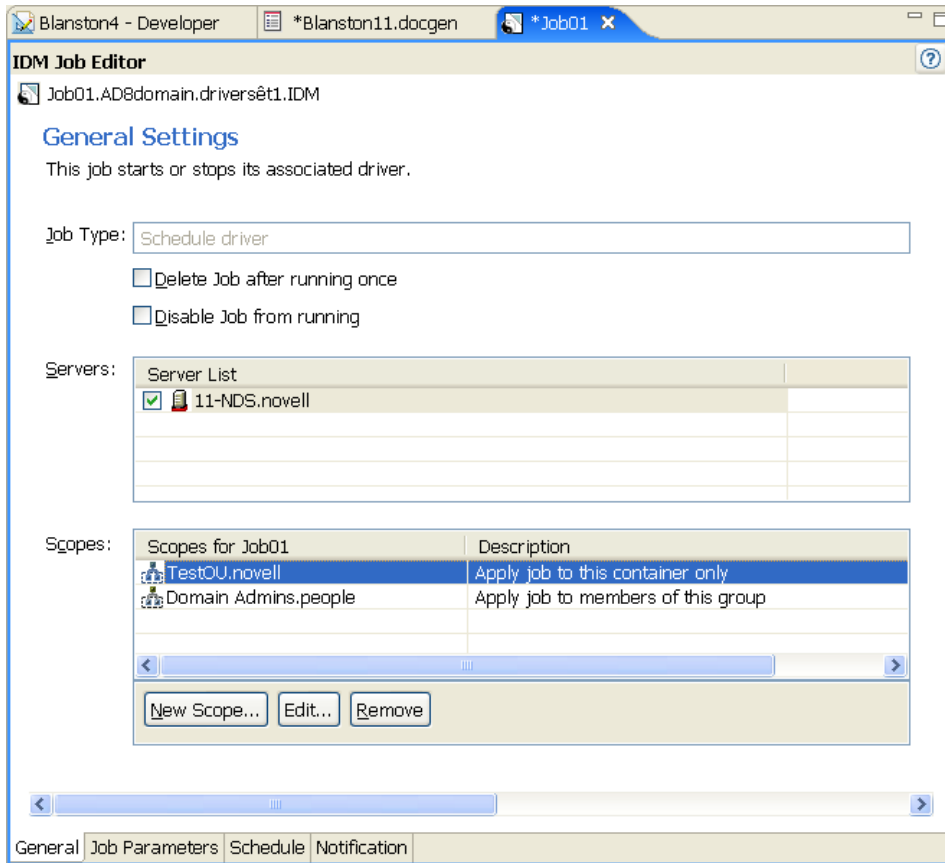
The IDM Job Editor has four tabs at the bottom of its view:

- ◆ [Section 18.2.1, “Job Editor Selections Under the General Tab,” on page 363](#)
- ◆ [Section 18.2.2, “Job Editor Selections Under the Job Parameter Tab,” on page 367](#)
- ◆ [Section 18.2.3, “Job Editor Selections Under the Scheduler Tab,” on page 371](#)
- ◆ [Section 18.2.4, “Job Editor Selections Under the Notification Tab,” on page 373](#)

18.2.1 Job Editor Selections Under the General Tab

The top of the General tab shows the Java class name of the job. This is followed by the job type, which shows the type of job you selected. Under the Job Type heading, you can enable or disable the job, or to delete the job after it runs.

Figure 18-2 Items Under the General Tab

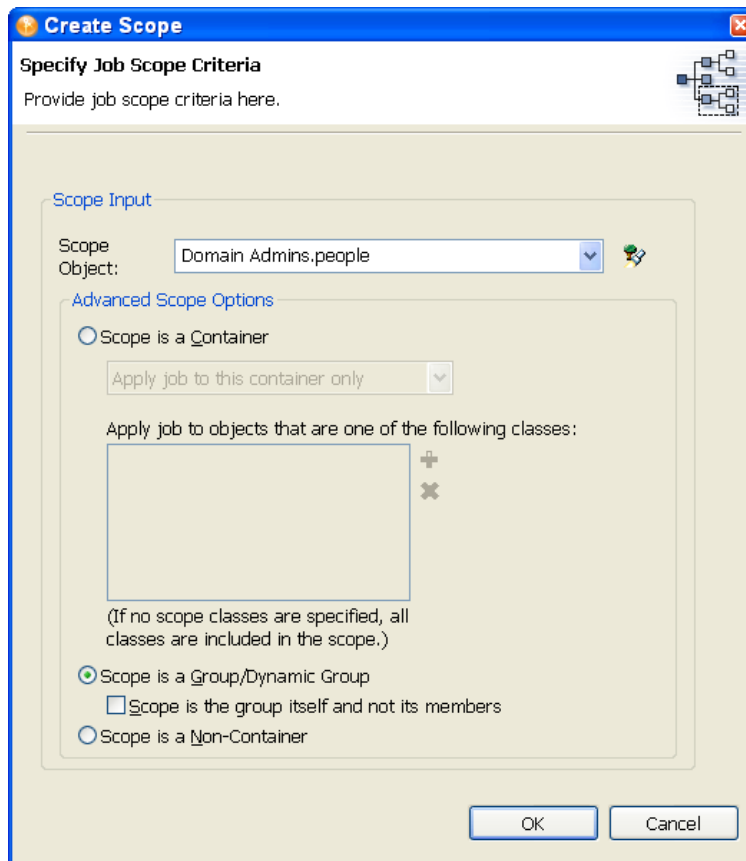


- 1 To delete the job after it runs, select *Delete job after running once*.
- 2 To disable the job from running, select *Disable job from running*.
- 3 In the *Server List* column, select the server or servers where this job should run.

A filtered list of servers is available to help you assign this job. A custom job can be installed on one server but not on another. In this case, the server without this custom job is filtered out of the Server List.

A job can be assigned to multiple servers as long as it has been installed on each server. Designer only allows this association if the jobs are properly installed and packaged so that the Metadirectory engine can see them.

- 4 To add a scope to the *Scopes* column, click *New Scope*.

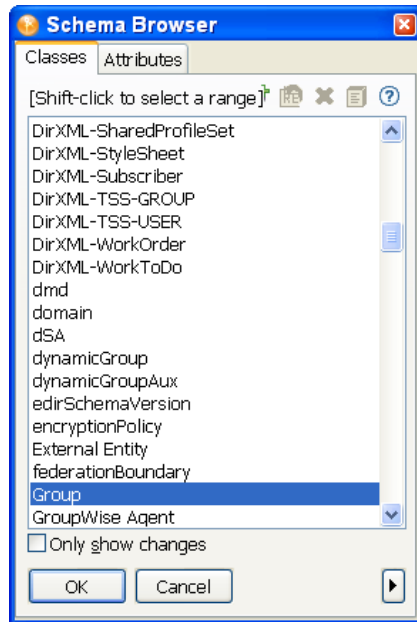


- 5 To select a scope object, type the Distinguished name of the object or use the *Browse* icon to browse to the object. Click *OK* to add the scope object.

Scopes allow you to define the objects that this job applies to. An object in eDirectory™ can be a container, a dynamic group, a group, or a leaf object. If you select a group object, you can apply the job to the group's members, or only to the group. If you select a container object, you can apply the job to all descendants in that container, to all of the children in the container, or to the container only.

- 6 If the object is a container, select *Scope is a Container*. Then select how you want to apply the job:
- ♦ Apply job to this container only
 - ♦ Apply job to children of this container
 - ♦ Apply job to all descendants of this container
- 7 (Optional) If you select *Apply job to children of this container* or *Apply job to all descendants of this container*, you can specify the classes and attributes you want to scope. Click the Plus

icon to bring up the Schema Browser window to select the classes you want to scope. Select the class schema, then click *OK*.



The classes are added to the Classes box. To remove a class, select it and click the Minus icon.

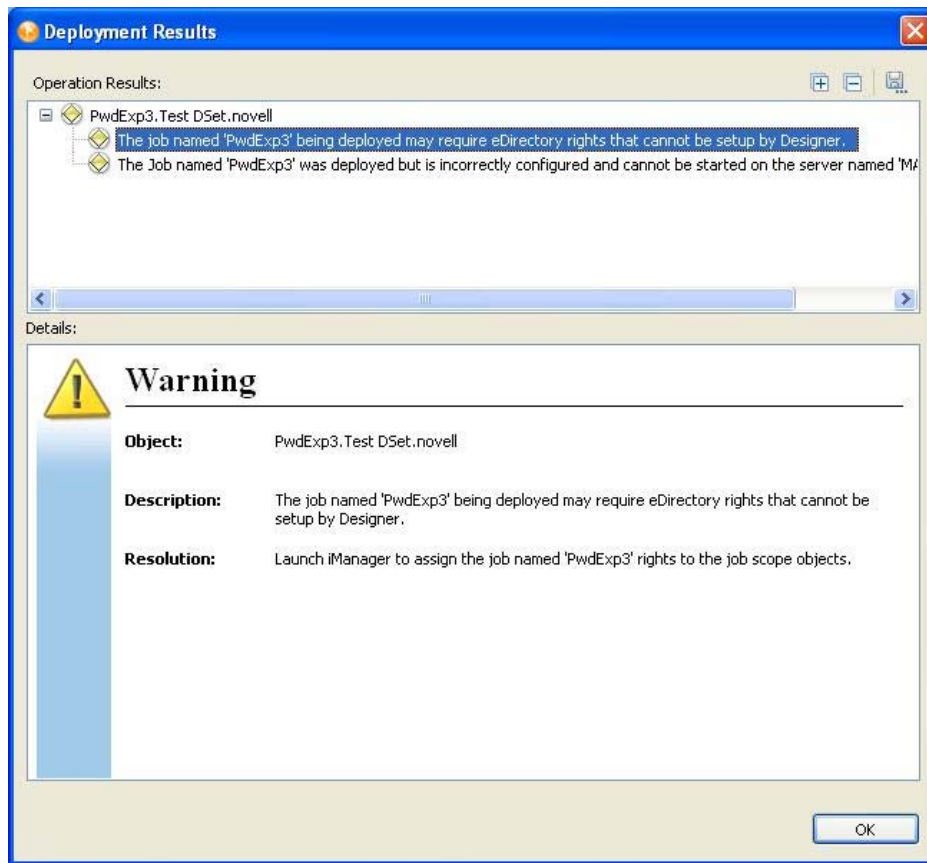
- 8 If the object is a group or a dynamic group, select *Scope is a Group/Dynamic Group*. You can then select the *Scope is the group itself and not its members* option if the scope is for the group.
- 9 If the object is a non-container, select *Scope is a Non-Container*.
- 10 After the scope criteria are selected, click *OK* to return to the General Settings page.
- 11 If you need to edit a scope, select the scope name, then click *Edit*.
- 12 To remove a scope, select the scope name, then click *Remove*.

Deploying a Job with Scope Objects

Jobs might need access to eDirectory data and certain Identity Manager actions, such as starting and stopping drivers. Such access is subject to eDirectory rights assignments and is controlled by the rights that are granted to the `DirXMLJob` object. While Identity Manager actions are controlled by special attributes, normal eDirectory rights are needed for data reads and writes.

When you deploy a job object that has scope objects, there might be eDirectory rights assignments that Designer cannot properly set up. The rights needed to complete the task depend on the scope objects that are assigned to the job object.

Figure 18-3 Warning Messages When Deploying a Job with Scope Objects



If you see this warning when deploying job objects, use the iManager utility to assign eDirectory rights to the job object so it can properly access the job scope objects and complete its task.

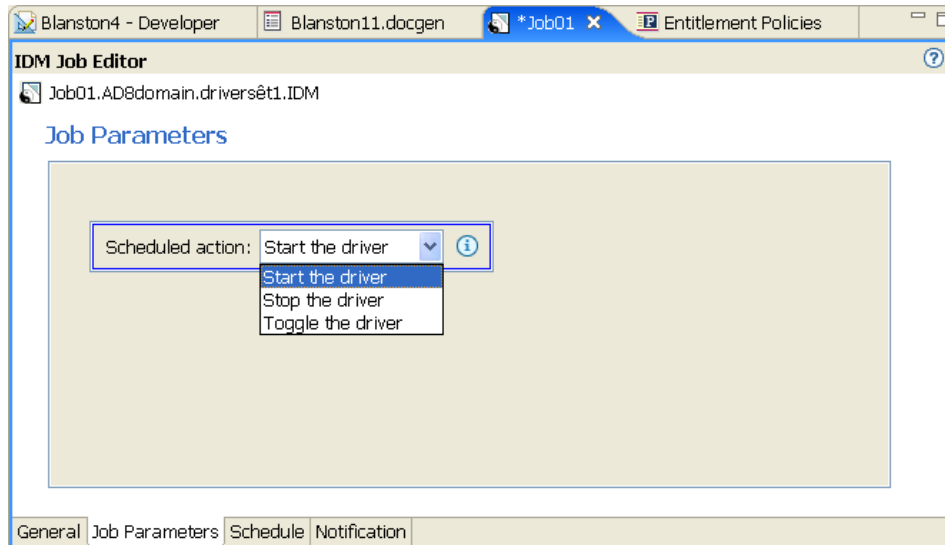
18.2.2 Job Editor Selections Under the Job Parameter Tab

The Job Parameter page allows you to add additional parameters to the job and to view the parameters as they are presently set up. What you can do depends on the type of job you selected.

- ♦ [“Parameters for the Schedule Driver Job” on page 368](#)
- ♦ [“Parameters for the Generate Random Passwords Job” on page 369](#)
- ♦ [“Parameters for the Subscriber Channel Trigger Job” on page 370](#)

Parameters for the Schedule Driver Job

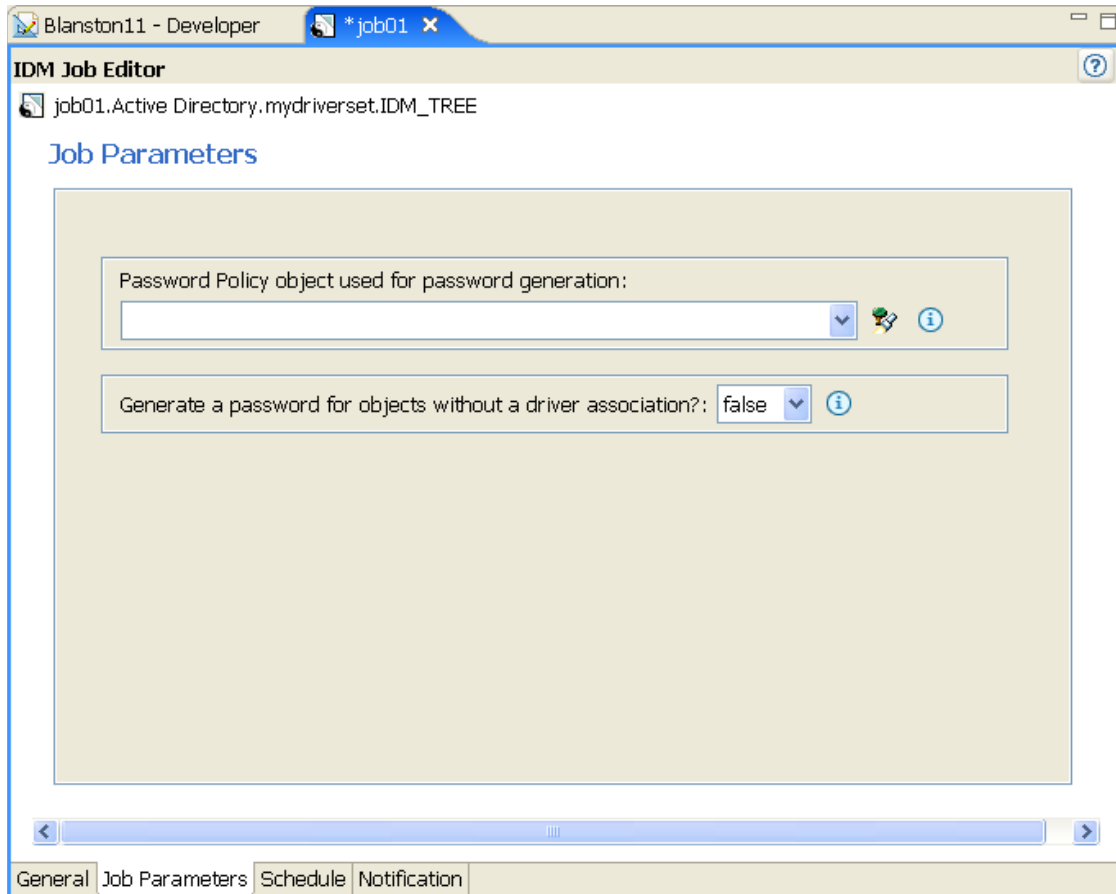
Figure 18-4 The Job Parameter Page for a Schedule Driver Job



- 1 If you want the job to start the driver, select *Start the driver*.
- 2 If you want the job to stop the driver, select *Stop the driver*.
- 3 If you want the job to stop the driver if it's started or to start the driver if it's stopped, select *Toggle the driver*.

Parameters for the Generate Random Passwords Job

Figure 18-5 The Job Parameter Page for the Generate Random Password Job



- 1 Type the Password policy object's Distinguished name, or use the *Browse* icon to select the Password policy you want to use for password generation.
- 2 If you want to generate passwords for scoped objects without a driver association, select *True*. Otherwise, select *False*.

Parameters for the Subscriber Channel Trigger Job

Figure 18-6 The Job Parameter Page for the Subscriber Channel Trigger Job

The screenshot shows the 'IDM Job Editor' window with the 'Job Parameters' tab selected. The job name is 'Job3.Active Directory.mydriverset.IDM_TREE'. The parameters are as follows:

- Submit a trigger document for objects without a driver association?: false
- Use Job CN as trigger document identifier?: false
- Trigger element source value: 1
- Method for submitting trigger documents: direct (bypass cache)
- Start driver if not running: true
- Stop driver when finished processing trigger(s): true

At the bottom, there are tabs for 'General', 'Job Parameters', 'Schedule', and 'Notification'.

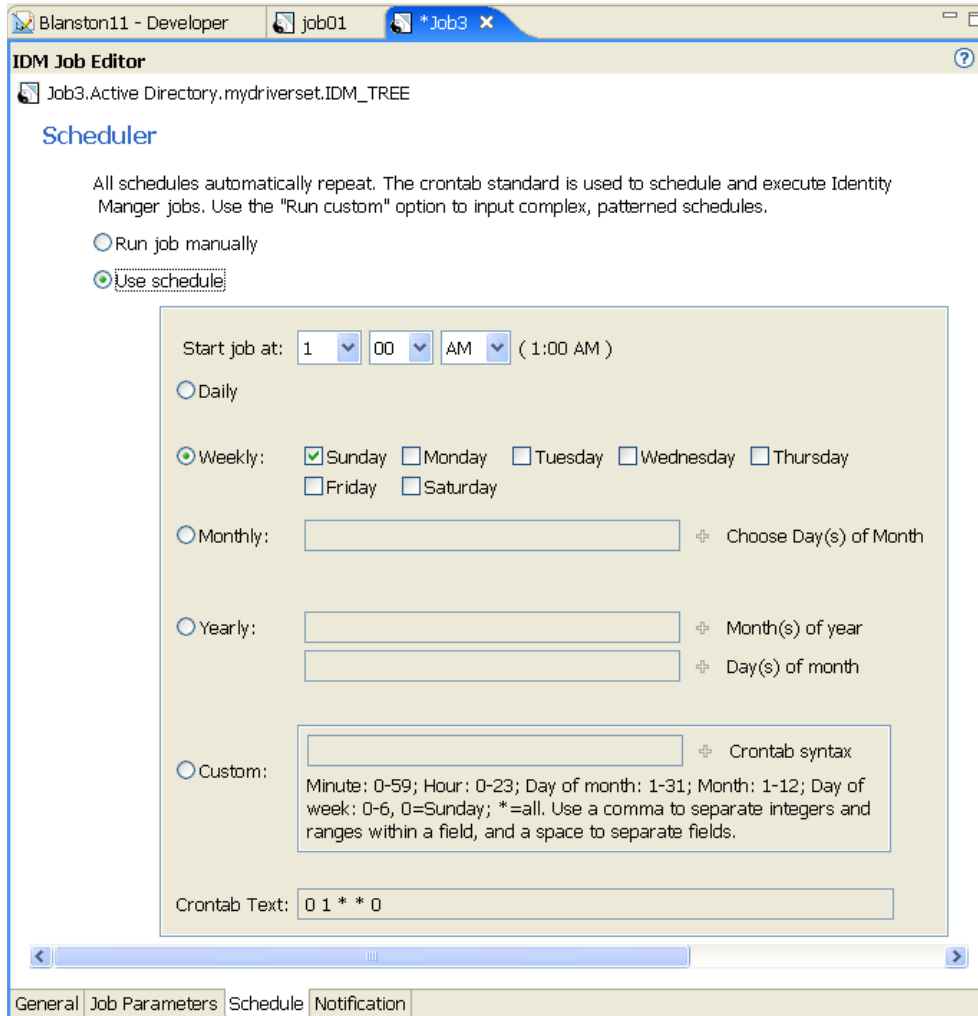
- 1 If you want to submit a trigger document for scoped objects that do not have a driver association, select *True*. Otherwise, keep the default of *False*.
- 2 If you want to use the job's Common Name (CN) as a document identifier trigger, keep the default of *True*. Otherwise, select *False*.
- 3 (Optional) If you select *False*, enter the string that the job can use as the value for the trigger element's "Source" attribute.
- 4 Select a method for submitting the trigger documents. If you want to queue the job the trigger is from, keep the default of *Queue (use cache)*. Otherwise, select *Direct (bypass cache)*.
- 5 (Optional) If you select *Direct (bypass cache)*, you are presented with the *Start driver if not running* option. If you want to start the driver if it is not running, keep the default of *True*. Otherwise, select *False*.
- 6 (Optional) If you select *True* on the *Start driver if not running* option, you are presented with the *Stop driver when finished processing triggers* option with the default of *True*. Use the default to stop the driver once it finishes processing the trigger job, or select *False* to keep the driver running.

Customized job definitions will have their own parameter set.

18.2.3 Job Editor Selections Under the Scheduler Tab

The Scheduler tab allows you to set up when you want to run the job.

Figure 18-7 The Job Options for the Scheduler Tab



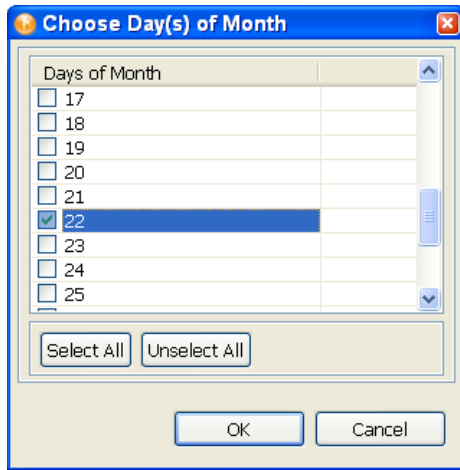
- 1 Select the *Use schedule* option to set the date and time, and whether to run the job daily, weekly, monthly, yearly.

Or

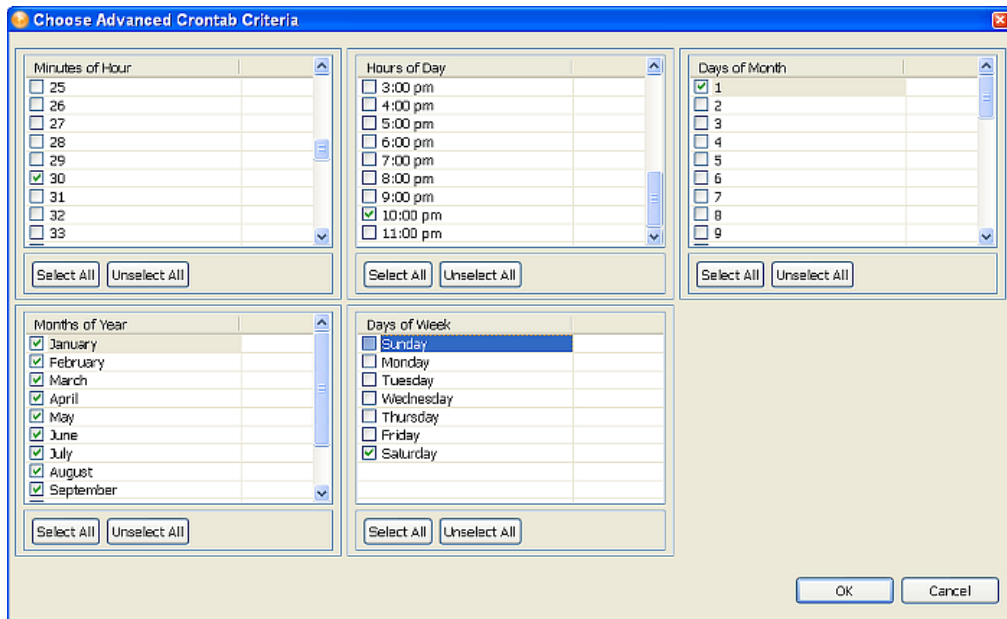
Select the *Run job manually* option to run the job when you choose to.

- 2 With *Use schedule* selected, set the time when you want the job to start running. Use the drop-down menus to select the hours, minutes, and AM or PM. The default is 1:00 AM.
- 3 If you want to run the job repeatedly, use the *Daily*, *Weekly*, *Monthly*, *Yearly*, or *Custom* fields to select when you want it to run.

For example, if you want the job to run weekly, select *Weekly*, then the day you want it to run on. If you want the job to run once a month, select *Monthly*, then click the *Plus* icon to select the day of the month.



- 4 If you select *Custom*, you can choose minutes, hours, days, months and days of the week from the Choose Advanced Crontab Criteria page.



The Advanced Crontab Criteria page default has everything selected. Click *Unselect All*, choose the time and days you want to run the job, then click *OK* to return to the Scheduler page.

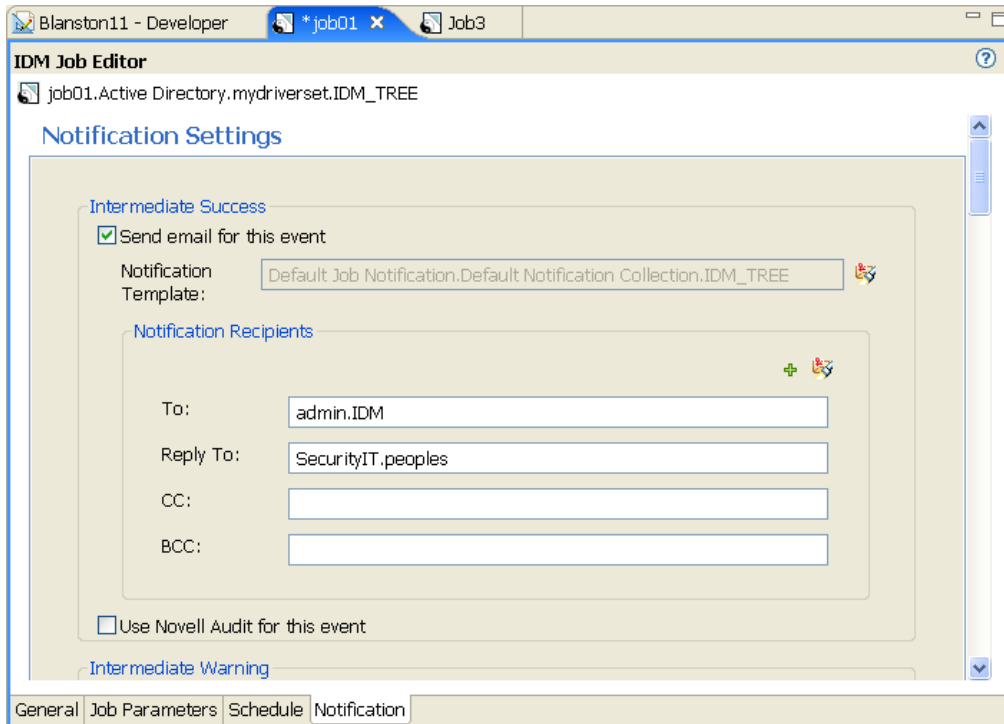
The information displayed in the *Crontab Text* field displays any settings you make on the Scheduler page. For example, if you click *Monthly* and select two days, those two days are displayed in the *Crontab Text* field.

18.2.4 Job Editor Selections Under the Notification Tab

The Notification Settings page allows you to define what you want to do with the job results. It is divided into two parts, Intermediate and Final, with the Success, Warning, Error, and Aborted results for each part.

The Notification Settings page allows you to set how you want to be notified for each result. Actions include sending an audit result or sending an e-mail when the result completes.

Figure 18-8 The Job Options Under the Notification Tab



- 1 If you select *Send email for this event*, Designer supplies the *Default Job Notification.Default Notification.security* template in the *Notification Template* field. To change templates, click the *Model Browser* icon to select another template.

- 2 Under Notification Recipients, select who you want to send the results to by typing in the user's or group's fully Distinguished name. You can use the *Plus* icon to create a mail profile.

Create Mail Recipient Profile

Recipient Profile

Enter email recipients. To and Reply are required fields.

Name: Network Administrator

Email Recipients

To: admin.IDM

Reply To: SecurityIT.peoples

CC:

BCC:

OK Cancel

The To and Reply fields are required for a profile.

- 3 When you have filled in the information, click *OK*.
- 4 If you want the results to go to Novell Audit, select *Use Novell Audit for this event*.
- 5 Use Steps 1 through Step 4 for each of the options:
 - ◆ Intermediate Success
 - ◆ Intermediate Warning
 - ◆ Intermediate Error
 - ◆ Intermediate Abort
 - ◆ Final Success
 - ◆ Final Warning
 - ◆ Final Error
 - ◆ Final Abort

If you do not select an option, no action is taken for the result.

The Deploy feature in Designer places a project, a set of drivers, a single driver, channels, and policies into a deployed Identity Manager system in an eDirectory™ tree. This can be a production tree or a test tree.

Use the Export feature to make backups of all of your projects and the drivers you want to implement. This way, if something happens to the driver in production, you have a backup.

Use the Deploy feature after you have thoroughly tested the policies that make up your drivers. To test policies, use the Policy Simulator (right-click a policy to see the simulation results of the policy that is being tested) or use the Project Checker to ensure that the project is valid. Then use Deploy to test the policy in a test environment; after you test the driver, you can deploy the driver into production.

You can also use the Import feature to import an existing eDirectory driver, a channel, or a policy; after it is imported, you can modify the object or objects, run the Policy Simulator to ensure that the object is working correctly, then deploy the object back into a test tree for further analysis. For more information about policies, see [Understanding Policies for Identity Manager 3.5 \(http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy/data/bookinfo.html\)](http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy/data/bookinfo.html).

To help you decide on changes to make before deploying, you can use the Compare feature to see differences between the objects you are deploying and those that already reside in an eDirectory tree. See [Section 19.7, “Using the Compare Feature When Deploying,” on page 383](#).

- ◆ [Section 19.1, “Preparing to Deploy,” on page 375](#)
- ◆ [Section 19.2, “Deploying a Project to an Identity Vault,” on page 376](#)
- ◆ [Section 19.3, “Deploying a Driver Set to an Identity Vault,” on page 376](#)
- ◆ [Section 19.4, “Deploying a Driver to an Identity Vault,” on page 379](#)
- ◆ [Section 19.5, “Deploying a Channel to an Identity Vault,” on page 381](#)
- ◆ [Section 19.6, “Deploying a Policy to an Identity Vault,” on page 382](#)
- ◆ [Section 19.7, “Using the Compare Feature When Deploying,” on page 383](#)
- ◆ [Section 19.8, “Troubleshooting Deployed Objects,” on page 387](#)
- ◆ [Section 19.9, “Exporting a Project,” on page 387](#)
- ◆ [Section 19.10, “Exporting To a File,” on page 390](#)

19.1 Preparing to Deploy

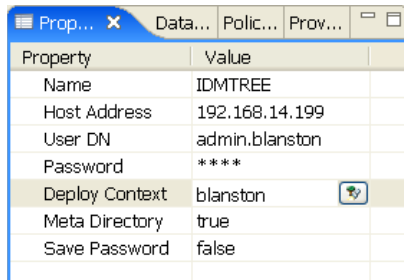
Before deploying a project, run Project Checker and fix any errors that appear.

- 1 Click *Window > Show View > Project Checker*, then click the *Run the Project Checker* icon.

After you have corrected any problems to the project, make a backup copy of the project before deploying.

Before you deploy objects into an Identity Vault, you need to designate the Deployment DN (distinguished name), or the place in the tree where you plan on deploying the Identity Manager project or objects. To do this:

- 1 In Designer, select the Identity Vault that contains the object or objects you want to deploy, then look in the Properties view below the Project/Outline view. (You can also open the Identity Vault's or driver's Properties window.)



The screenshot shows a window titled 'Prop...' with a tab 'Data...'. It contains a table with the following data:

Property	Value
Name	IDMTREE
Host Address	192.168.14.199
User DN	admin.blanston
Password	****
Deploy Context	blanston
Meta Directory	true
Save Password	false

- 2 In the Properties view, fill in the Identity Vault's Name, Host Address, User DN, Password, and Deployment DN information if it is not already present.
- 3 Click the *Browse* button to find the Deploy Context distinguished name on an existing tree if the other information is accurate and Designer can attach to the tree. You need this information to deploy anything, even a policy.

You can also use the driver set's Deploy Context entry if you want to deploy a driver set to a different context than the one designated in the Identity Vault's Properties view. The driver set's Deploy Context entry overrules the Identity Vault's Deploy Context entry.

IMPORTANT: You must have enough rights to access the eDirectory tree that is associated with the Identity Vault to which you want to deploy.

19.2 Deploying a Project to an Identity Vault

To deploy a project to an eDirectory tree that is running Identity Manager, you use the same procedure that you use for deploying a driver set, a driver, channels, or policies. The procedure is described in [Section 19.3, "Deploying a Driver Set to an Identity Vault,"](#) on page 376.

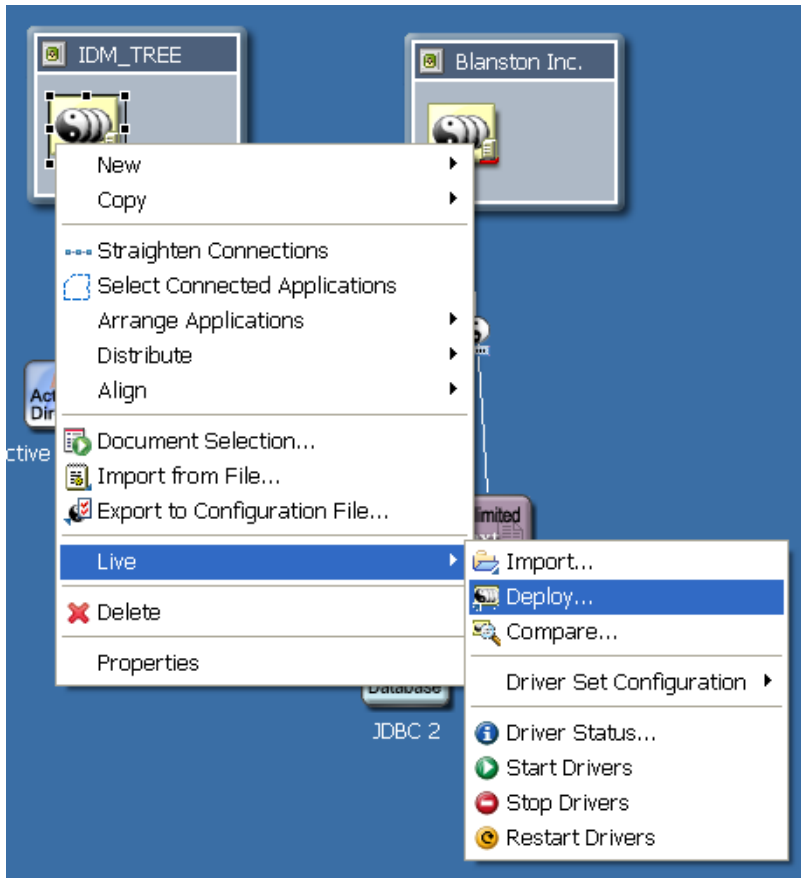
To deploy an Identity Manager-based project or an object in a project, you must have access to the eDirectory tree that is associated with the Identity Vault you are designing. You also need to know the deployment DN (distinguished name) context, or the place in the tree where you plan to deploy the Identity Manager driver set or driver objects.

19.3 Deploying a Driver Set to an Identity Vault

Suppose you finish a new driver set that you want to deploy into a test tree, or suppose you have imported a driver set, made modifications, and now you want to deploy the driver set back into its

working tree. Use the following procedure to deploy an Identity Manager Driver Set object (and all contained Identity Manager drivers) into an existing Identity Manager system in an eDirectory Tree:

- 1 Right-click the Driver Set icon in the Modeler view, then click *Live > Deploy*.

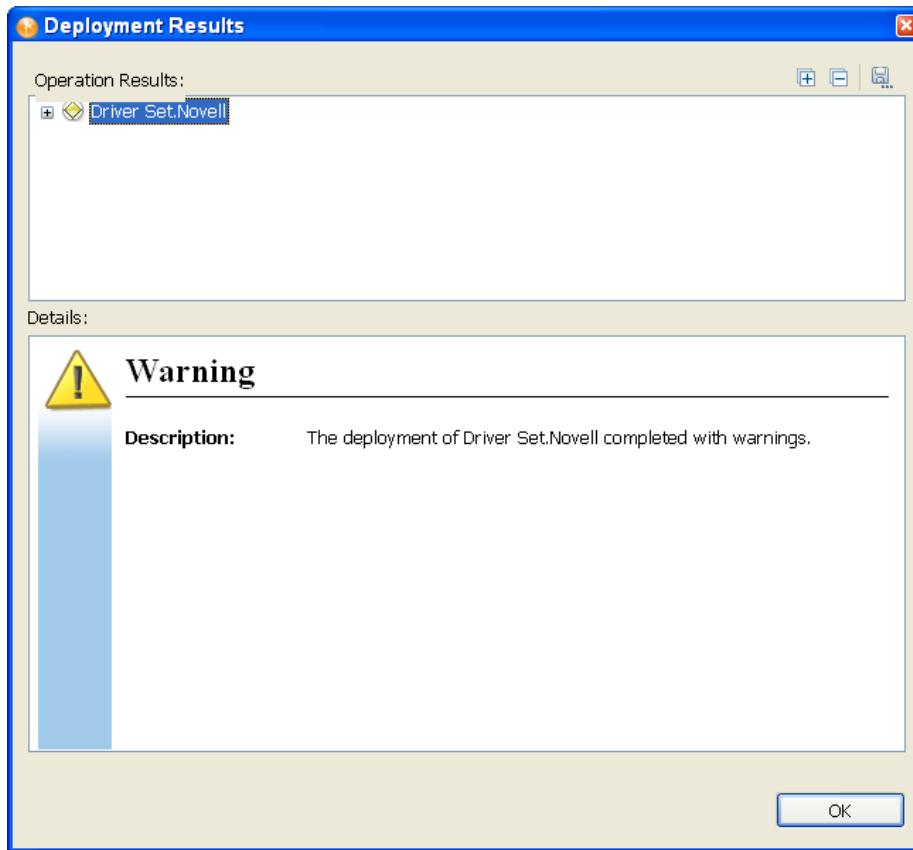


You can also select the Driver Set object from the Outline view. Click the *Outline* tab, right-click the Driver Set object, then click *Live > Deploy*.

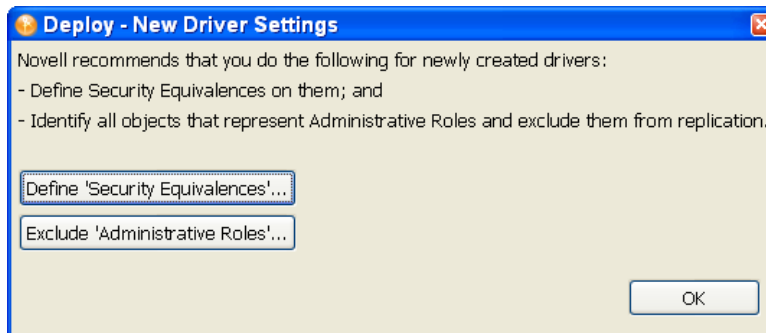
The Identity Vault Credentials window displays if Designer can't authenticate to the eDirectory tree specified in the Identity Vault, or if you do not have the Deployment DN designated in the Properties tab of the Identity Vault you are deploying to.

- 2 Use the Compare feature to see differences between the objects you are deploying and those that already reside in an eDirectory tree. See [Section 19.7, "Using the Compare Feature When Deploying,"](#) on page 383.
- 3 In the Deployment Summary window, click *Deploy*.

- 4 Click *OK* to close the Information window.



- 5 (Conditional.) If you see other informational messages, decide what action to take.
You might also see a message in the Deployment Results window stating that the deployment was unsuccessful. Click on the error messages in the Operation Results portion of the window to see the error descriptions and possible reasons in the Details portion.
- 6 (Conditional) If this is a new deployment, the Deploy - New Driver Settings window displays. Define security equivalences on the driver set and identify all objects that represent Administrative roles and exclude them from being replicated.



In both instances, Novell recommends that you select the Admin object, and any other objects that qualify in your network environment.

19.3.1 eDir-to-eDir Deployments and SSL/TLS

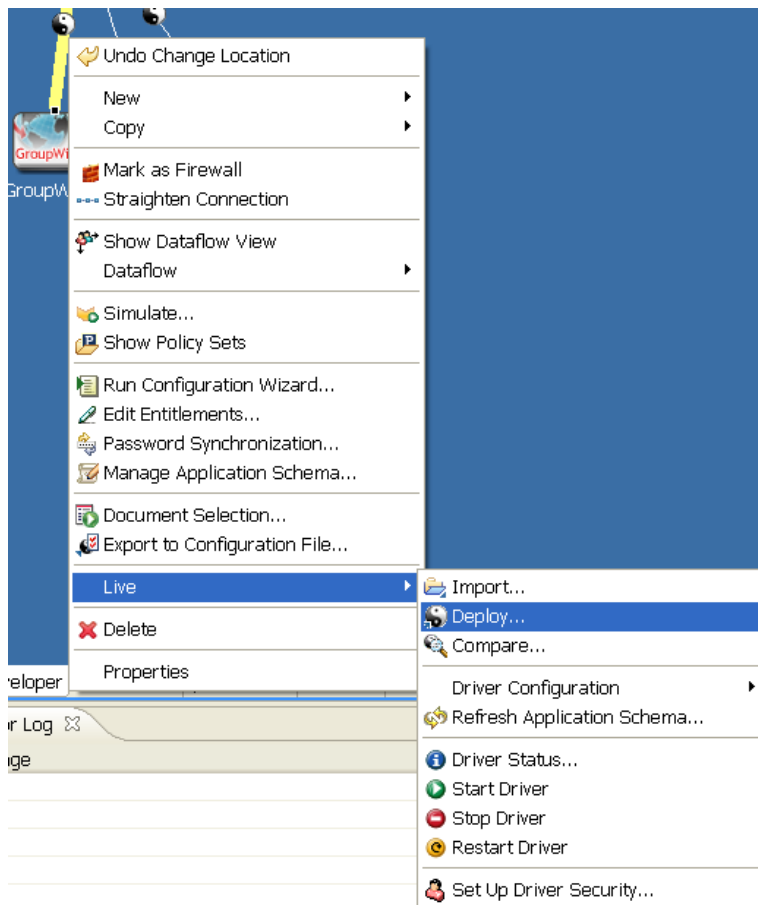
By default, always deploy both sides of an eDirectory-to-eDirectory connection when you have SSL and TLS enabled. If SSL/TLS are enabled, Designer creates the certificates in the eDirectory tree when you deploy the drivers. SSL and TLS are not enabled nor configured by default.

To check your present SSL settings, select *Window > Preferences > Designer for IDM > Configuration > eDir-to-eDir SSL/TLS*. Once configured, the Deploy feature adheres to the SSL preference settings under *Certificate Overwrite Policy*.

19.4 Deploying a Driver to an Identity Vault

Suppose you finished a new driver object that you want to deploy into a test tree, or suppose you have imported a driver object, made significant modifications, and now you want to deploy that driver object back into its working tree. Use the following procedure to deploy an Identity Manager Driver object (and all contained channels and policies) into a driver set:

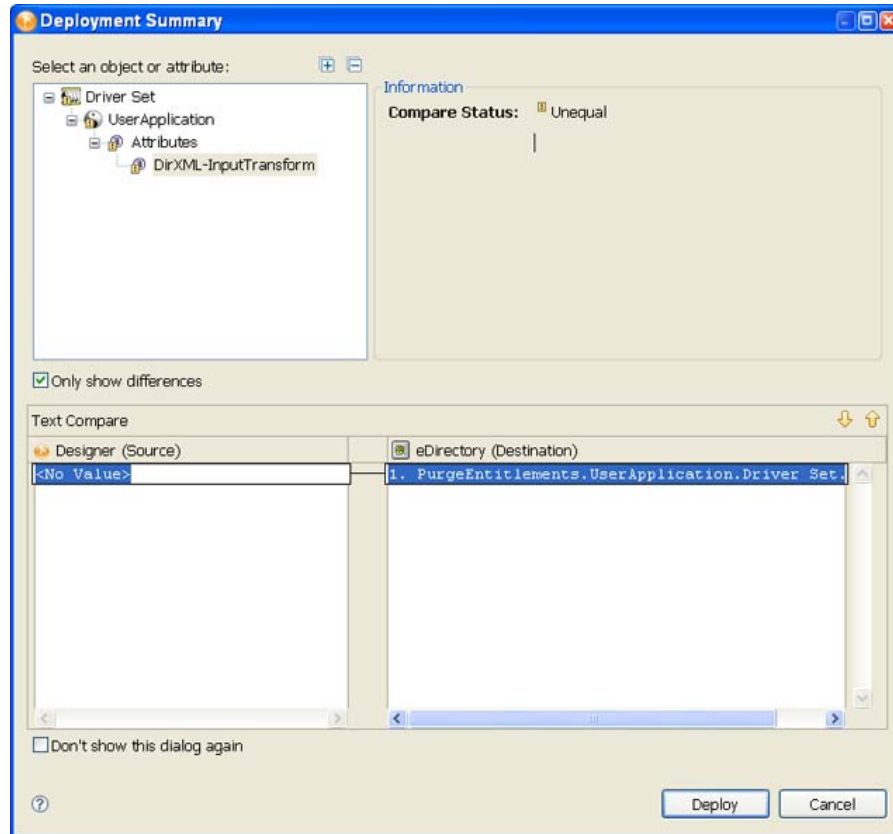
- 1 Select an Identity Vault in the Modeler view.
- 2 Right-click a driver object connected to a Driver Set icon in the Identity Vault.
The driver object is represented by a circle icon.
- 3 Click *Live > Deploy*.



You can also select the driver object from the Outline view. Click the *Outline* tab, right-click the driver object you want to deploy, then click *Live > Deploy*.

An error displays if Designer can't authenticate to the eDirectory tree specified in the Identity Vault, or if you do not have the Deployment DN designated in the Properties tab of the Identity Vault you are deploying to.

The Deployment Summary window shows you the differences between the objects you are deploying and those that already reside in an eDirectory tree. It is the same as the Compare feature. For more information on how to use the Compare window, see [Section 19.7, "Using the Compare Feature When Deploying,"](#) on page 383.



- 4 Click *Deploy* to begin the process.
- 5 Click *OK* to close the Deployment Results window.
- 6 (Conditional.) If you see other informational messages, decide what action to take.

You might also see a message in the Deployment Results window stating that the deployment was unsuccessful. Click on the error messages in the Operation Results portion of the window to see the error descriptions and possible reasons in the Details portion.

- 7 (Conditional) If this is a new deployment, the Deploy - New Driver Settings window displays. Define security equivalences on the driver set and identify all objects that represent Administrative roles and exclude them from being replicated.



In both instances, Novell recommends that you select the Admin object, and any other objects that qualify in your network settings.

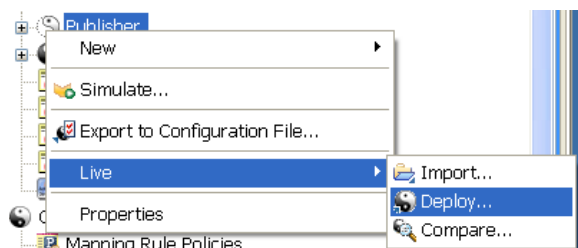
19.5 Deploying a Channel to an Identity Vault

A channel is a grouping of rules and policies, and Designer allows you to deploy a channel object into a driver if necessary. The Subscriber and Publisher channels describe the direction in which the information flows. The Subscriber channel takes the event from Identity Vault (eDirectory) and sends that event to the connected system (application, database, CSV file, etc). The Publisher channel takes the event from the application, database, CSV file, etc., and sends that event to the Identity Vault. The Subscriber and Publisher channels act independently; actions in one are not affected by what happens in the other.

Channel objects must be a part of a newly created driver, or they must be a part of an existing driver that now needs to be modified. Driver objects are created through the Designer or iManager utilities. Because channel objects are a part of a driver object, you deploy a channel object into an existing driver object. If you simply deploy the channel object, Designer creates a skeleton driver as a placeholder for the channel object.

To deploy an Identity Manager channel (a Subscriber channel or a Publisher channel) object and all contained policies into a driver in an Identity Vault:

- 1 In the *Outline* tab, select the channel object under the driver object. The driver object is represented by a circle icon; the Publisher icon shows a black dot on the icon and the Subscriber icon shows a white dot.
- 2 Right-click the channel object you want to deploy, then click *Live > Deploy*.



An error displays if Designer can't authenticate to the eDirectory tree specified in the Identity Vault, or if you do not have the Deployment DN designated in the Properties tab of the Identity Vault you are deploying to.

The Deployment Summary window shows you the differences between the objects you are deploying and those that already reside in an eDirectory tree. It is the same window format as the Compare feature. For more information on how to use the Compare window, see [Section 19.7, "Using the Compare Feature When Deploying," on page 383](#).

An error displays if Designer can't authenticate to the eDirectory tree specified in the Identity Vault, or if you do not have the Deployment DN designated in the Properties tab of the Identity Vault to which you are deploying.

- 3 In the Deployment Summary window, click *Deploy*.
- 4 After the channel deploys, click *OK* to close the Deployment Results window.

19.6 Deploying a Policy to an Identity Vault

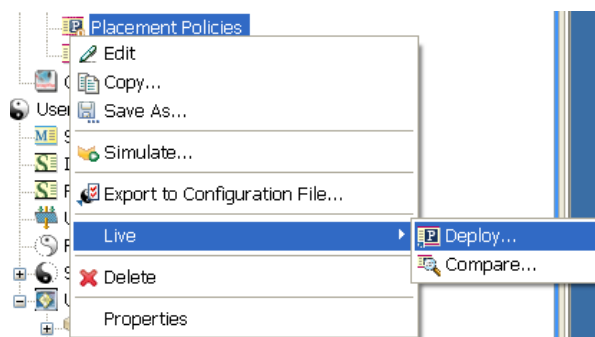
A policy is a collection of rules and arguments that allow you to configure an application so it can send and receive events between itself and an Identity Vault (eDirectory). You use policies to manipulate the data you receive from an Identity Vault or from the application. Each driver performs different tasks and policies tell the driver how to manipulate the data to perform those tasks. For more information about policies, see [Understanding Policies for Identity Manager 3.5 \(http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy/data/bookinfo.html\)](http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy/data/bookinfo.html).

To deploy an Identity Manager-based Policy object (for example, a rule or a style sheet) into a driver or channel (Subscriber or Publisher):

- 1 Click the Outline tab and select a policy under a driver object or a channel object.

Policies can be of the type DirXML[®] Script, Schema Mapping, or XSLT style sheet, and each type has its own icon.

- 2 Right-click a policy object, then select *Live > Deploy*.



An error displays if Designer can't authenticate to the eDirectory tree specified in the Identity Vault, or if you do not have the Deployment DN designated in the Properties tab of the Identity Vault you are deploying to.

The Deployment Summary window shows you the differences between the objects you are deploying and those that already reside in an eDirectory tree. It is the same window format as the Compare feature. For more information on how to use the Compare window, see [Section 19.7, "Using the Compare Feature When Deploying," on page 383](#).

- 3 In the Deployment Summary window, click *Deploy*.
- 4 After the policy deploys, click *OK* to close the Deployment Results window.

19.7 Using the Compare Feature When Deploying

Designer's Compare feature allows you to see differences between driver sets, drivers, channels, and policies that are stored in projects and those that are running in deployed systems. Previous versions of Designer only provided conflict resolution when importing a Driver. While importing, you could select which policies of the driver you wanted to update, but you could not view any differences between existing and new values.

Designer now provides conflict resolution on an object-by-object basis and allows you to view the differences between existing and new values when importing and deploying driver sets, drivers, channels and policies. For example, before deploying a driver object in Designer to a driver object that already exists in the Identity Vault, you can run Compare. Compare shows whether the driver objects are equal (no action is necessary) or unequal. If unequal, you can choose not to reconcile the driver objects, choose to update the driver object in Designer, or choose to update the driver object in eDirectory.

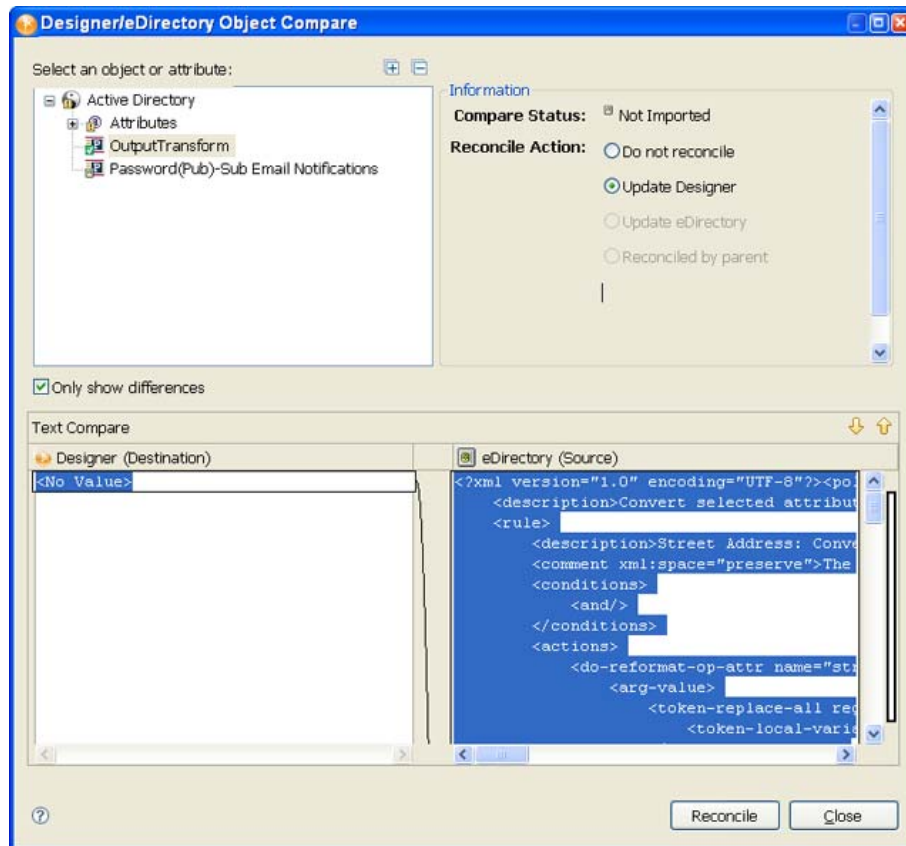
You can run the Compare feature at any time. If you choose to reconcile the differences between drivers objects in Designer and eDirectory while in Compare, you won't need to run Import or Deploy.

- ◆ [Section 19.7.1, "Using Compare when Deploying a Driver Object," on page 384](#)
- ◆ [Section 19.7.2, "Using Compare Before Deploying a Channel Object," on page 387](#)
- ◆ [Section 19.7.3, "Using Compare Before Deploying a Policy," on page 387](#)
- ◆ [Section 19.7.4, "Matching Attributes with Designer Properties," on page 387](#)

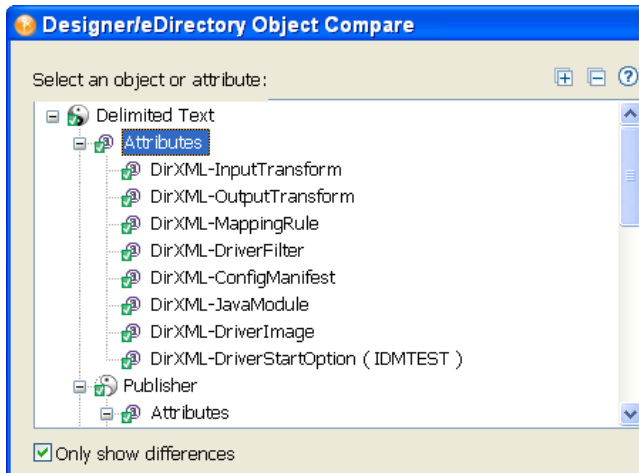
19.7.1 Using Compare when Deploying a Driver Object

Suppose you want to determine if you have deployed all of the changes you have made to a driver object in Designer to the same driver in the Identity Vault.

- 1 Right-click the driver object in either the Modeler view or in the Outline view. Select *Live > Compare* to bring up the Designer/eDirectory Object Compare window.

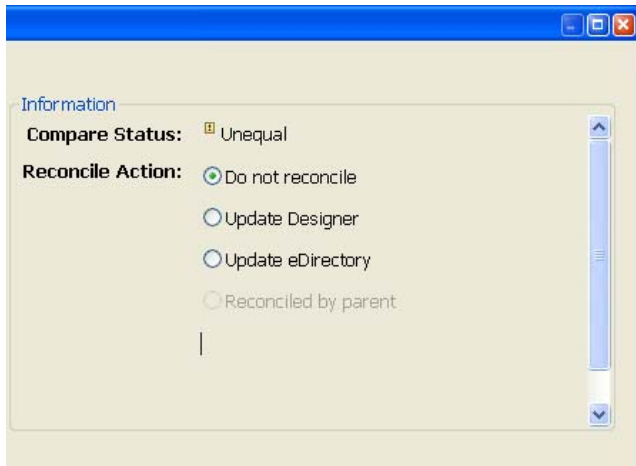


- In the *Select an object or attribute* portion of the window, you see the listed objects and attributes. Select the attributes and child objects to see the actual differences displayed in the *Text Compare* portion of the window.



The plus box at the right side of the *Select an object or attribute* allows you to expand all elements in the parent object, and the minus box collapses all of the elements. The ? icon displays the Summary/Compare dialog box help. Server-specific attributes which are attributes that have a value for each server that is associated with a driver set, are displayed in the *Attributes* list with the server name in parentheses to the right of the attribute name.

- By default, the Compare window only displays values that are different between eDirectory and Designer. To view all of the object values, deselect *Only Show Differences*.
- Check to see the status of the values that are shown. Values that are equal are shown as *Equal* on the *Compare Status* line in the Information portion of the Compare window.



The following table describes what you see in the *Compare Status* line:

Compare Status	Description
Equal	The selected attribute's value or all attributes of the selected object are the same in eDirectory and Designer.

Compare Status	Description
Unequal	The value of the selected attribute, or one or more attributes of the selected object, are different in eDirectory and Designer.
Not Deployed	The selected object or the object containing the selected attribute is not deployed to eDirectory.
Not Imported	The selected object or object containing the selected attribute does not exist in Designer.

The overlay image displayed in the object and attribute tree identifies objects or attributes that need reconciliation.

- Under the Information portion of the Compare window, select how you want to reconcile the differences between the Source and Destination.

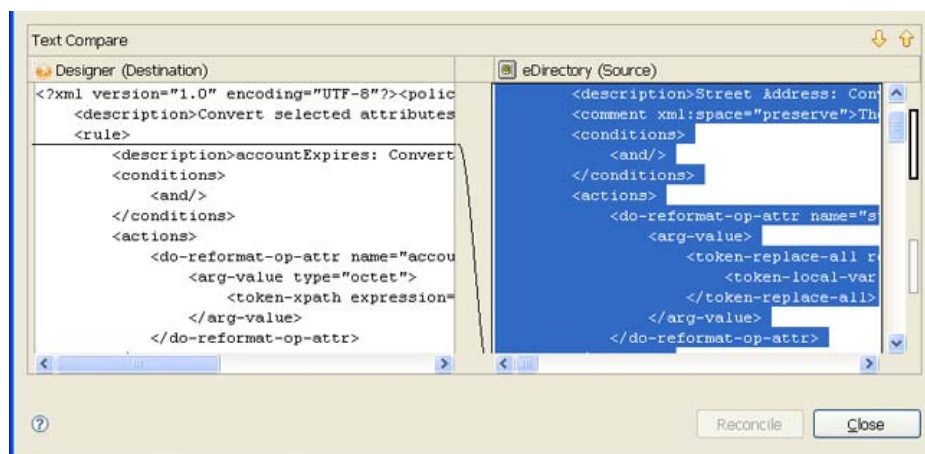
If *Compare Status* shows *Unequal*, you have three choices:

- ◆ To do nothing, keep the default value of *Do Not Reconcile*.
- ◆ To update the driver in Designer so that it contains the same information as the driver in eDirectory, select *Update Designer*.
- ◆ To update the driver in eDirectory to reflect the changes you have just made to the driver in Designer, select *Update eDirectory*.

The green check box in the bottom corner of the icons shows all of the child objects that are being reconciled with the parent object. If you select the parent object to perform the update, then all of the child objects under the parent reflect that choice and you see the *Reconciled By Parent* button selected. If you do not choose a parent object, you can reconcile each child object individually. You can also see a small Designer icon and an eDirectory icon, showing how objects are being reconciled.

- Check to see the *Text Compare* values.

The *Text Compare* values displayed in the bottom portion of the Designer/eDirectory Object Compare window vary, depending on the object being compared. For instance, Compare shows changes to policies or XML data. The Text Compare dialog box uses the Eclipse Compare editor to compare attributes that contain XML data, such as policy data, driver filters, or configuration data. The differences in the code are highlighted in blue.



- 7 After you view the differences, click *Reconcile* to perform the reconciliation actions for each object in the tree, or click *Close* to close the Designer/eDirectory Object Compare window.

19.7.2 Using Compare Before Deploying a Channel Object

Suppose you want to deploy a channel object from the Identity Vault and the same channel already exists in Designer. You can compare the two channels to see similarities and differences.

- 1 Right-click the channel object in the Outline view.
- 2 Select *Live > Compare* to bring up the Designer/eDirectory Object Compare window.
All Compare windows behave the same as described under the [Section 19.7.1, “Using Compare when Deploying a Driver Object,” on page 384.](#)

19.7.3 Using Compare Before Deploying a Policy

Suppose you want to deploy a policy object from the Identity Vault and the same policy already exists in Designer. You can compare the two policies to see similarities and differences

- 1 Right-click the policy object in the Outline view.
- 2 Select *Live > Compare* to bring up the Designer/eDirectory Object Compare window.
All Compare windows behave the same as described under the [Section 19.7.1, “Using Compare when Deploying a Driver Object,” on page 384.](#)

19.7.4 Matching Attributes with Designer Properties

The attributes of the object are displayed in the single select attribute list. Selecting an attribute displays its value below the attribute list with the Designer value on the left and the eDirectory value on the right. The name displayed in the list is the eDirectory attribute name.

Three tables map the eDirectory attribute to the Designer property page or control, where you can change or set the attribute (you can't make changes inside the Compare window). [Table 15-1 on page 295](#) shows Driver Set eDirectory attributes, [Table 15-2 on page 296](#) shows Driver eDirectory attributes, and [Table 15-3 on page 297](#) shows Channel eDirectory attributes.

19.8 Troubleshooting Deployed Objects

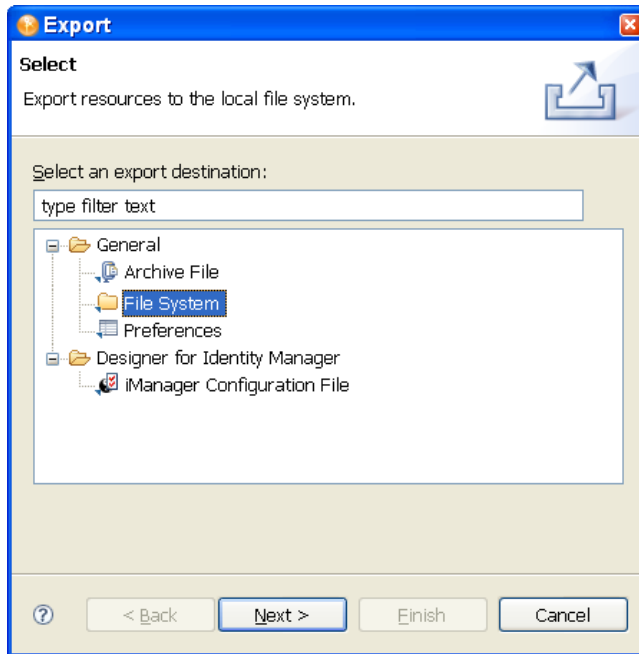
For information on troubleshooting deployed objects, see [Section 24.4, “Deploying Identity Manager Objects,” on page 495.](#)

19.9 Exporting a Project

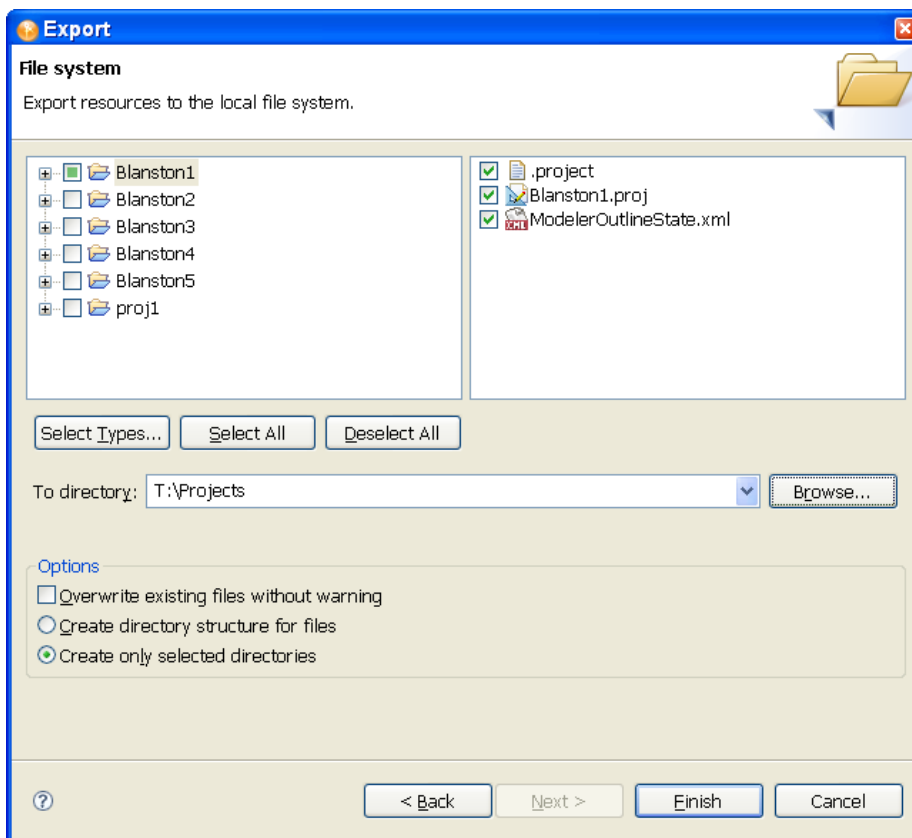
Designer's export feature allows you to export Projects and Driver Configuration files to a local, removable, or network directory.

- 1 Click *File > Export*.

- From the Export window, you can export an existing Identity Manager Project to the File System or to an Archive File. Select *General > File System*, then click *Next*.



- In the Export File System window, select the projects you want to export.



- 4 Click *Select All* to select all projects in Designer's `\workspace` directory (for Windows, the default location is `C:\Documents and Settings\user's_login_name\workspace`).

or

click *Deselect All* to clear the selections. You can then select the projects you want to export.

IMPORTANT: You must select all items relating to a project if a subsequent import of that project is to work.

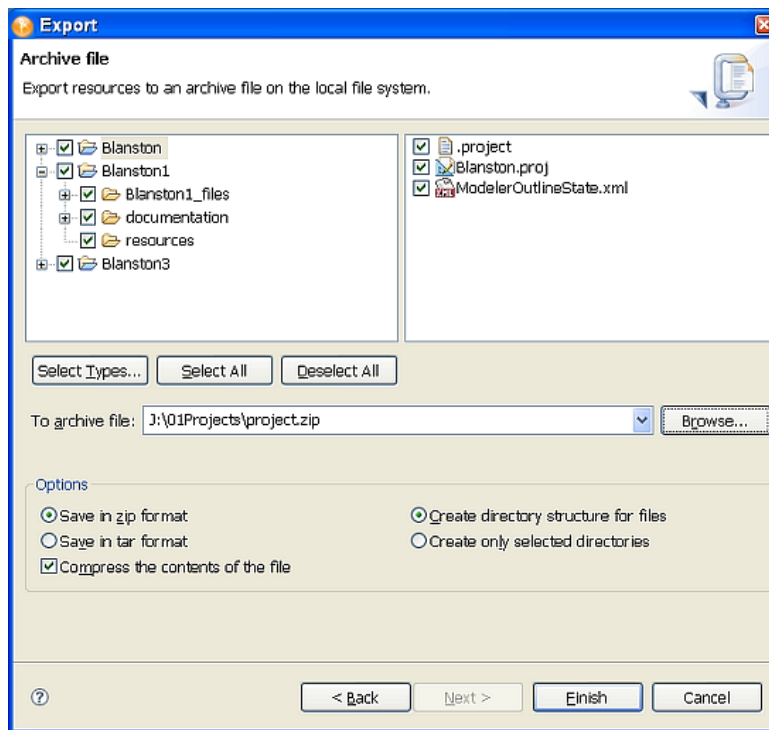
The *Select Types* button is a feature of Eclipse (the platform that Designer is running on) and is not a Designer feature.

You can also browse to the directory location where you want to select the resources.

- 5 After you designate the directory to which to export the projects, click *Create directory structure for files*, then click *Finish*.
- 6 In the Save Resources window, select the projects you want to save, then click *OK*.

You can also export projects to an archive file:

- 1 Click *File > Export*.
- 2 Select the *General > Archive File* option in the Export window, then click *Next*.



- 3 Select the projects you want to archive.
- 4 Designate where you want the archive file saved.
- 5 Select the archive format (zip or tar).
- 6 Select whether you want to compress the contents of the file and whether to keep the directory structure for files (recommended), then click *Finish*.

19.10 Exporting To a File

You can use the export feature to export everything you create in Designer, from projects containing all Identity Vaults and their driver sets down to a single policy. If you export a driver configuration file that contains only a policy, Designer creates the parent containment objects, such as a channel, a driver, or a driver set, as part of the exported policy object. Such parent containment objects do not contain attributes; they are only the framework of the channel, driver, or driver set.

The exported .xml files are compatible with those used by the iManager driver configuration file plug-ins for Identity Manager 2.0.2 and later. This allows you to export configuration files from Designer and import those files through iManager or through Designer's import feature.

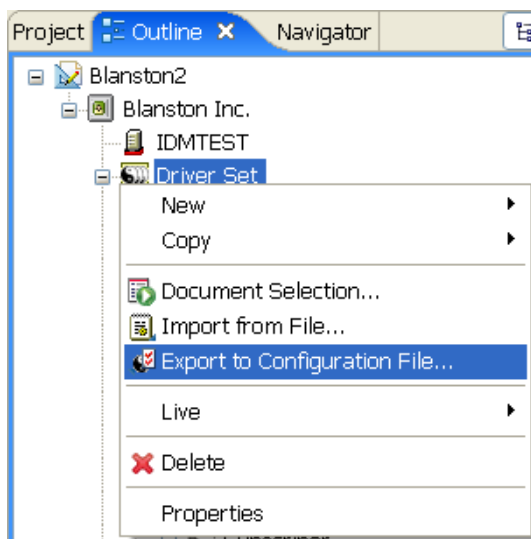
You can export a driver configuration to a file from a number of places, including:

- ◆ [Section 19.10.1, “The Export Context Menu,” on page 390](#)
- ◆ [Section 19.10.2, “Exporting Configuration Files from the Modeler View,” on page 391](#)
- ◆ [Section 19.10.3, “Exporting Configuration Files from the Outline View,” on page 391](#)

19.10.1 The Export Context Menu

To export all driver sets, drivers, channels, and policies found in the Modeler view:

- 1 Right-click the driver set in the Outline View.
- 2 Select *Export to Configuration File*.



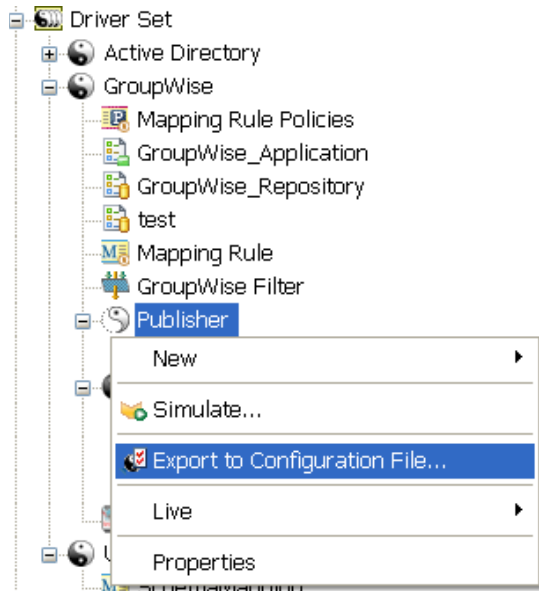
Designer takes the name of the driver sets and uses that name for the .xml file.

- 3 For future reference, name each driver set to denote that it is a driver set and denote the Identity Vault it comes from. You can also add a date to the name. Click *Save*.
- 4 To close the Export Configuration Results window, click *OK*.

19.10.2 Exporting Configuration Files from the Modeler View

To export a driver set configuration file:

- 1 Double-click the *System Model* icon under a project name in the Project view to open the project model in the Modeler view.
- 2 Right-click the Driver Set object inside an Identity Vault icon, then select *Export to Configuration File*.



- 3 In the Export Driver Configuration window, select a filename and location to use in future references. You can also add a date to the filename if you save a lot of driver iterations.

By default, Designer uses the name of the driver or driver set corresponding to the object selected. If you right-click an Identity Vault or Driver Set object, you see the Driver Set name in the File Name entry. If you have more than one Driver Set object in the Identity Vault, you see the Export Driver Configuration window with the name of that driver set in the File Name entry for each Driver Set object.

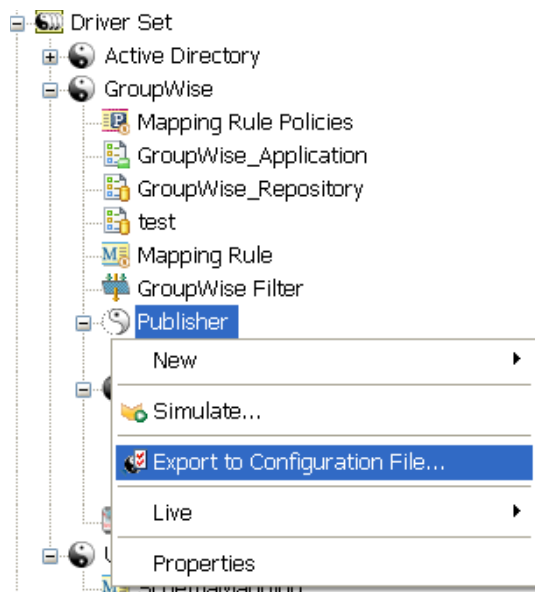
- 4 Select the directory where you want to store the file, then click *Save*.

19.10.3 Exporting Configuration Files from the Outline View

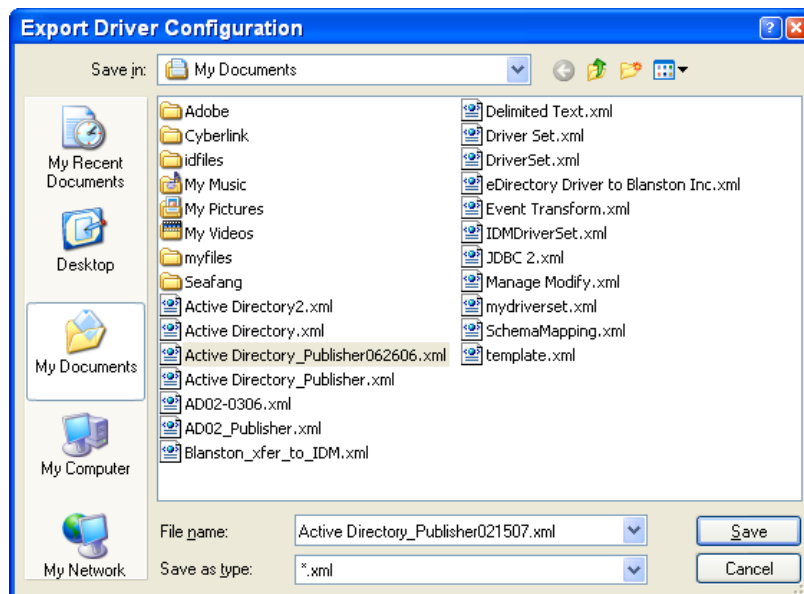
You can use the Outline View to save driver sets, drivers, channels, and policy configuration files to local, removable, or network directories. The following procedure documents steps for exporting channels and policies.

- 1 Double-click the *System Model* icon under a project name in the Project view. This brings up the project in the Modeler view.
- 2 Click the *Outline* tab.

- 3 Right-click a channel object under a driver object, then select *Export to Configuration File*.



- 4 From the Export Driver Configuration window, select a filename and location to use in future references. You can also add a date if you are backing up multiple iterations of the file.

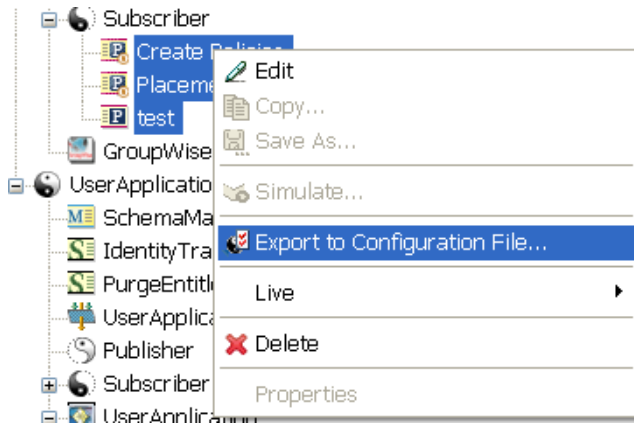


By default, Designer uses the name of the driver or object corresponding to the object selected. You might also need to designate that it is the Publisher channel of an Active Directory driver, along with the date when you saved the file.

- 5 Click *Save*.
- 6 In the Export Configuration Results window, click *OK*.

To export one or more policies from a driver or channel:

- 1 From the Outline view, right-click a Policy object and select *Export to Configuration File*.



You can also use the Ctrl key to select more than one policy, then right-click them as a group and select *Export Policy to Configuration File*.

- 2 From the Export Driver Configuration window, select a filename and location to use for future reference. You can also add a date if you are backing up multiple iterations of the file.
- 3 Click *Save* for each policy selected; each policy is saved to its own .xml file. By default, Designer uses the name of the policy or rule selected.
- 4 In the Export Configuration Results window, click *OK*.

This section provides an overview of the features of the Novell XML Editor. Topics include:

- ◆ [Section 20.1, “About the Novell XML Editor,” on page 395](#)
- ◆ [Section 20.2, “Using the Source Editor,” on page 401](#)
- ◆ [Section 20.3, “Using the Tree Editor,” on page 402](#)
- ◆ [Section 20.4, “Attaching Schema or DTD,” on page 403](#)
- ◆ [Section 20.5, “Setting XML Editor Preferences,” on page 405](#)

20.1 About the Novell XML Editor

The Novell XML Editor lets you create, edit, and validate XML files. You can edit XML files in either the Source or Tree editor. You can customize the certain behaviors, such as code completion, in the Preferences tab.

The Novell XML editor is built on the Web Standard Tools (WST) project architecture.

- ◆ [Section 20.1.1, “Creating XML Files,” on page 395](#)
- ◆ [Section 20.1.2, “Validating Files,” on page 400](#)
- ◆ [Section 20.1.3, “Outline View,” on page 400](#)
- ◆ [Section 20.1.4, “XPath Navigator,” on page 400](#)

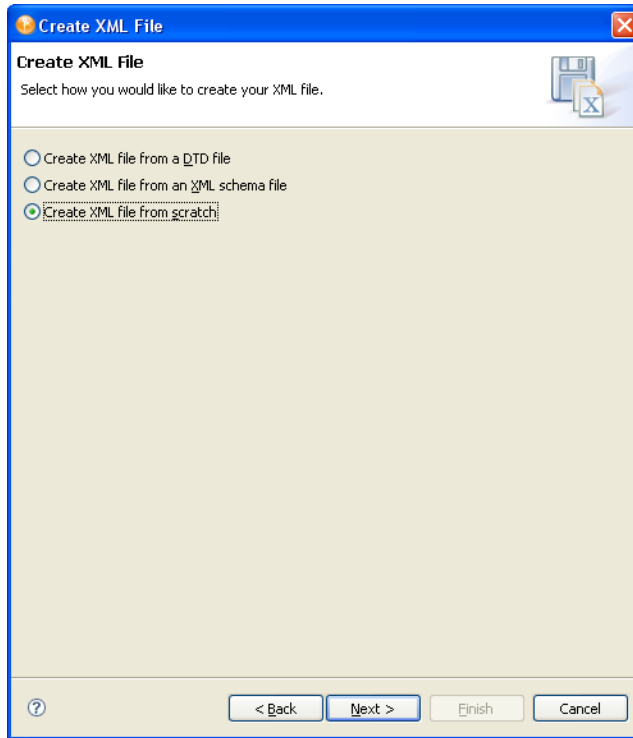
20.1.1 Creating XML Files

You use the *New XML File Wizard* to create new XML Files. The wizard can create an empty XML file or a generated XML file based on an XML schema or DTD. Generated files contain skeleton XML data that is generated based on a given root element and an XML Schema or DTD.

To launch the New XML File Wizard:

- 1 Select *File > New > More Projects*.

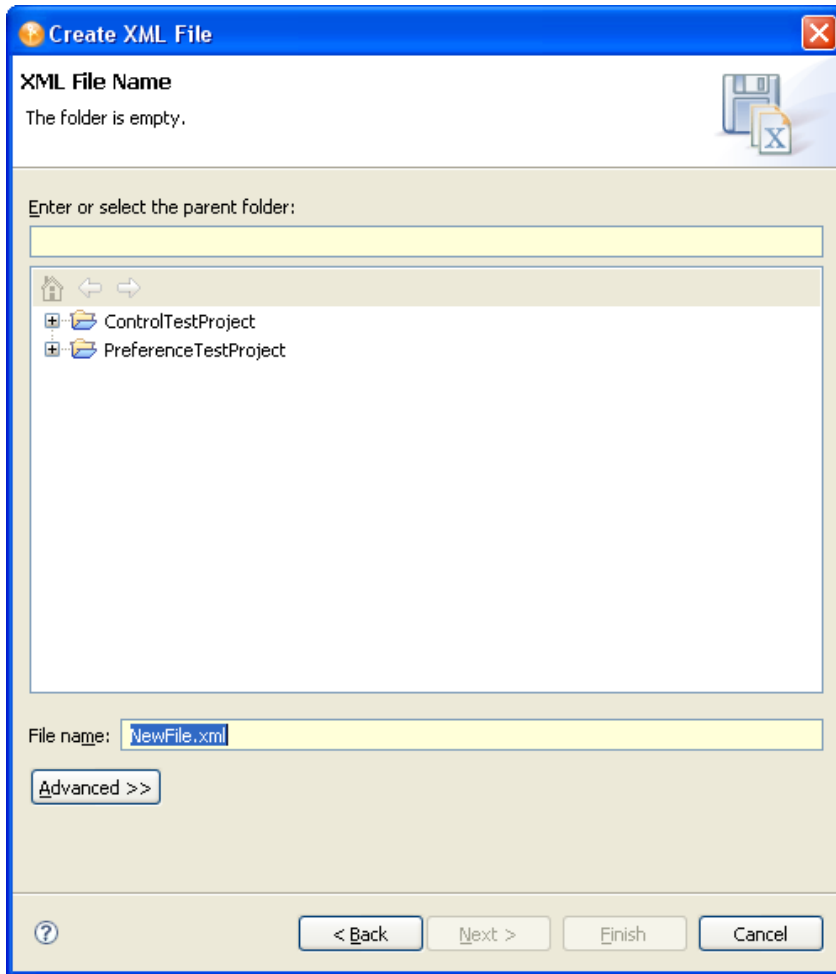
2 Open the XML Folder, select *XML* then select *Next*.



3 Choose one of the source options on which you want to base the new XML file.

Option	Description
Create XML file from a DTD file	Generates an XML document containing a root element and a skeleton based on a DTD that you either import or choose from an existing catalog entry.
Create XML File from an XML schema file	Generates a skeleton XML document containing a root element and skeleton based on a schema that you either import or choose from an existing catalog entry.
Create XML File from scratch	Creates an XML document containing the XML Declaration with the version and encoding attributes set to 1.0 and UTF-8 by default.

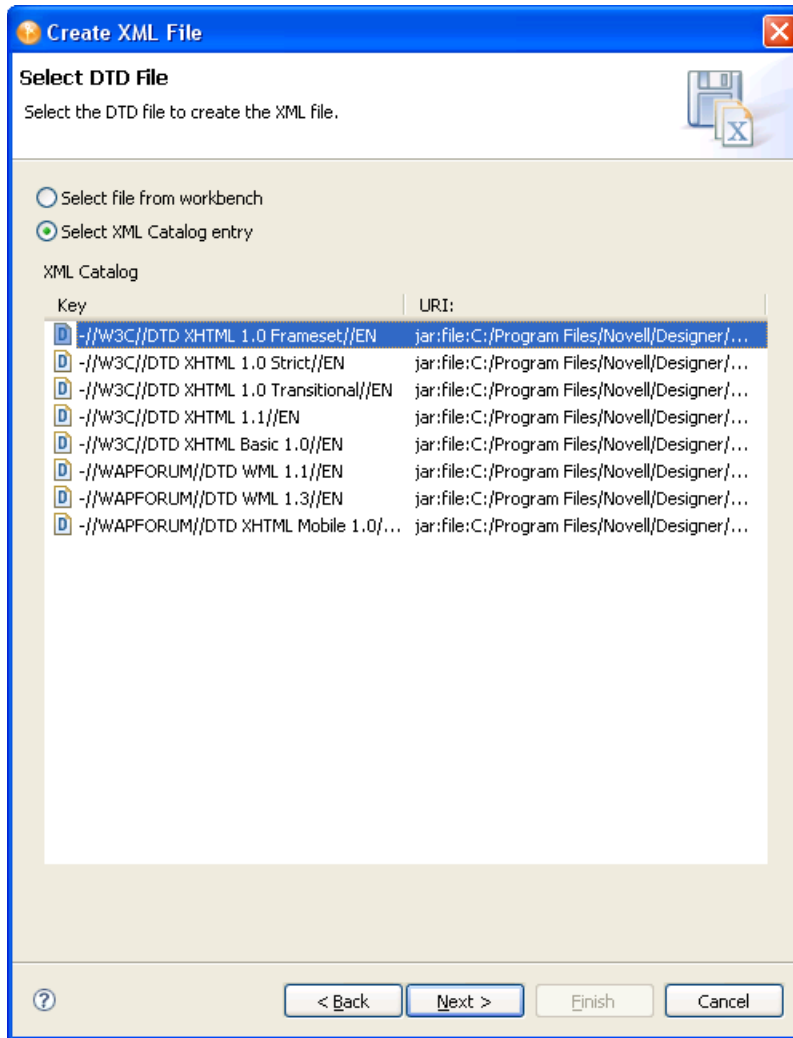
4 Click *Next*. The wizard prompts you to supply the name and location for the new file.



5 Fill in the fields as follows:

Field	Description
Enter or select Parent Folder	Specify where the wizard should create the new file.
File Name	Specify the name of the new file.
Advanced>>	Click this button if you want to specify that the new XML file should link to another file in the file system.

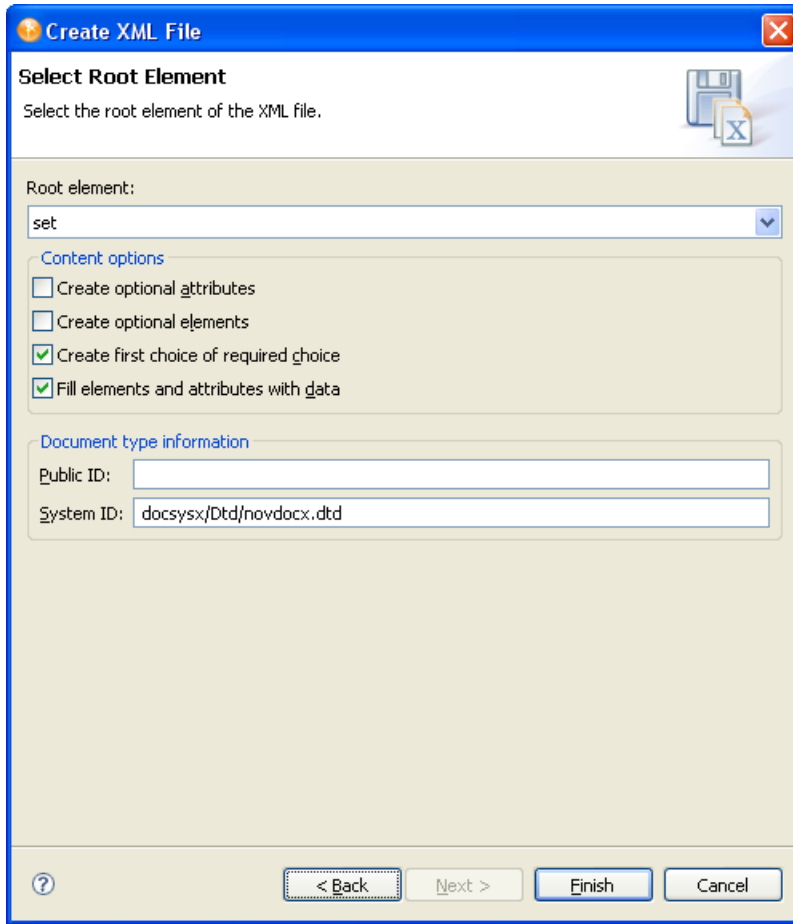
- 6 If you chose to create the file from scratch, click *Finish*, otherwise click *Next*. You are prompted for the catalog entry (shown below) or Schema or DTD.



- 7 Fill in the fields as follows, then click Next:

Field	Description
Select file from workbench	If you choose this option, you must select from a list of DTDs or schemas in your workspace. You can also choose to import a new schema into your project if the schema is not available.
Select XML catalog entry	Choose one of the XML Catalog entries from the list. You can edit this list in the <i>Preferences > Web and XML > XML Catalogs</i> .

8 Click *Next*. You are prompted to specify the Root Element.



9 Fill in the fields as follows:

Field	Description
Root element	Choose or type the new document's root element.
Create optional attributes	Select if you want the wizard to generate optional attributes.
Create optional elements	Select if you want the wizard to generate optional elements.
Create first choice of required choice	Select if you want the skeleton XML to always contain the first choice in a required choice. If this is not selected, no elements are inserted for the choice.
Fill elements and attributes with data	Select if you want the wizard to generate dummy data in the file for elements and attributes. The generated XML inserts the node name as the data of the elements

Field	Description
Public ID	Specify the file's Public ID
System ID	Specify the file's System ID.

10 Click *Finish*.

You can launch the editor by double-clicking an existing XML file.

20.1.2 Validating Files

You can validate your XML files using the right-mouse menu in the Source editor. If any validation errors or warnings occur, they are displayed in the Problems view.

20.1.3 Outline View

The editor provides an Outline view containing a tree that displays the structure of the XML document including its nodes are the elements, attributes, text nodes, comments, and so on from the document.

The Outline tree is closely connected to the Source editor and the Tree editor. When you edit a document in either the Source or Tree editor, the Outline tree updates automatically. If the editing results in a document that isn't well-formed, the structure displayed in the tree might seem odd. But the structure corresponds as closely as possible to the well-formed parts of the document.

Editing or generally moving the cursor in the Source Editor or changing the selection in the Tree editor expands and selects the corresponding node (if possible) in the Outline tree. This makes it possible to easily locate the current place in the document.

In a similar fashion, selecting a node in the Outline tree moves the cursor in the Source editor to the textual position of the node (if the Source editor is active) or change the selection in the Tree editor (if the Tree editor is active). The Outline view provides structural editing capabilities such as inserting and removing nodes.

20.1.4 XPath Navigator

The XPath Navigator view supports context-sensitive editing of XPath expressions and syntax highlighting. It automatically attaches to the currently selected XML editor and uses its Document node as the evaluation context. The namespace context shows all namespaces in scope on its document element.

The view consists of two parts—an editor pane and a results table. When the user types an expression in the editor pane and pauses for 0.5 seconds, the result is shown in the table. If the result is a node list, each row in the table displays an icon for the node type, a short description of the node, and the location of the node in the text (line numbers). Selecting a row in the table selects the text of the corresponding node in the XML editor. However, this is only supported in the Source Editor.

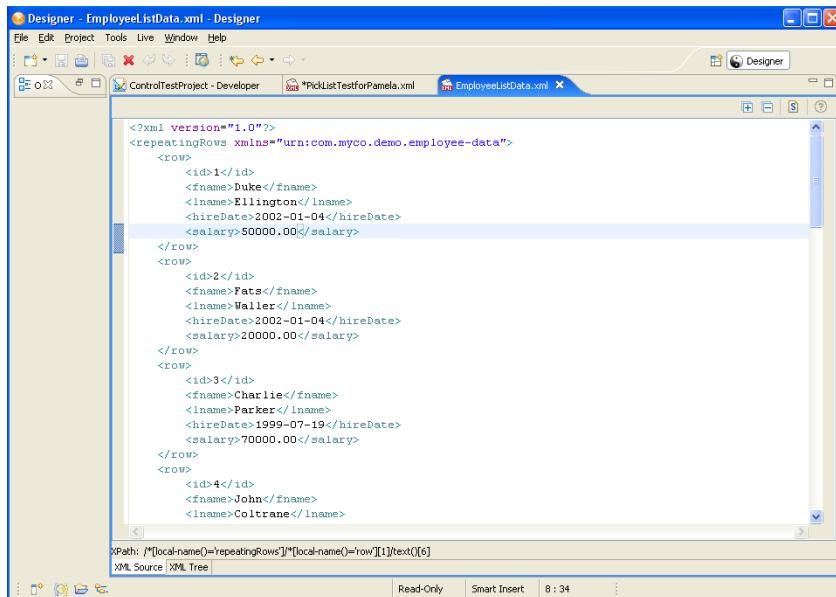
Typing Ctrl-Space, '/', '[' or '(' triggers code-completion—the expression is evaluated up until the cursor location, and insertable elements are shown in a drop-down box.

20.2 Using the Source Editor

The Source editor supports the following features:


- ◆ Syntax highlighting.
- ◆ Context-sensitive code-completion based on DTD and/or XML Schema.
The code completion is based on the existing content of the XML document if no DTD or XML schema is associated with the XML document. When code completion is activated and the XML document contains `<root><a><a></root>` once you type the second `<a>`, the editor suggests that you add `b` as a child of the `a` element.
- ◆ As-you-type validation. If the XML is invalid (for example, the `>` is removed from a tag), the editor indicates the error.
- ◆ General text editing operations such as undo, redo, cut, copy, paste, select all.




Figure 20-1 XML Source Editor



The XML Editor toolbar contains the following:

Table 20-1 XML Editor Toolbar

Button	Description
	<p>Expands all foldings (if code folding is enabled). You can enable/disable code folding in two ways.</p> <ul style="list-style-type: none"> ◆ Selecting <i>Windows > Preferences > General > Editors</i>. Select <i>Structured Editors</i>. Select <i>Enable Code Folding</i>. ◆ In the Source editor, right-mouse in the left ruler. Access the Folding submenu.

Button	Description
	Collapses all folding (if code folding is enabled)
	Attaches a schema. For more information on using this feature, see Section 20.4, "Attaching Schema or DTD," on page 403
	Shows help

The Source editor right-mouse menu contains these options:

Table 20-2 XML Editor Source Editor Right-mouse Menu Options

Menu choice	Description
Revert File	
Cut, Copy, Paste, Undo, Save	Performs the common editor function.
Format	Document: Formats the entire document as specified in the preferences. Active elements: Formats only selected elements.
Clear Validation Errors	Clears reported validation errors from the Problems view.
Validate	Validates the XML document and shows errors in the Problems view.
Preferences	This is the same as setting preferences using the <i>Windows > Preferences</i> option. For more information, see Section 20.5, "Setting XML Editor Preferences," on page 405.

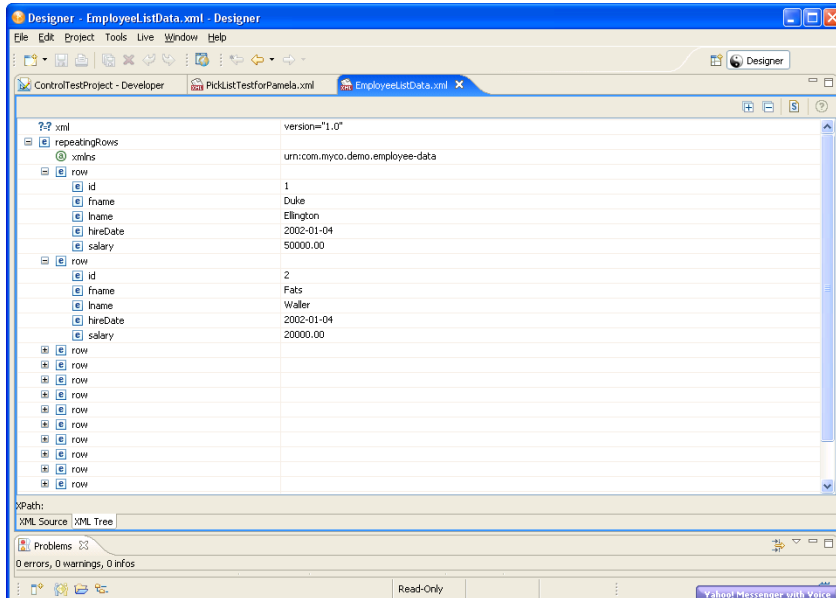
20.3 Using the Tree Editor

The Tree editor supports these features:

- ◆ Direct editing: You can directly edit the text fields, for example, attribute values. It is possible to edit element names, attribute names and values, namespace names and values, text, and comments.
- ◆ Insertion: You can insert new nodes using the Tree editor's right mouse menu. The right mouse menu allows you to insert nodes as children before or after the selected node. If the node is an element, you can insert attributes. The submenus for Add Child, Add After, Add Before contain the node that can be legally added. If no schema or DTD is associated with the document the submenus contain New Attribute or New Element.
- ◆ Deletion: To delete a node, select it and either press the Delete key or use the right mouse menu item Remove.
- ◆ Drag-and-drop inside the tree and between trees.
- ◆ General editing operations such as undo, redo, cut, copy, and paste.





The Tree Editor is a table-tree editor that displays the XML nodes where the value of each node is displayed in a table-cell next to the tree node. It uses the standard WST icons for elements, attributes, text, and so on.

Figure 20-2 XML Editor Tree Editor



The Tree Editor supports these options on the toolbar:

Table 20-3 Tree Editor Toolbar

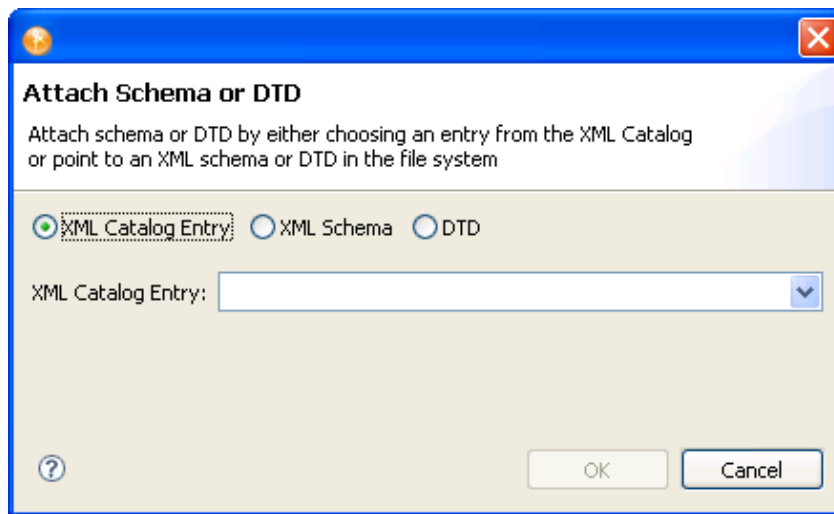
Button	Description
	Expands all node
	Collapses all nodes
	Attaches a schema. For more information on using this features, see Section 20.4, "Attaching Schema or DTD," on page 403.
	Launches help

20.4 Attaching Schema or DTD

Both Source and Tree editors allow you to attach an XML schema or DTD from the toolbar.

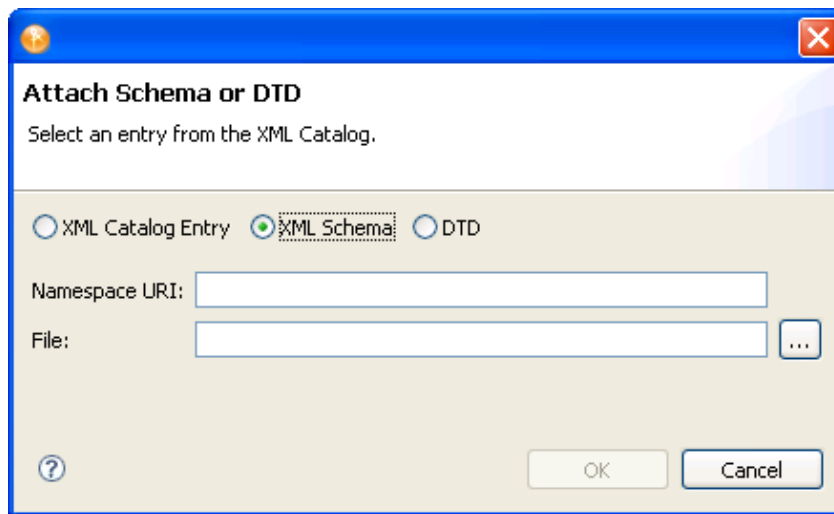
To choose from a list of entries in the XML catalog, choose an entry from *XML Catalog Entry* list.

Figure 20-3 *Attach an XML Catalog*



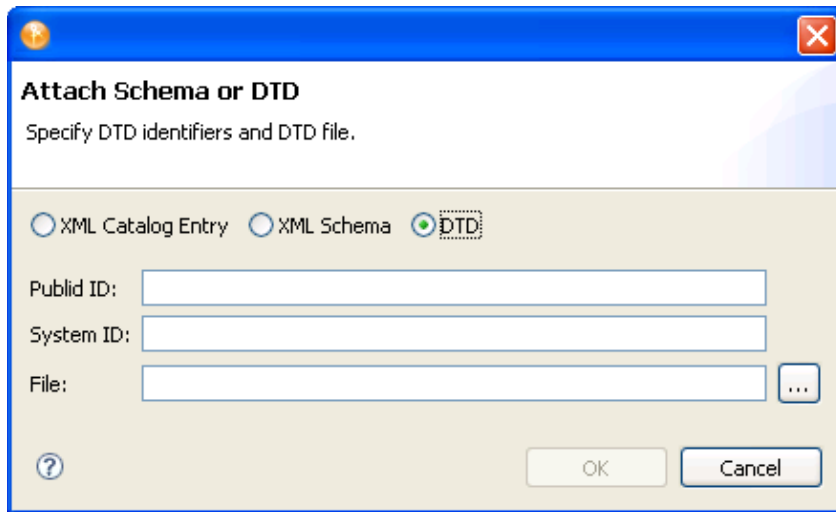
To reference an XML schema on disk, choose XML Schema, then specify the namespace URI and the file's disk location.

Figure 20-4 *Attach an XML Schema*



To reference a DTD on disk, choose DTD, then specify the Public and System IDs and the file's disk location:

Figure 20-5 Attach a DTD



20.5 Setting XML Editor Preferences

You can customize some XML Editor behaviors by setting preferences. You access the preferences page through *Windows > Preferences > Web and XML*. You can learn more about these preferences in the following sections:

- ◆ [Section 23.7.4, “Task Tags,” on page 482](#)
- ◆ [Section 23.7.5, “XML Catalog,” on page 482](#)
- ◆ [Section 23.7.6, “XML Files,” on page 484](#)

- ◆ [Section 21.1, “Converting Earlier Projects,” on page 407](#)
- ◆ [Section 21.2, “Opening a Web Browser,” on page 413](#)
- ◆ [Section 21.3, “Launching iManager,” on page 413](#)
- ◆ [Section 21.4, “Checking Your Projects,” on page 414](#)
- ◆ [Section 21.5, “Managing Directory Objects,” on page 423](#)
- ◆ [Section 21.6, “Configuring TLS for eDir-to-eDir Drivers,” on page 427](#)
- ◆ [Section 21.7, “Viewing DSTrace Log Files,” on page 431](#)
- ◆ [Section 21.8, “Working with Generic Resources,” on page 434](#)
- ◆ [Section 21.9, “Updating Designer,” on page 435](#)

For information on managing workspaces, perspectives, and views, see [Chapter 8, “Workspaces, Perspectives, and Views,” on page 99](#):

21.1 Converting Earlier Projects

The internal format for storing project information in Designer 2.1 differs from the internal format used in earlier releases. Therefore, the Project Converter in Designer 2.1 converts earlier project information to the Designer 2.1 file format.

Conversion is necessary when the following occur to an object in the model:

- ◆ The object has been renamed.
- ◆ The object has been removed.
- ◆ An attribute name on the object was changed.
- ◆ An object’s relationship was changed by one of the following:
 - ◆ The object points to a different object.
 - ◆ A different object points to this object.
 - ◆ The object is contained by a different object.
 - ◆ The object contains a different object.
- ◆ The multiplicity of the relationship with another object has changed from one-to-many or from many-to-one.

Conversion is not necessary under the following conditions:

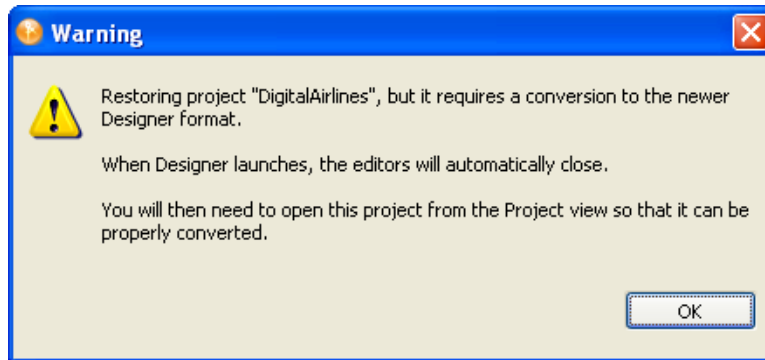
- ◆ A class is added
- ◆ An attribute is added.
- ◆ An additional relationship is added between objects.
- ◆ A class inherits from a different class.
- ◆ [Section 21.1.1, “Converting Projects That Were Open,” on page 408](#)
- ◆ [Section 21.1.2, “Converting Projects That Were Not Open,” on page 408](#)

- ◆ Section 21.1.3, “Running Later Projects on Earlier Designer Versions,” on page 412

21.1.1 Converting Projects That Were Open

If an editor was open when you last closed Designer, and Designer requires a later file format when Designer is launched again, you receive a prompt:

Figure 21-1 Prompt: Newer Format



When you proceed, Designer closes all the editors, even if they belong to other projects, because the projects probably need to be converted. This procedure ensures that you must manually re-open each project. The Project Converter can then back up and convert the project if Designer detects that these tasks are necessary.

When editors are automatically closed, the input focus is set on the Project view.

Scenario: An Editor Is Open. Using Designer 2.0 M4, Nelson created the StarPlus project. It was open in the Modeler when Nelson closed Designer.

Nelson installs Designer 2.1. When Designer 2.1 launches, it closes the editor and then opens it again. Designer detects that the StarPlus project requires a new file format. The Project Converter Wizard backs up the project and converts it to the new Designer 2.1 file format.

21.1.2 Converting Projects That Were Not Open

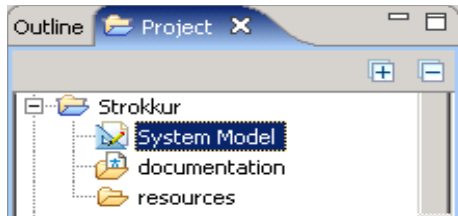
To convert projects that were not open in an editor when Designer was closed, open the project by doing one of the following:

- ◆ Double-click the project in the Project view.
- ◆ Right-click the project in the Project view, then select *Open*.

Although you can open a project in the Navigator view by clicking the project's `.proj` file, Novell recommends that you use the Project view instead. Otherwise, the Navigator view takes you into the raw file system.

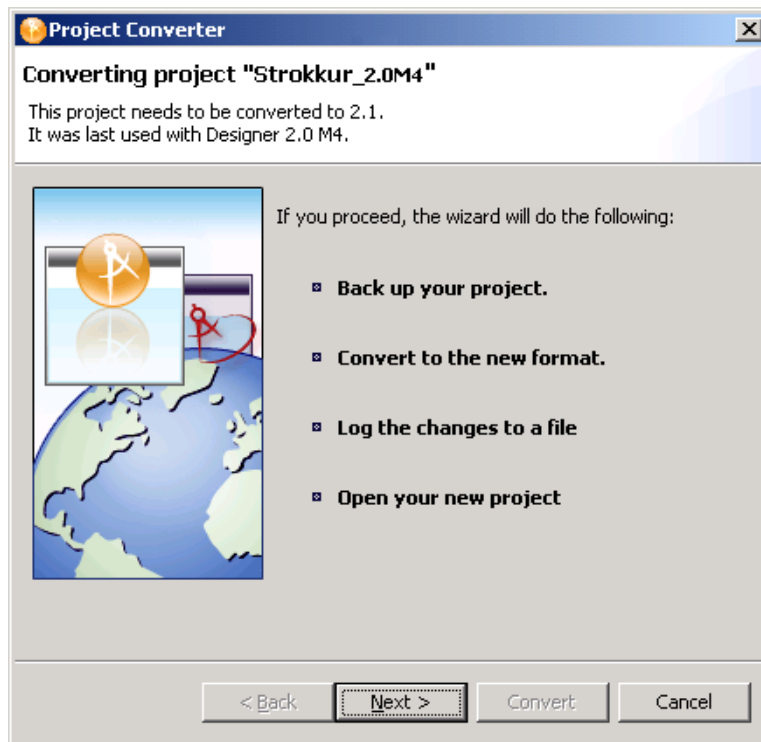
To convert an earlier project:

- 1 In the Project view, expand the project, then double-click *System Model*.



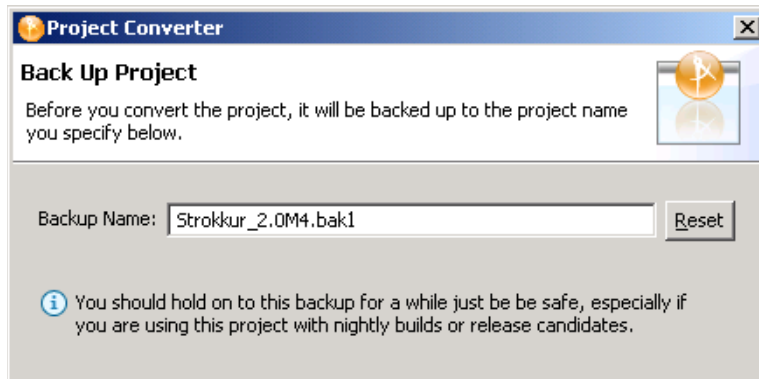
When Designer opens the project file, Designer understands the new format. Designer successfully loads the project into memory because the file format matches the model elements in the newer Designer.

- 2 Review the steps, then click *Next*.



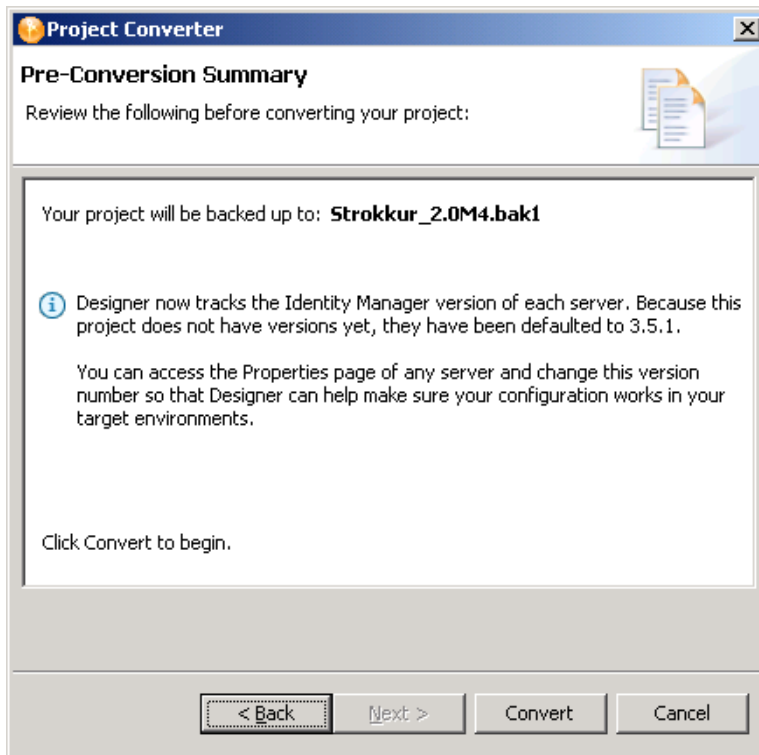
- 3 Name the project, then click *Next*.

The Project Converter backs up your project before converting. You can accept or change the default name.



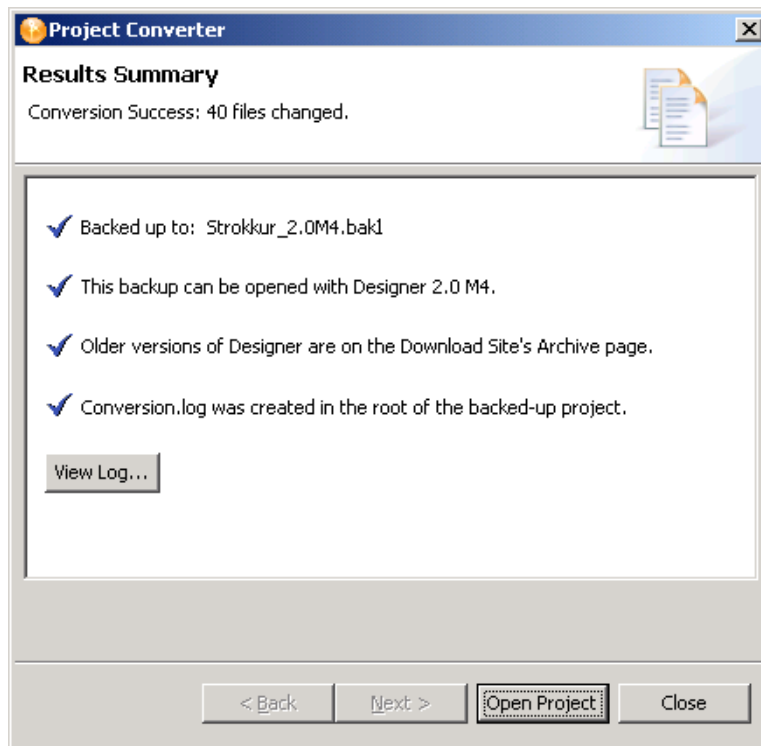
If you edit the name but want to return to the default, click *Reset*.

- 4 Convert the project by clicking *Convert*.

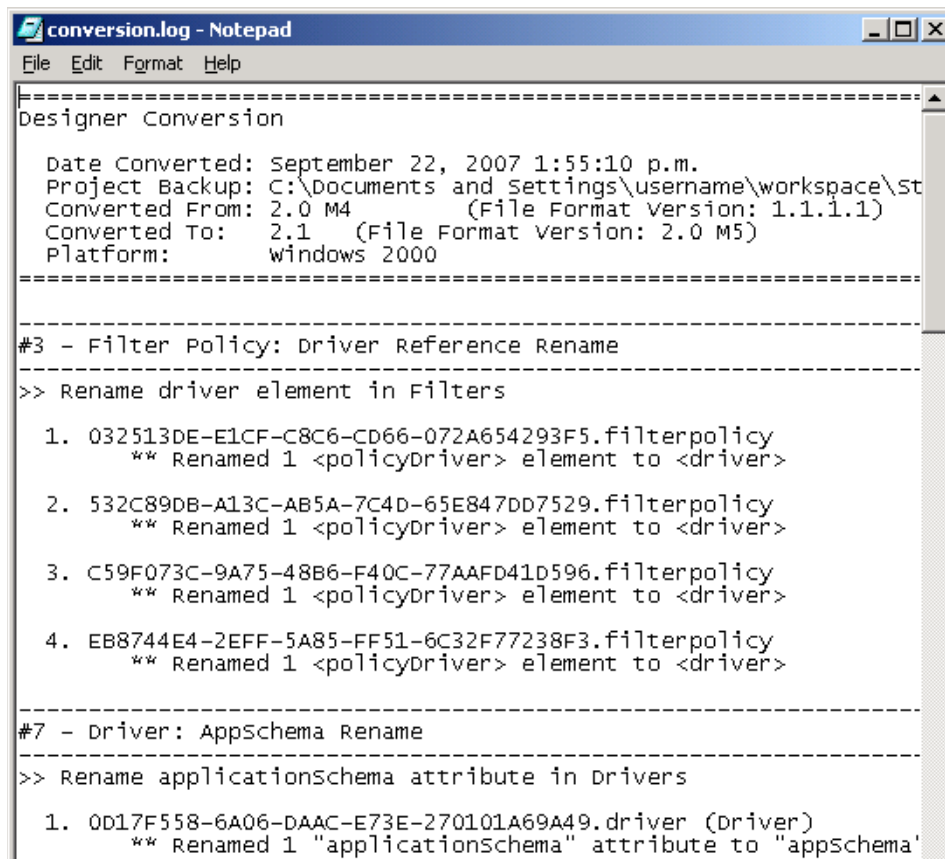


The converter changes, adds, and removes references, attributes, and elements. It might also create new files or delete old ones. It converts the project file to the new, correct file format. A progress bar displays during the backup and conversion. Converting very large projects might take a few seconds.

5 View the conversion log by clicking *View Log*.



The `conversion.log` file is in the project folder in the Workspace directory (for example, `c:\documents and settings\skopai\digitalairlines\conversion.log`).



```
conversion.log - Notepad
File Edit Format Help
=====
Designer Conversion
Date Converted: September 22, 2007 1:55:10 p.m.
Project Backup: C:\Documents and Settings\username\workspace\st
Converted From: 2.0 M4 (File Format Version: 1.1.1.1)
Converted To: 2.1 (File Format Version: 2.0 M5)
Platform: windows 2000
=====
#3 - Filter Policy: Driver Reference Rename
-----
>> Rename driver element in Filters
1. 032513DE-E1CF-C8C6-CD66-072A654293F5.filterpolicy
   ** Renamed 1 <policyDriver> element to <driver>
2. 532C89DB-A13C-AB5A-7C4D-65E847DD7529.filterpolicy
   ** Renamed 1 <policyDriver> element to <driver>
3. C59F073C-9A75-48B6-F40C-77AAFD41D596.filterpolicy
   ** Renamed 1 <policyDriver> element to <driver>
4. EB8744E4-2EFF-5A85-FF51-6C32F77238F3.filterpolicy
   ** Renamed 1 <policyDriver> element to <driver>
-----
#7 - Driver: AppSchema Rename
-----
>> Rename applicationSchema attribute in Drivers
1. 0D17F558-6A06-DAAC-E73E-270101A69A49.driver (Driver)
   ** Renamed 1 "applicationSchema" attribute to "appschema"
```

6 Open the project.

Regardless of the internal format (for example, Designer 1.2 or Designer 2.1), Designer always deploys to the proper format of the target Identity Manager environment.

The converter ensures only forward compatibility. It does not ensure backward compatibility. You can't convert to a newer release of Designer and then convert back if you want to return to an older release. If you need to go back to an earlier format, use the project's backup file.

21.1.3 Running Later Projects on Earlier Designer Versions

Designer 2.0 M5 or later (including Designer 2.1) does not let you load a project created in later Designer versions if the file format has changed between versions.

For example, Designer 2.0 M4 and Designer 2.1 use different formats. If you create a project in 2.1, you cannot open that project in 2.0 M4. Instead, a message informs you that you can't open the project because it requires the later version of Designer.

Even if the version formats are the same, it isn't a good practice to run later projects on earlier versions of Designer. Later versions of Designer have additional bug fixes and features that might make more use of the existing format. Therefore, going back to an earlier Designer version could result in a degraded experience.

21.2 Opening a Web Browser

You can open a Web browser by using Designer. The Web browser button is available from the main toolbar.

When you first launch the browser, you are prompted for a home page. After you enter the URL, it is stored in Preferences.

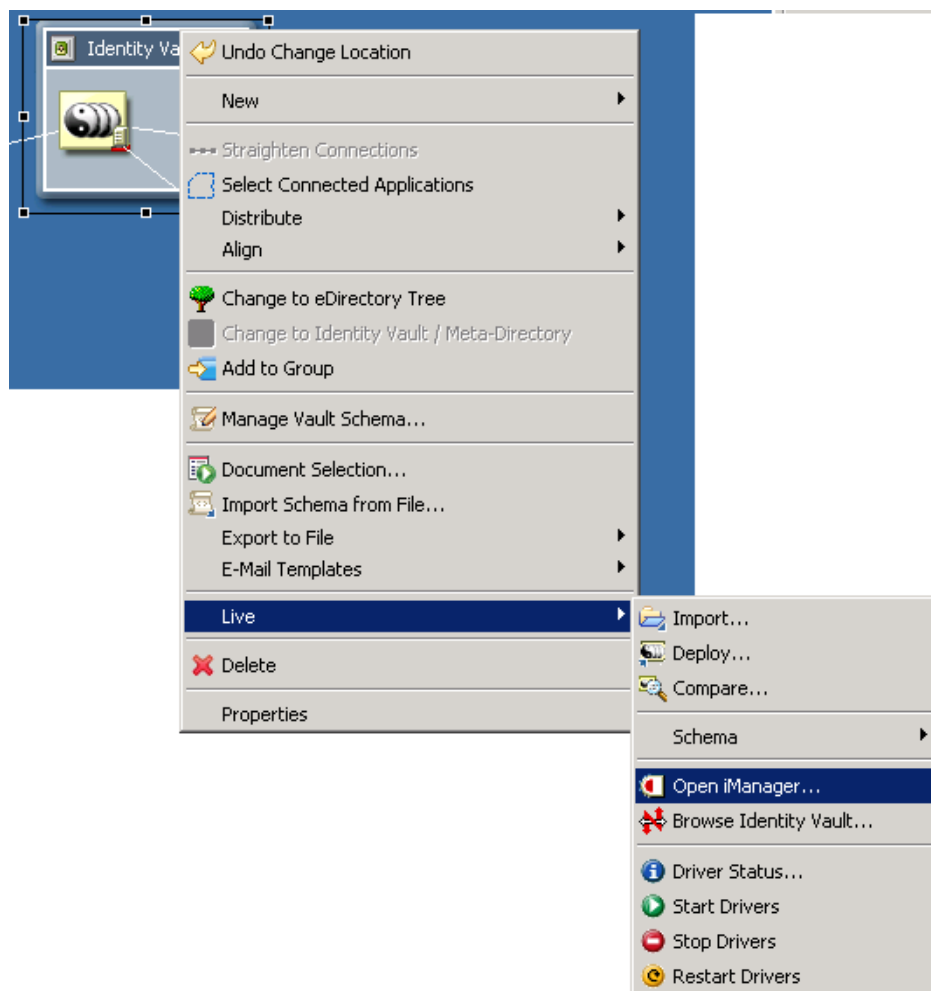
To change the URL:

- 1 Select *Window > Preferences*.
- 2 Select *Designer for IDM*.
- 3 Click the *Browser* tab.
- 4 Type the new URL, then click *OK*.

21.3 Launching iManager

To launch iManager from within Designer:

- 1 Right-click the Identity Vault, then select *Live > Open iManager*.



You can also select *Tools > iManager*.

2 Enter credentials.

Designer stores the credentials in a history for subsequent launches.

3 Click *OK*.

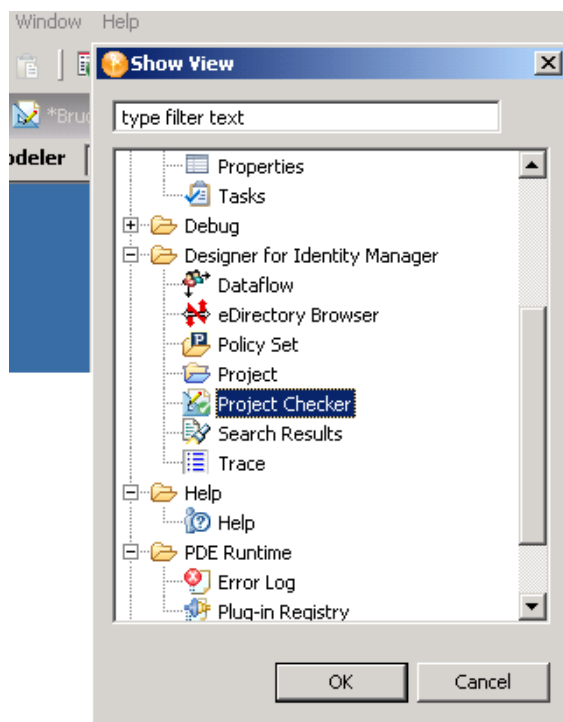
21.4 Checking Your Projects

Designer provides a Project Checker tool to check your project. The project can be checked at any time, but you should run the Project Checker before deploying your project. The Project Checker checks for proper design, contexts, server associations, policies, missing user data, and dependency problems that would cause the deployment of project into the Identity Vault to fail. It only checks the objects in Designer; it does not check the current objects in the Identity Vault.

- ♦ [Section 21.4.1, “How to Check Your Project,” on page 414](#)
- ♦ [Section 21.4.2, “Customizing the Project Checker,” on page 418](#)
- ♦ [Section 21.4.3, “Items That Are Checked,” on page 421](#)

21.4.1 How to Check Your Project

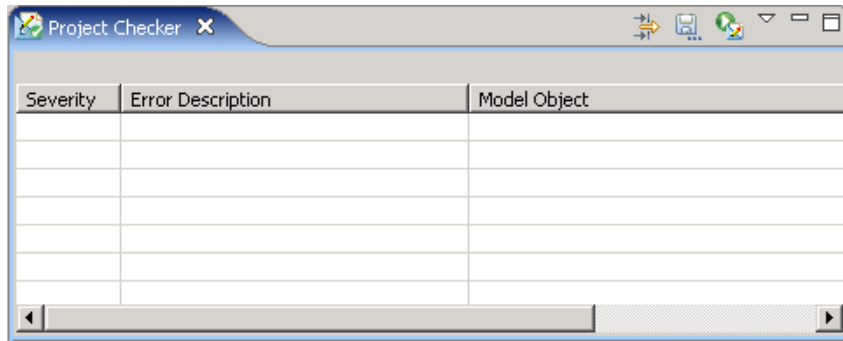
- 1 Select the project in the Modeler or in the Project view.
- 2 Select *Window > Show View > Other*.
- 3 Expand *Designer for Identity Manager*, then double-click *Project Checker*.



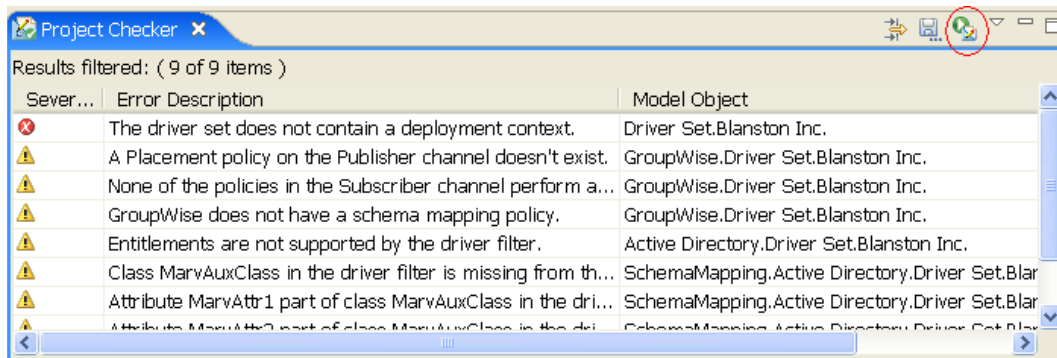
You can also click the Project Checker icon  at the top of Designer to bring up the Project Checker view.

The Project Checker is displayed in a new view.

Figure 21-2 Project Checker



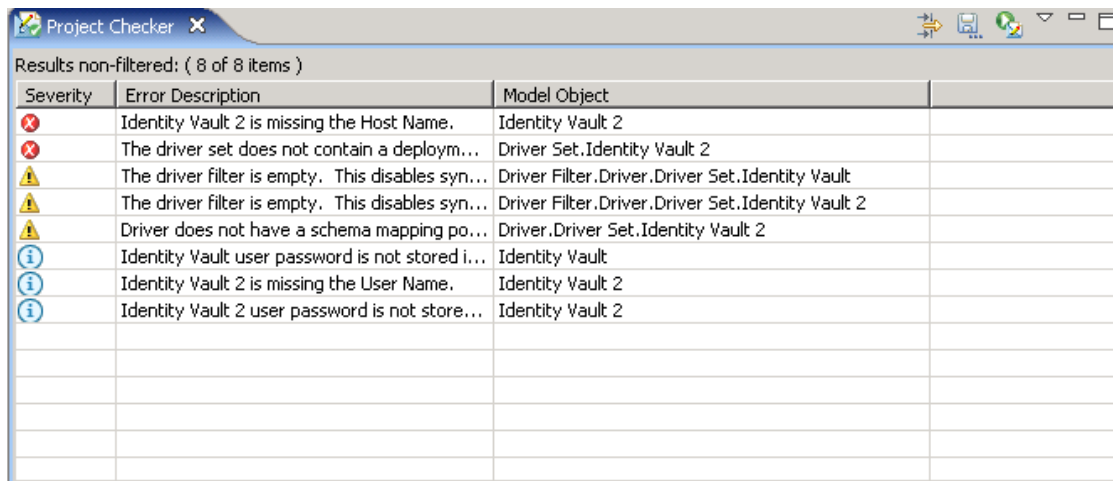
4 Click the *Run the Project Checker* icon.



If you haven't saved the project, Project Checker prompts you to save it.

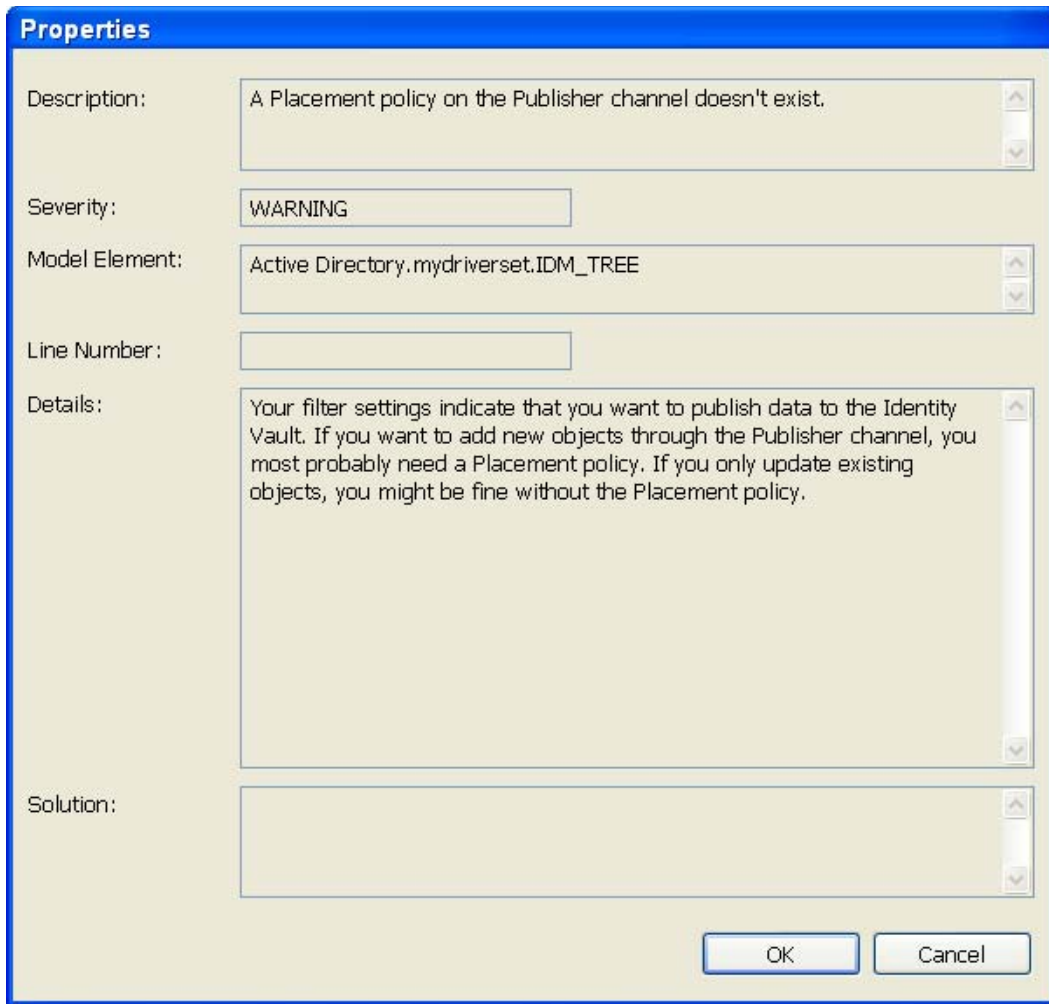
The Project Checker displays versioning conflicts, errors, warnings, and information messages about the project.

Figure 21-3 Errors Listed in the Project Checker View



To see more information, double-click an item.

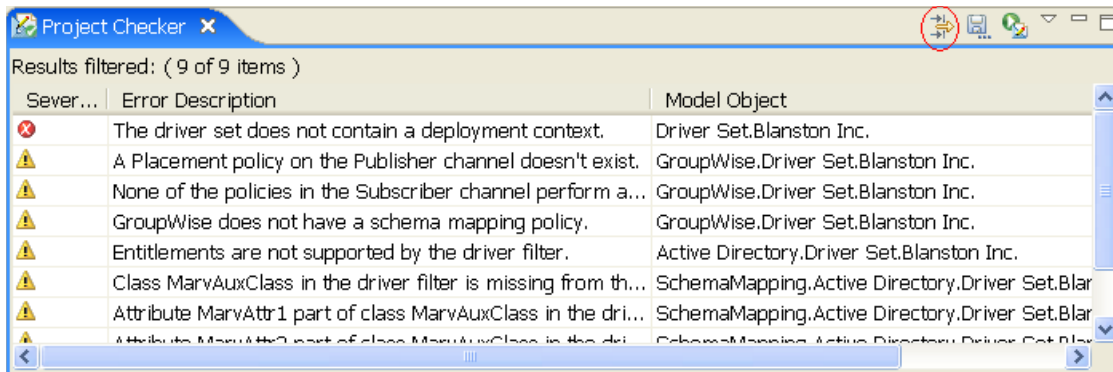
Figure 21-4 Details of an Item



The expanded view lists a description of the message, the severity of the message, and which object caused the error. If there is a line number, it lists the line number where the problem occurred. If there is a known solution for the message, it is also listed.

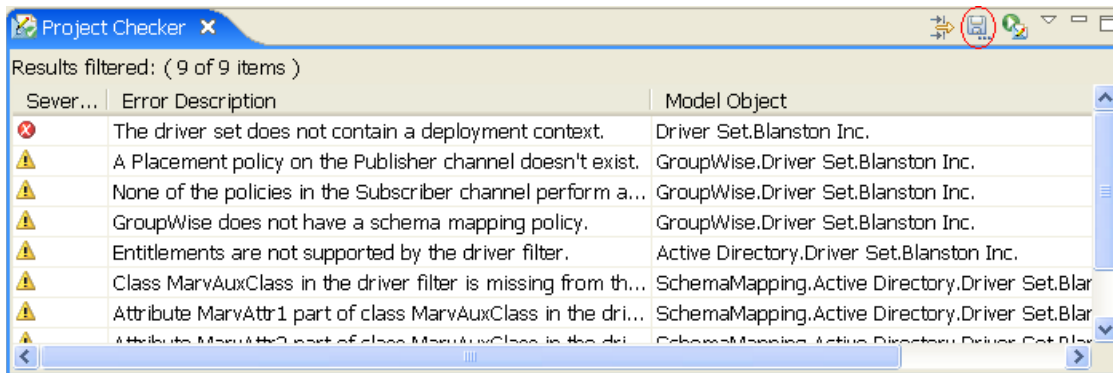
You can use the *Configure Filters* icon to customize which object types are checked. For more information, see [“Customizing the Project Checker” on page 418.](#)

Figure 21-5 The Configure Filters Icon



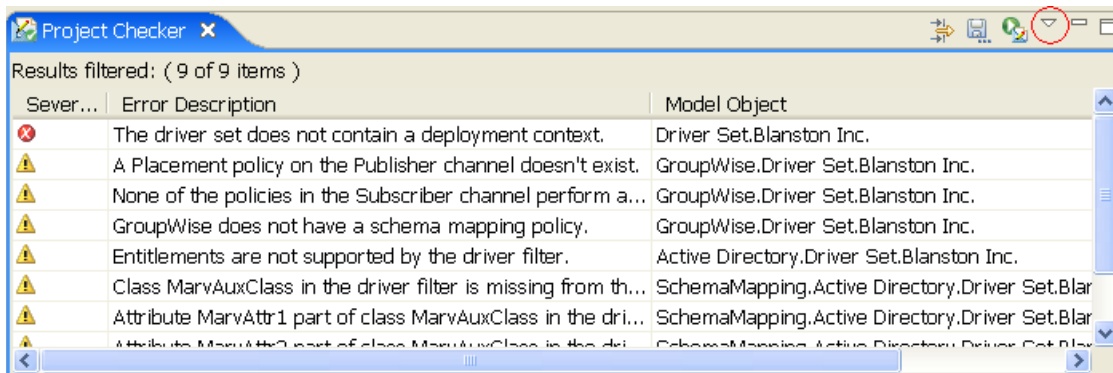
You can save the results to a file. Browse to a location and specify a filename and file type, then click *Save*.

Figure 21-6 Saving the Results of the Project Checker



The Project Checker allows you to view the messages in a hierarchical format or in a flat format. It allows you to limit the object-specific messages that are displayed. Click the *Menu* icon for more choices.

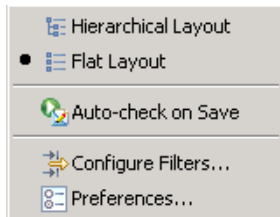
Figure 21-7 Changing the View



The drop-down menu has the following options:

- ◆ **Hierarchical Layout:** Displays the messages under Identity Manager and Provisioning headings.
- ◆ **Flat Layout:** Displays all messages without the hierarchical structure.
- ◆ **Auto-check on Save:** Automatically runs the Project Checker every time you save a project.
- ◆ **Configure Filters:** Allows you to customize which objects are checked.
- ◆ **Preferences:** Limits the number of messages that are visible. The default is 100 items.

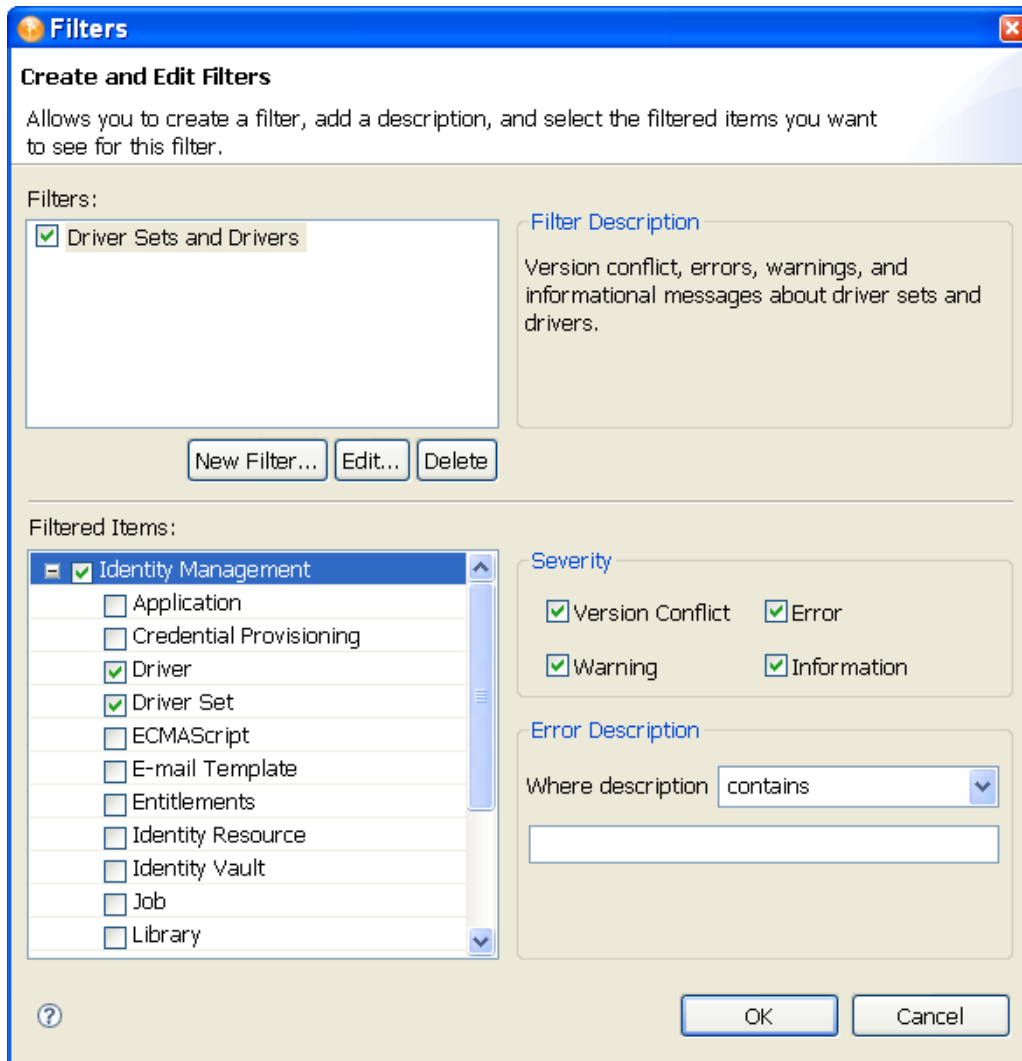
Figure 21-8 Views




21.4.2 Customizing the Project Checker

You can customize the Project Checker by creating and editing filters. The filters allow you to receive messages about the items you want to verify. You can create multiple filters, but only one filter can be used at a time.

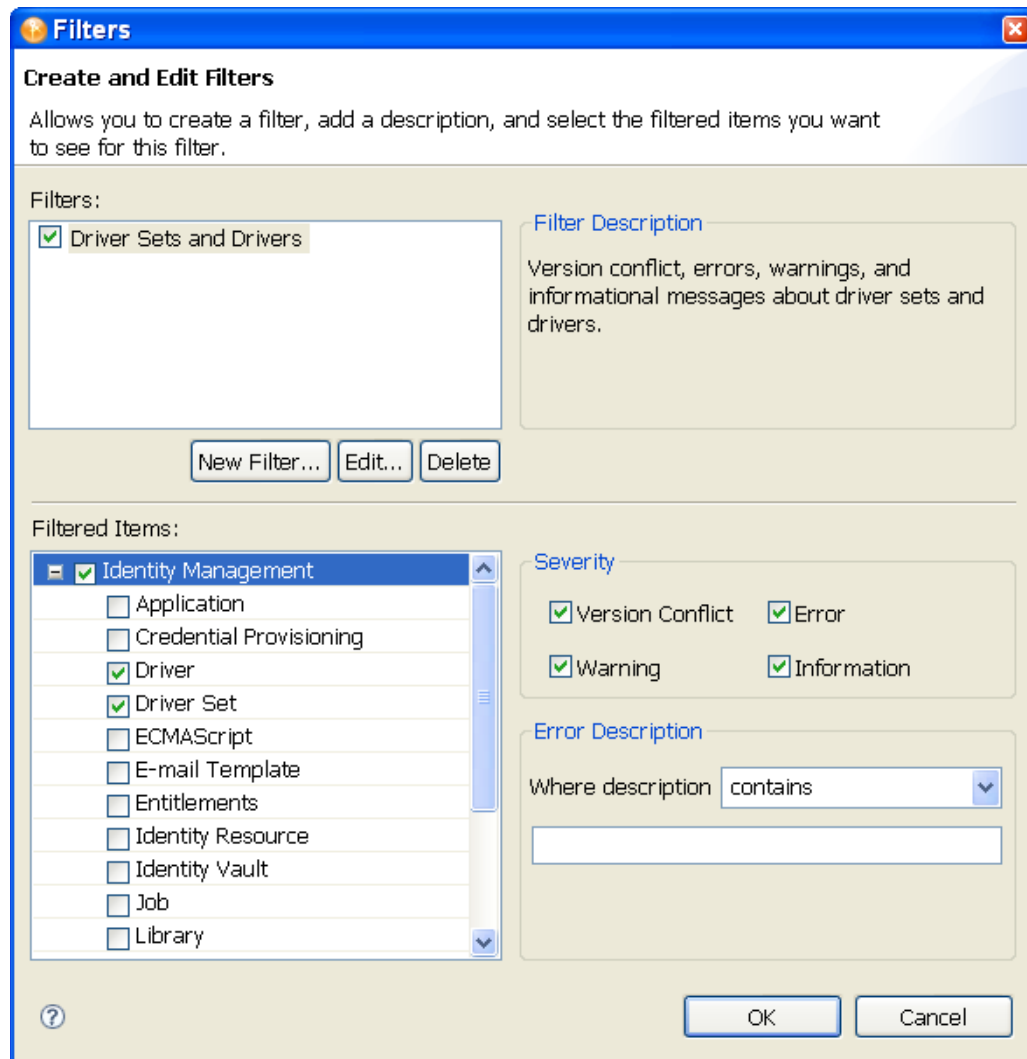
Figure 21-9 Customizing the Project Checker



To create a filter:

- 1 In the Project Checker, click the *Configure Filters* icon .
- 2 Click *New Filter*.
- 3 Specify a name and description for the filter.

You can select which items are checked, what types of messages are returned about the items, and use key words to limit the messages returned. For example, you can search for all messages about the Driver Set and Driver objects that contain the word “attribute”.



4 Click *OK*.

To edit the name and description of the filter:

- 1 Select the filter, then click *Edit*.
- 2 After you have completed the changes, click *OK*.

To delete a filter:

- 1 Select the filter.
- 2 Click *Delete*.

21.4.3 Items That Are Checked

The Project Checker looks at specific items in the project. It checks the items in the User Application as well as the rest of Identity Manager.

The following table describes the specific items that are checked. The list will increase with each release of Designer.

Table 21-1 Identity Manager Items That Are Checked

Item	Description
Driver	<ul style="list-style-type: none">◆ Checks for the presence of a Schema Mapping policy.◆ Checks for an invalid Active Directory container.◆ Checks the trace level setting. If it is set to more than 0, an informational message is displayed.◆ Checks to see if the LoopBack driver is being used instead of the eDirectory™ driver.◆ Verifies that the GUID attribute is set to synchronize on the Subscriber channel.◆ Verifies that the GUID attribute is not set to synchronize on the Publisher channel.◆ Checks the classes on the Publisher and Subscriber channels that are set to <i>Ignore</i> and verifies that the attributes for these classes are not set to <i>Synchronize</i>.◆ Checks for the presence of a filter and makes sure it is not empty.◆ Checks to make sure that the Publisher Placement policy does not contain <i>set operation destination DN</i> or <i>set xml attribute</i> operations.◆ Checks for the presence of a Publisher Placement policy.◆ Checks to make sure that no policy on the Publisher channel contains <i>set operation destination DN</i> or <i>set xml attribute</i> operations.◆ Checks to make sure that the Subscriber Placement policy does not contain <i>set operation destination DN</i> or <i>set xml attribute</i> operations.◆ Checks to see if the Subscriber Placement policy is missing.◆ Checks to make sure that no policy on the Subscriber channel contains <i>set operation destination DN</i> or <i>set xml attribute</i> operations.◆ Checks to make sure that the npsmDistributionPassword attribute and the public-private key pair attributes do not simultaneously exist in the User class.◆ Checks to make sure that the authentication method on the Active Directory driver is set to <i>Negotiate</i> when synchronizing passwords.◆ Checks the filter for invalid data.◆ Checks the driver to see if it is publishing both NDS® and Distribution passwords. If it is, this is an invalid setting.◆ Checks for the presence of the npsmDistributionPassword attribute in the User class in the Filter, if password synchronization is enabled.◆ Checks that the npsmDistributionPassword attribute is set to sync or notify, if password synchronization is enabled.

Item	Description
Driver Set	<ul style="list-style-type: none"> ◆ Checks to make sure that the deployment context for the Driver Set object is set. ◆ Checks to make sure that a server object is associated with the Driver Set object.
E-mail Template	Checks to see if the e-mail notification template is empty.
Entitlements	<ul style="list-style-type: none"> ◆ Checks to see if the driver supports entitlements. ◆ Checks to see if the attribute DirXML-EntitlementRef is added to the Subscriber channel, if there are policies that use entitlements in the driver. The DirXML-EntitlementRef must be set to Notify or Synchronize for the entitlements to work.
ECMAScript	Checks to see that the ECMAScript object can run only on Identity Manager 3.5.
Identity Vault	<ul style="list-style-type: none"> ◆ Checks to see if the username to authenticate to the Identity Vault is missing. ◆ Checks to see if the hostname for the Identity Vault server is missing. ◆ Checks to see that the password for the user is not stored in the project.
Job	Checks to see that the job object can run only on Identity Manager 3.5.
Library	Checks to see that the library object can run only on Identity Manager 3.5.
Mapping Table	Checks to see that the mapping table object can run only on Identity Manager 3.5.
Policy	<ul style="list-style-type: none"> ◆ If there are global configuration values in the policy, it checks to make sure they exist on the Driver or Driver Set object. ◆ Checks to see if local variables are defined before they are used. ◆ Validates the policy against the DTD.
Schema	<ul style="list-style-type: none"> ◆ Checks to see if the class is missing from the schema. ◆ Checks to see if the attribute is missing from the schema. ◆ Checks to see if the attribute for the class is missing from the schema.

Table 21-2 Provisioning Items That Are Checked

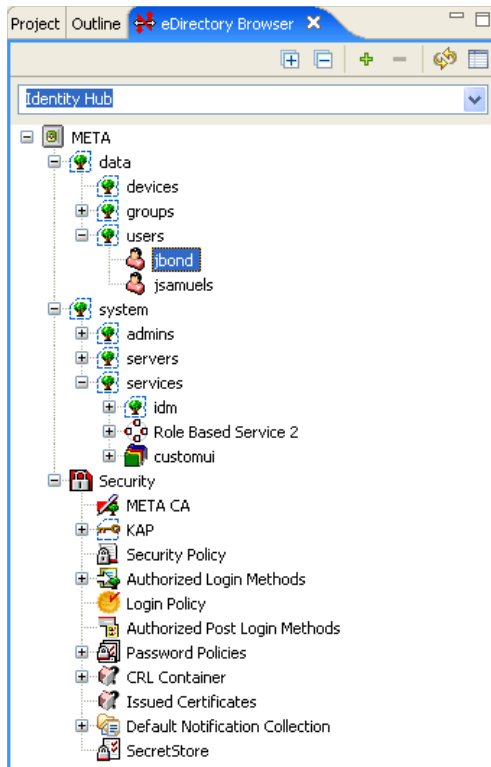
Item	Description
Configuration	Verifies that the XML is well-formed and complies with the schema that defines the elements needed for entities, attributes, lists, relationships, and so on.
Entity	<ul style="list-style-type: none"> ◆ Checks every entity to ensure that references to other entities and global lists are valid. ◆ Ensures that every entity has at least one attribute defined.
List	Ensures that every local and global list contains at least one item.
Org Chart Relationship	Verifies that the entities and attributes of a relationship have been deployed.
Provisioning Request Definition	Verifies that a workflow follows rules for activities and flow paths.

21.5 Managing Directory Objects

Sometimes it is necessary to locate or modify objects during your project development. Rather than using a separate management interface, you can use the eDirectory Object Manager to browse to and edit attributes of objects in the following locations:

- ♦ The Identity Vault
- ♦ Other eDirectory trees

Figure 21-10 Sample eDirectory Object Manager view



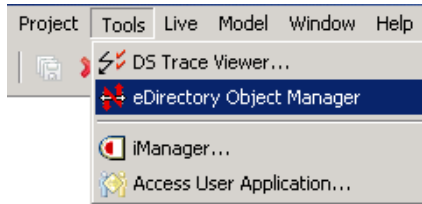
To launch the eDirectory Object Manager, use the tool-based method or the task-based method. The method you use is largely a matter of preference and the target directory that you will browse.

- ♦ [Section 21.5.1, “Tool-Based Method,” on page 424](#)
- ♦ [Section 21.5.2, “Task-Based Method,” on page 425](#)
- ♦ [Section 21.5.3, “Browse, View, or Modify Object Attributes,” on page 425](#)
- ♦ [Section 21.5.4, “Advanced Editing,” on page 427](#)

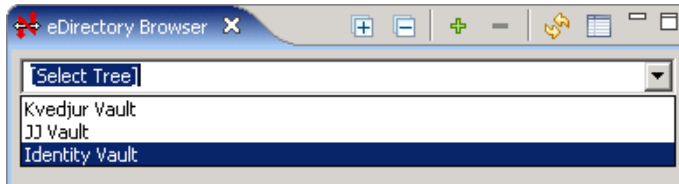
21.5.1 Tool-Based Method

To access an eDirectory tree other than the Identity Vault, or if there is no Identity Vault defined in your current project, use the tool-based method. You can always launch the eDirectory Object Manager from the Tools menu, even when an Identity Vault isn't selected in the Modeler.

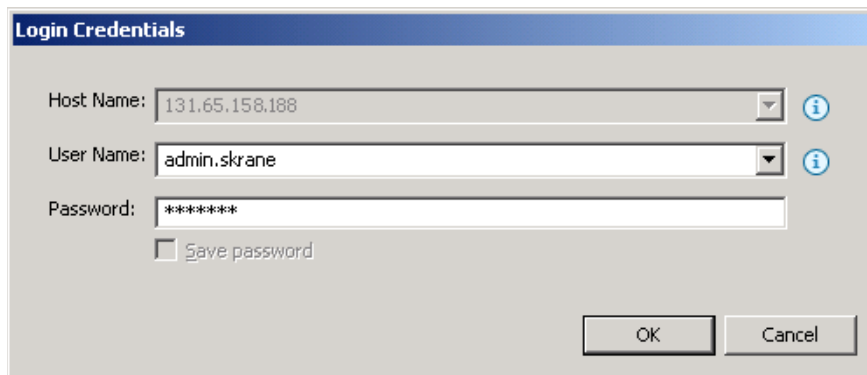
- 1 Click *Tools > eDirectory Object Manager*.



- 2 Select an Identity Vault.









- 3 Provide the appropriate directory credentials, then click *OK*.



- 4 Access information by using icons on the eDirectory Object Manager's toolbar

Table 21-3 Icons on the eDirectory Object Manager's toolbar

Icon	Descriptions
	Expands all containers in the currently selected tree. IMPORTANT: This might be a heavyweight operation if you have a million-object tree.
	Collapses all expanded containers in the currently selected tree.
	Adds a new custom tree, which persists across sessions.

Icon	Descriptions
	Removes trees previously added with the Add Tree operation. Automatically discovered trees cannot be removed.
	Refreshes the currently active tree.
	Displays an object's properties.

Expansion states and selection states are persistent between sessions per tree.

21.5.2 Task-Based Method

To use the eDirectory Object Manager to browse the Identity Vault in your current project:

- 1 Select *Live > Browse Identity Vault*.

You can also right-click the Identity Vault object and select *Live > Browse Identity Vault*.

- 2 In the eDirectory Browser, browse to and select an object.

The eDirectory Object Manager automatically does the following:

- ♦ Populates the field by inserting the Identity Vault directory credentials (Host name, User name, Password)
- ♦ Displays the Identity Vault directory structure

21.5.3 Browse, View, or Modify Object Attributes

After you have populated eDirectory Object Manager with one or more directories, you can browse the directory tree for specific objects, and view and modify object attributes.

Table 21-4 *Objects That You Can Modify*



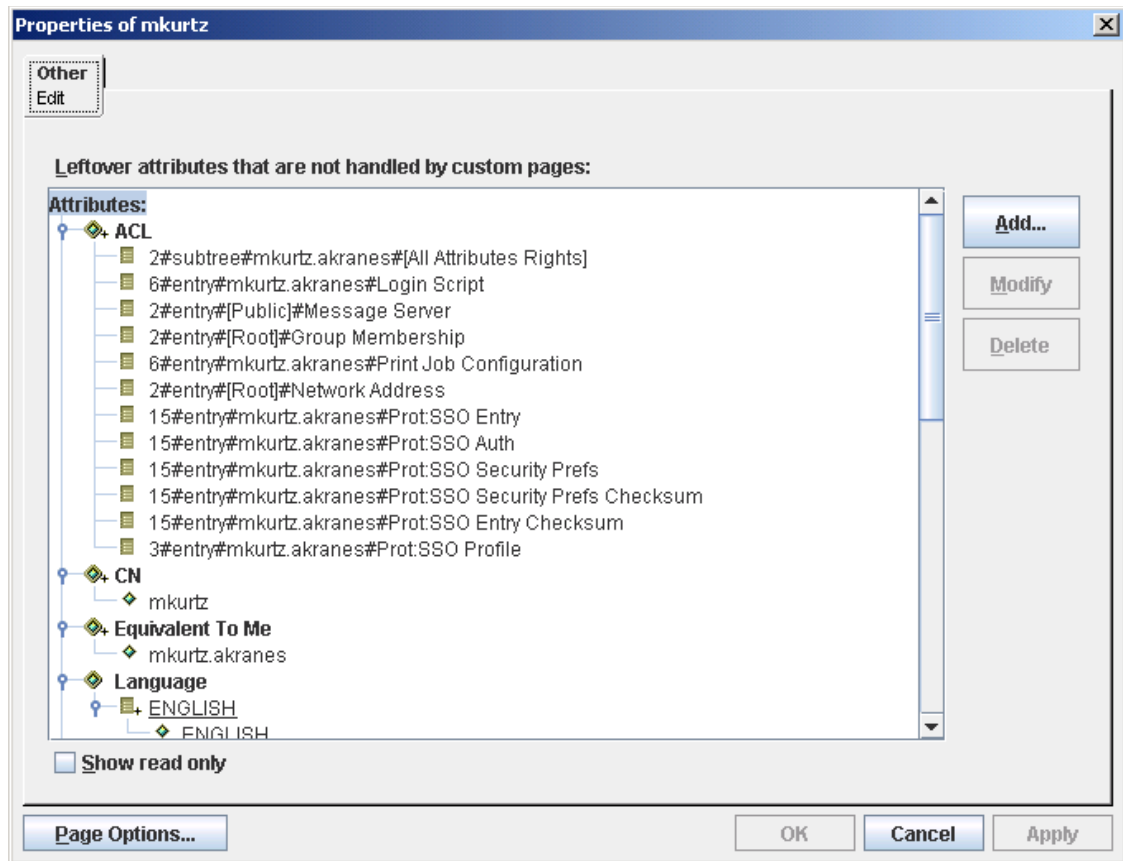
Object	Description
Container Object	<p>Double-click to expand a collapsed container or to collapse an expanded container.</p> <p>Right-click and select <i>Properties</i> to open that object's attributes page.</p> <p>Select the object and click the <i>Open properties of this object</i> icon  in the Action bar.</p>
Leaf Object	<p>Double-click (or right-click and select <i>Properties</i>) to open that object's attributes page.</p> <p>Select the object and click the <i>Open properties of this object</i> icon  in the Action bar.</p>

Figure 21-11 eDirectory Object Manager Attributes List



21.5.4 Advanced Editing

Figure 21-12 The Advanced Object Editor

Properties of bjesse

Advanced Snapin | Other |
My second snap-in

Given Name: Ben

Last Name: Jesse

Full Name: Ben Jesse

Account: disabled

Password Expires: July 6, 2007 7:04:00 PM MDT

Description: Ben Jesse, troubleshooter

Photo:

Groups: 00-Agents.groups.data

[1/1] Add Delete

eMail: 707@mi.uk

Phone: 222-222-2222

Fax: 444-444-4444

Pager: 555-555-5555

mobile: 333-333-3333

OK Cancel Apply

21.6 Configuring TLS for eDir-to-eDir Drivers

- ◆ [Section 21.6.1, “Prerequisites,” on page 427](#)
- ◆ [Section 21.6.2, “Enabling TLS,” on page 428](#)
- ◆ [Section 21.6.3, “Creating Certificates,” on page 430](#)

21.6.1 Prerequisites

- ◆ Identity Vaults exist in your physical network tree as well as in the Modeler.
- ◆ Each Identity Vault is set up. Otherwise, you are prompted for setup information when you try to create certificates.
- ◆ Each driver set is associated with a server.
- ◆ Using the eDir-to-eDir driver’s General property page, verify that each driver has a name and a deploy context. The context might be inherited from the driver set.
- ◆ The eDir-to-eDir drivers have been deployed. Otherwise, Designer cannot create certificates. To find out whether the driver has been deployed:

1. Right-click the eDir-to-eDir driver.

2. Click *Live > Deploy*.
3. In the eDir-eDir Driver Deployment dialog box, click *No*.

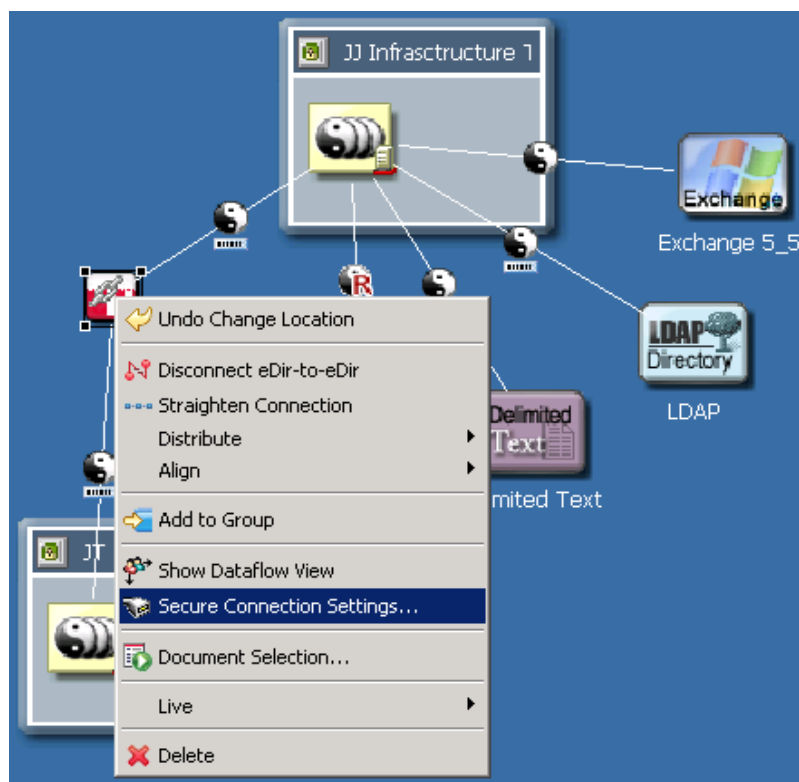
If the driver has been deployed, the *Compare Status* field in the Deployment Summary dialog box displays *Equal* or *Unequal*. Otherwise, the field displays *Not Deployed*.

After objects have been deployed, the objects should show as equal unless passwords are set in eDirectory that are not set in Designer. Designer does not deploy passwords unless they are specifically set in Designer. This exception prevents overwriting passwords in eDirectory because Designer cannot import them.

21.6.2 Enabling TLS

- 1 Launch the TLS Configuration dialog box.

A common way to launch the dialog box is to right-click the eDir-to-eDir application, then click *Secure Connection Settings*.

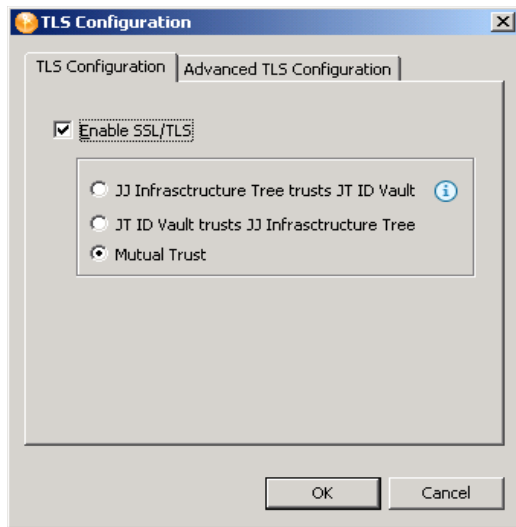


Other launch points:

- ♦ Select the eDir-to-eDir application, then click *Model > eDir-to-eDir > Secure Connection Settings*.
- ♦ Right-click *eDir-to-eDir* in the Outline view, then click *Secure Connection Settings*.
- ♦ Right-click an eDir-to-eDir driver, click *Properties > Driver Configuration > Authentication*, then click *Configure TLS*.

The *Configure TLS* button displays only on eDir-to-eDir driver pages.

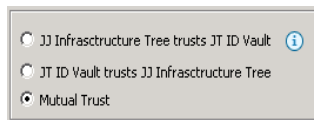
2 Click *Enable SSL/TLS*.



3 (Optional) Use the *Advanced TLS Configuration* to select key size, hash algorithm, and validity period.

The validity period is important for when a certificate has expired and you need to overwrite or create a new one.

4 Select a direction of trust.



These options apply to certificates that Novell creates for eDirectory. The options do not apply to third-party security certificates.

The default is Mutual Trust, which could be considered the most secure.

Unless you want to use the certificate for authentication, the option that you select doesn't matter. If only encryption is important, you can select any one of the three options.

If authentication is important, select the option that gives you the appropriate trust.

Scenario: JJ Infrastructure Tree Trusts JT ID Vault. JJ Infrastructure Tree is the organizational Certificate Authority. JJ Infrastructure Tree signed a certificate and placed it in JT ID Vault. JT ID Vault trusts JJ Infrastructure Tree. The two vaults synchronize data through a secure connection.

If the two vaults break their trusted relationship, JJ Infrastructure Tree can prevent sensitive data from being synchronized by revoking its certificate.

Scenario: JT ID Vault Trusts JJ Infrastructure Tree. JJ Infrastructure Tree creates two certificates. One is placed in JJ Infrastructure Tree, and the other is placed in JT ID Vault. The two vaults synchronize data through a secure connection.

If the two vaults break their trusted relationship, JJ Infrastructure Tree can prevent sensitive data from being synchronized by revoking its certificate.

Scenario: Mutual Trust. JT ID Vault and JJ Infrastructure Tree both sign certificates.

5 Click *OK*.

After you click *OK*, Designer does the following:

- ◆ Modifies both eDirectory drivers.
- ◆ Locks the *User ID* field, which displays on the driver configuration's Authentication page, because both drivers must use that field.

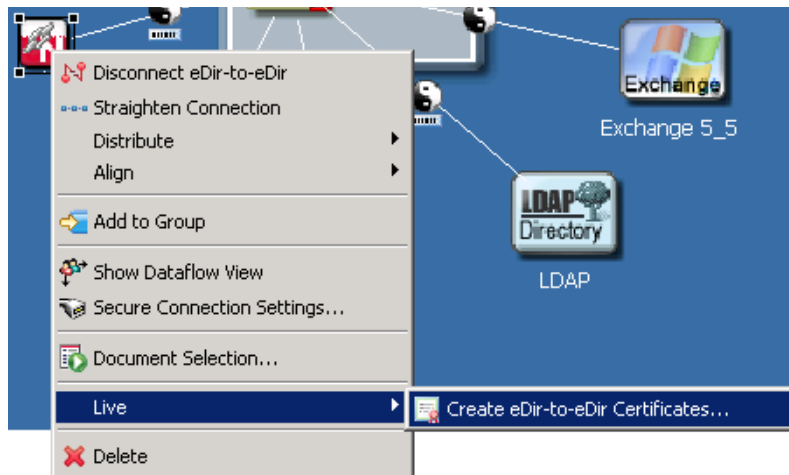
You can enable or configure TLS without immediately deploying the drivers. You can turn the settings on. However, you can't create SSL/TLS certificates unless the drivers have been deployed into their respective Identity Vaults. If you enable SSL/TLS but want to create certificates later, you can do so. When you later deploy the eDir-to-eDir drivers, Designer guides you through steps to automatically create certificates.

21.6.3 Creating Certificates

A driver's Properties page enables you to configure a driver so that you can deploy it. Similarly, the Enable SSL/TLS option enables you to set up your configuration for TLS and then, when you are ready, create and deploy the certificates. When you deploy a configured driver set or select *Create eDir-to-eDir Certificates*, Designer creates the certificates in the directory.

This section assumes that you have enabled and configured SSL/TLS for the deployed eDir-to-eDir drivers.

1 Right-click the eDir2eDir application.



2 Click *Live > Create eDir-to-eDir Certificates*.

You can also do one of the following:

- ◆ Right-click the eDir2eDir object in the Outline view, then click *Create eDir-to-eDir Certificates*.
- ◆ The first time that you enable and configure SSL/TLS on driver's *Authentication* tab, click *OK*, then follow prompts. A Create Certificates dialog box appears. Click *Yes*.

Scenario: Enabling TLS. TLS has not been enabled. Sandy selects *Live > Create eDir-to-eDir Certificates*. Designer prompts Sandy to enable SSL/TLS. Sandy clicks *OK*, enables TLS, selects a direction of trust, and clicks *OK*. Designer creates certificates.

Scenario: Deploying eDir-to-eDir Drivers. Sandy has configured the eDir-to-eDir drivers and the driver set. A context displays in the driver set's *Deploy Context* field. Sandy is ready to deploy the driver set.

Sandy right-clicks the driver set, then clicks *Live > Deploy Driver Set*. Designer prompts Sandy to deploy both eDirectory drivers. (Otherwise, Designer can't successfully create certificates.) Sandy clicks *Yes*. Designer builds a deployment summary, then lists items that are associated with the Identity Vaults and will be deployed. To deploy the drivers, Sandy clicks *Deploy*.

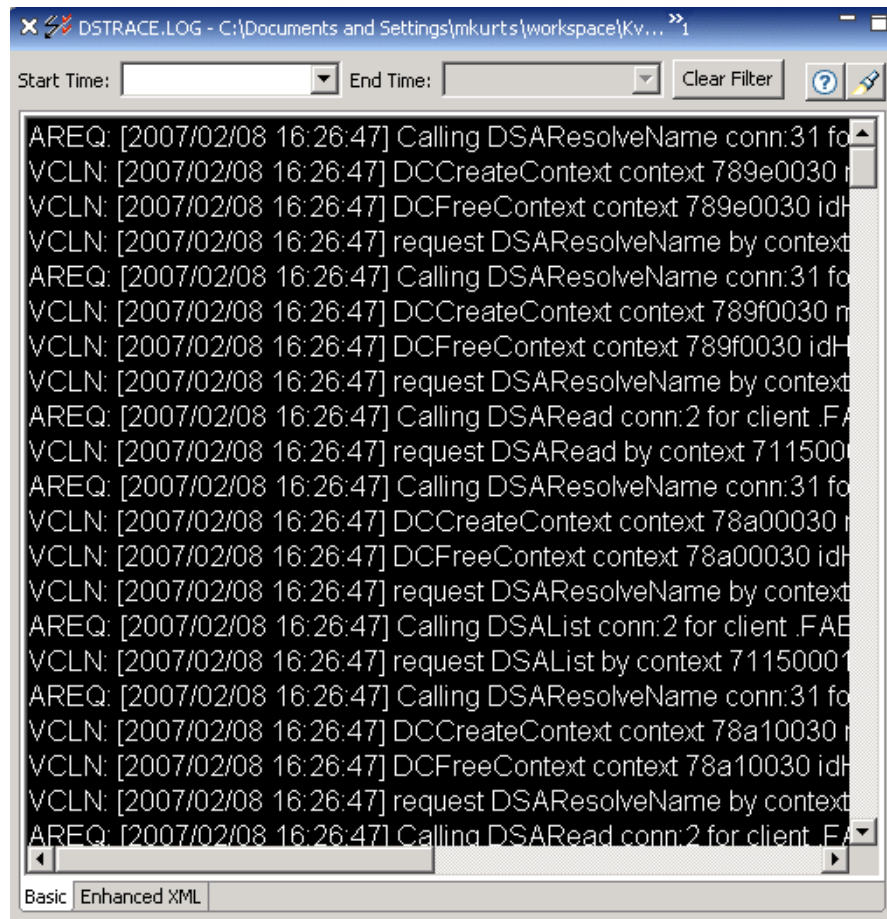
Because the driver set is already configured, Designer creates the certificates.

For additional information on eDir-to-eDir certificates, see [eDir-to-eDir SSL/TLS](#) in *Preferences*.

21.7 Viewing DSTrace Log Files

The DSTrace view is an editor that enables you to view DSTrace log files. These log files contain information (including XML data) about the Metadirectory.

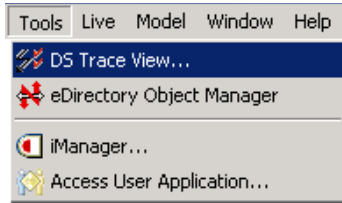
Figure 21-13 The DS Trace View



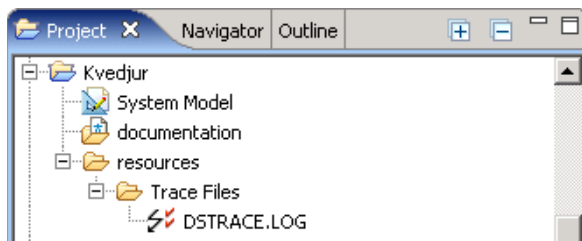
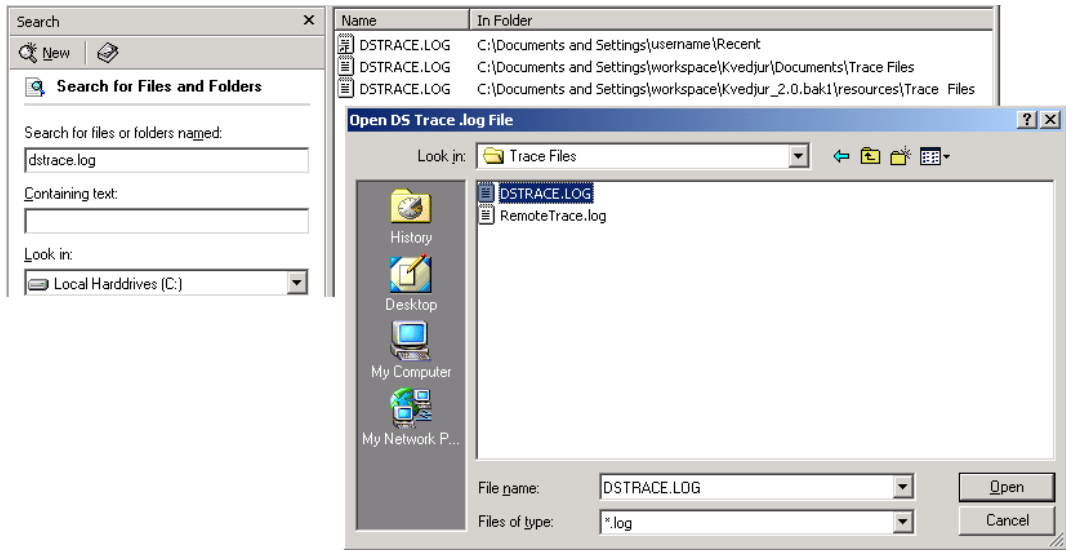
The DSTrace view is not the same as the Trace view, which provides information about Designer functionality. For information on the Trace view, see [Section 23.3.13, "Trace,"](#) on page 467.

To view DSTrace log files:

- 1 Click *Tools > DS Trace View*.

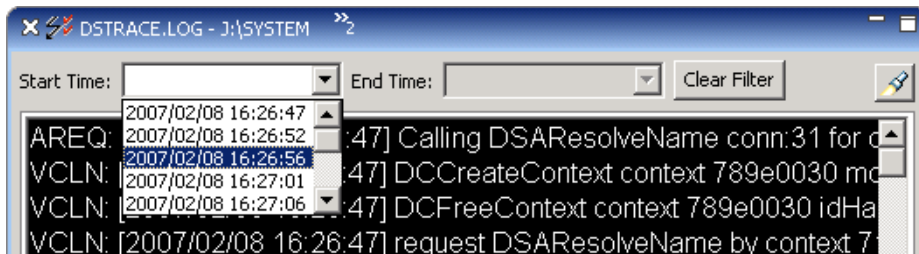


- 2 Navigate to and open a DSTrace log file.

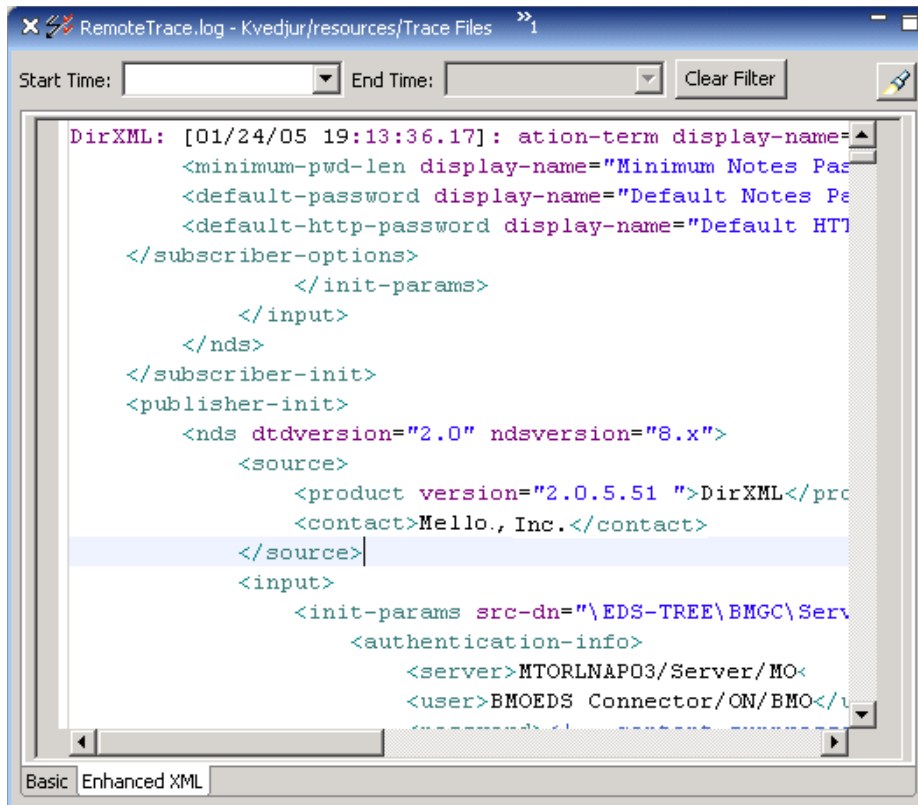


To import these log files, drag them from the file browser or the desktop and drop them in the Project view's *resources* folder.

- 3 Filter the trace based on date and time.

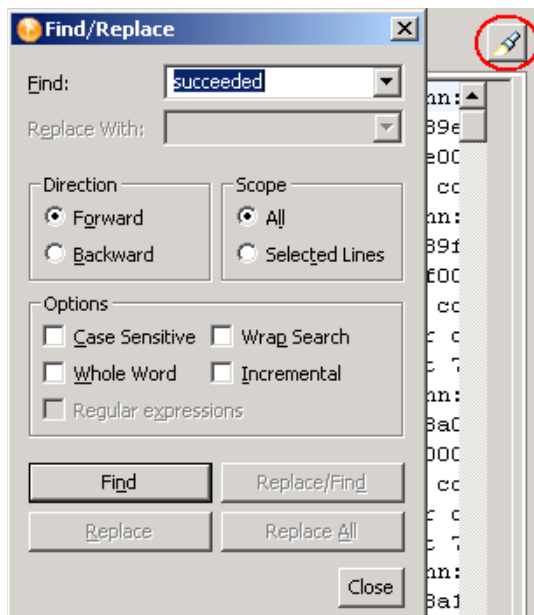


4 View XML information in the .log file by clicking *Enhanced XML*.



Designer highlights the XML code. This feature can be helpful when you want to see the various parts of the XML documents in the trace file. If no XML exists in the file, content in the Basic tab displays.

5 Search for content that you are interested in.



To quickly find information in the .log file, click the Search icon, enter text in the *Find* field, then click *Find*.

You can also access the Find/Replace dialog box by pressing Ctrl+F or by clicking the main menu's *Edit > Find/Replace* path.

21.8 Working with Generic Resources

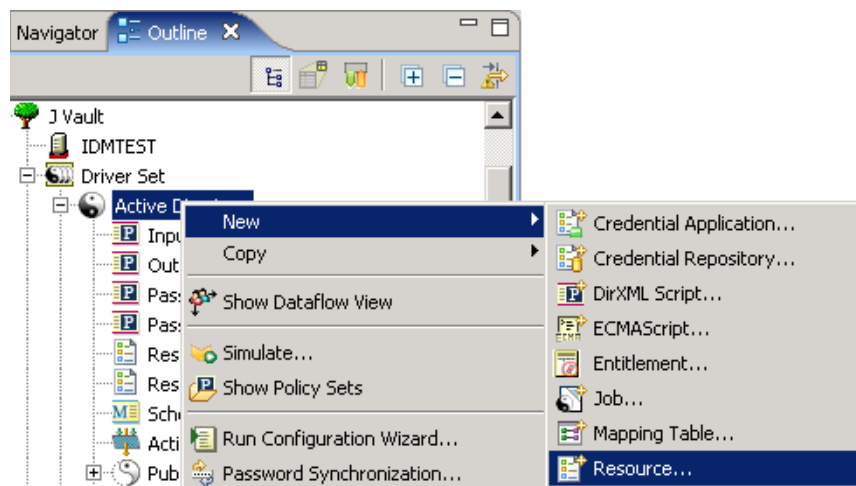
A Resource object is stored in a Driver object or a library. A Resource object stores parameters, which drivers use at any time. When multiple drivers need the same set of constant parameters, the drivers use a Resource object.

A Generic Resource object in Designer enables you to store information in XML or text format. The information can be a piece of documentation, notes, or some piece of data that policies access.

- ♦ [Section 21.8.1, “Creating a Generic Resource Object,” on page 434](#)
- ♦ [Section 21.8.2, “Editing a Generic Resource Object,” on page 435](#)

21.8.1 Creating a Generic Resource Object

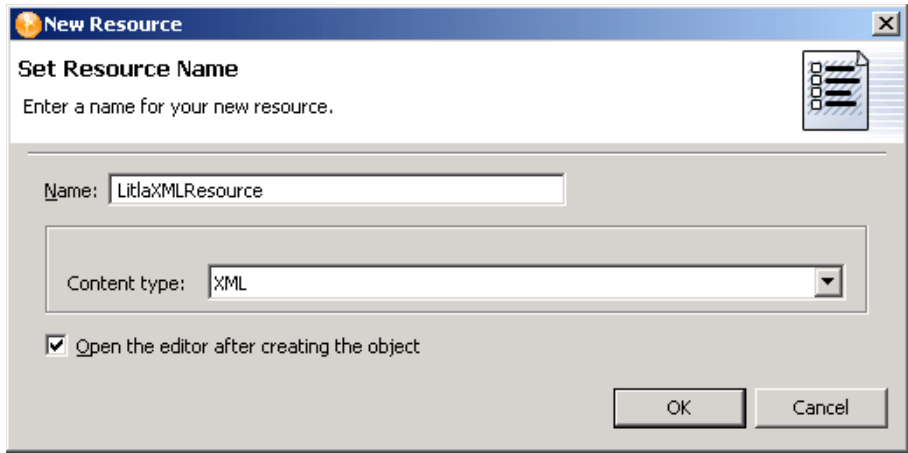
- 1 In the Outline view, right-click a driver, then select *New > Resource*.



You can also do one of the following:

- ♦ Right-click a driver, then select *New > Resource*.
- ♦ With the Dataflow view active, right-click a Subscriber or Publisher channel, then select *New > Resource*.
- ♦ In the Outline view, right-click a library, then select *New > Resource*.

- 2 Specify the name of the Generic Resource object.



- 3 Select the content type, *XML* or *Text*.
- 4 Select *Open the editor after creating the object*, then click *OK*.
- 5 In the File Conflict dialog box, click *Yes*.
- 6 Specify the desired XML or text, then press Ctrl+S to save the resource object.

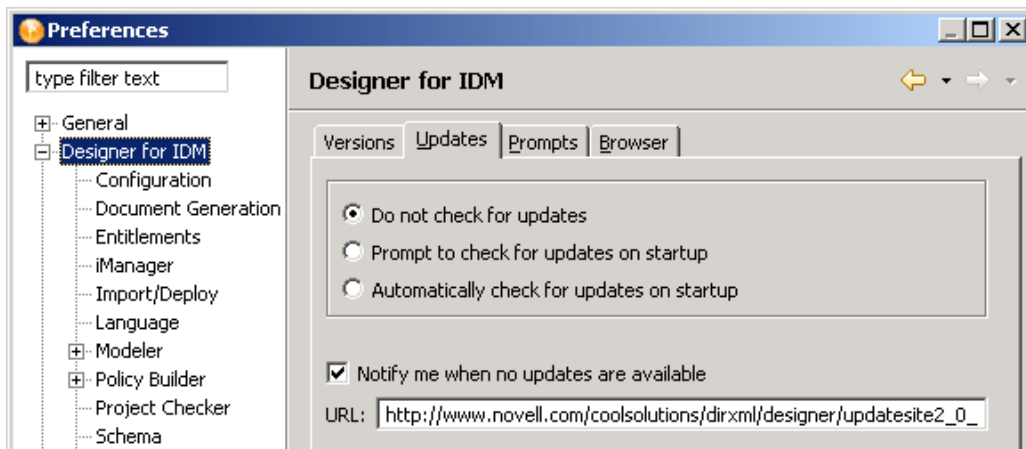
21.8.2 Editing a Generic Resource Object

- 1 In the Outline view, below the library, right-click the generic resource object, then select *Edit*.
- 2 In the File Conflict dialog box, click *Yes*.
- 3 Make changes, then save (Ctrl+S).

21.9 Updating Designer

When you start Designer, you are prompted about how you want to receive updates. You can change this setting in Preferences.

Figure 21-14 The Updates Tab



If you select to not automatically update Designer, you can get updates by using the Help menu or the Welcome page.

To update from the Help menu:

- 1** Click *Help > Check for Designer Updates*.
 - ◆ If your version of Designer is up-to-date, a prompt informs you that no updates are available.
 - ◆ If an update is available, a prompt lists components that you can update. Select components, then click *OK*.
 - ◆ If your version of Eclipse needs to be updated before you can install Designer 2.1, a dialog box prompts you to click the URL that takes you to the Designer download site.
- 2** Select updates, then click *OK*.

To update from the Welcome page:

- 1** Click *Help > Welcome*.
- 2** Click the *What's New* icon.



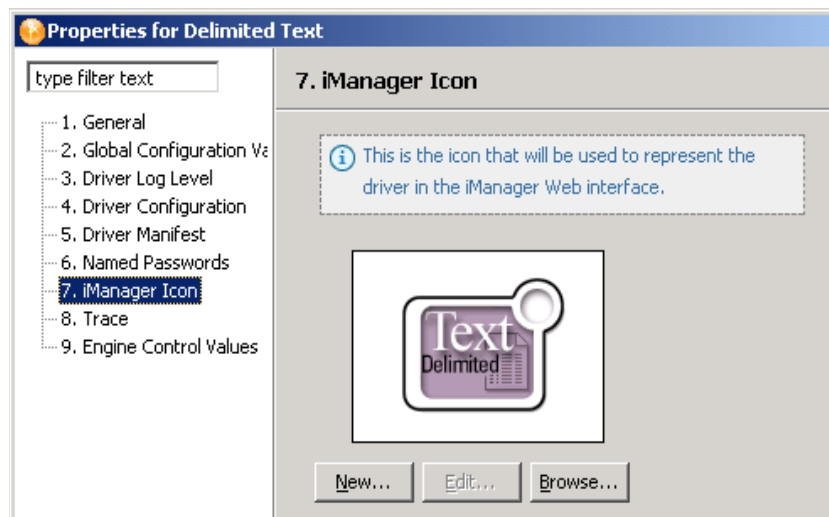
- 3** Click *New Updates*.
- 4** Follow prompts to download and install the latest Designer.

Editing Icons for Drivers and Applications

22

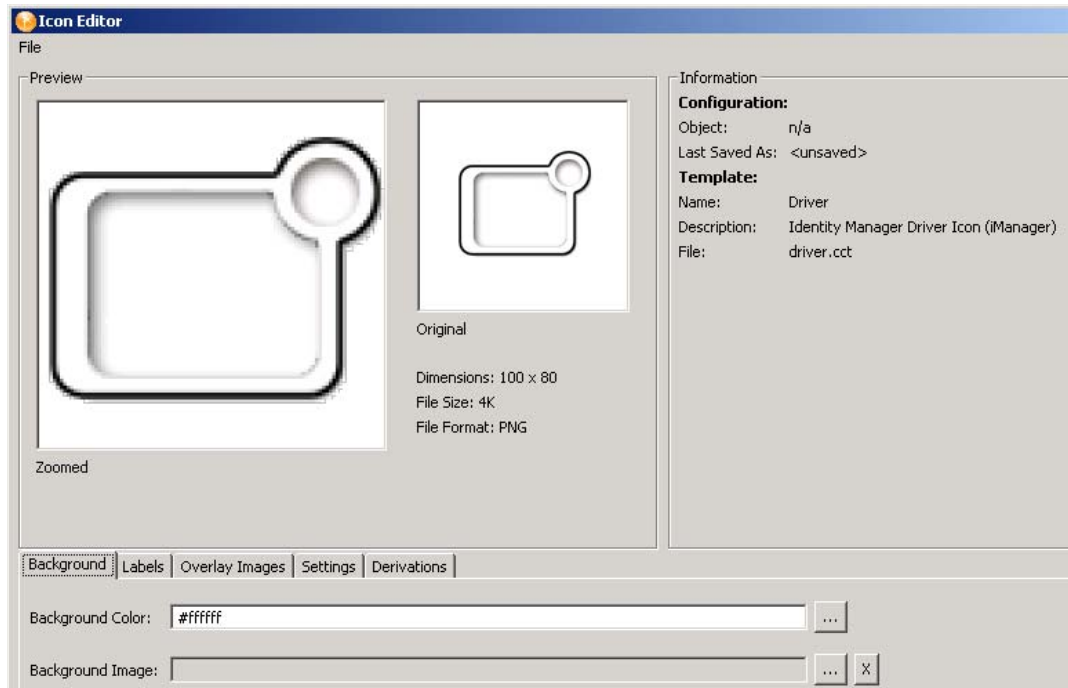
The Icon editor enables you to create customized icons for your drivers and applications. You can enrich predefined templates or images with labels, choose background images and colors, and overlay images. Within minutes, you can create a custom branded icon for your implementation, including your company's logo or name.

- 1 In the Modeler, select a driver, then click *Properties*.

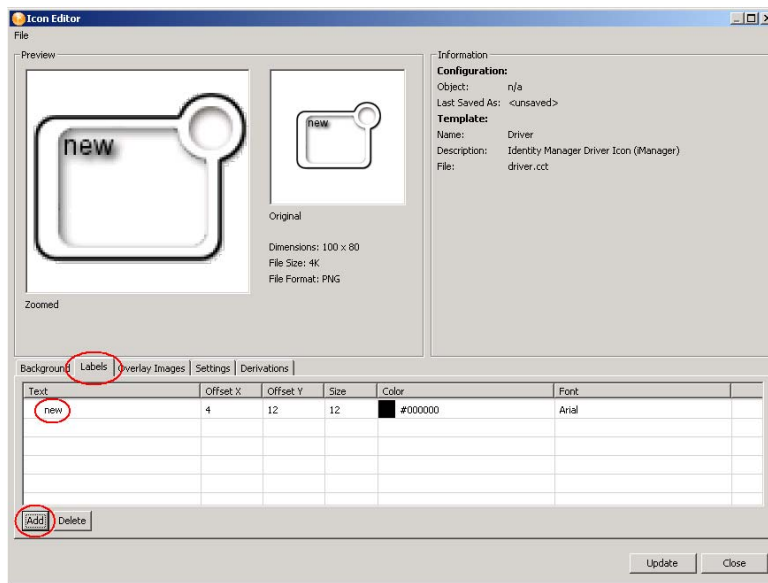


The object properties dialog box displays the default icon.

2 Click *iManager Icon > New*.



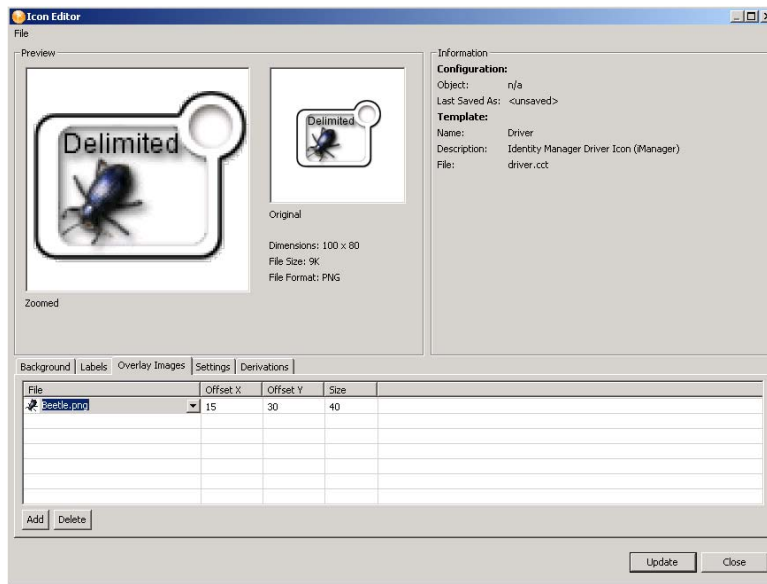
3 To enter text for your icon, select *Labels*, then click *Add*.



To edit the label, click *new* in the *text* column, type a new label, then click *Update*.

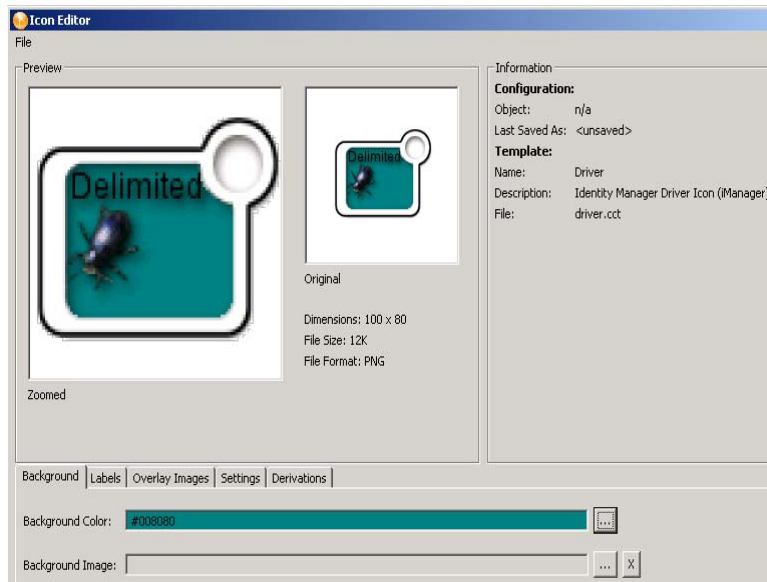
You can edit the label, control its placement through the Offset X and Offset Y options, and edit font size, font type, and color.

- 4 To add an overlay image to your icon, select *Overlay Images*, then click *Add*.



To select an image, click the cell in the *File* column, select a filename from the drop-down list, then click *Update*. To place the image farther below the text, increase the value in the *Offset Y* column.

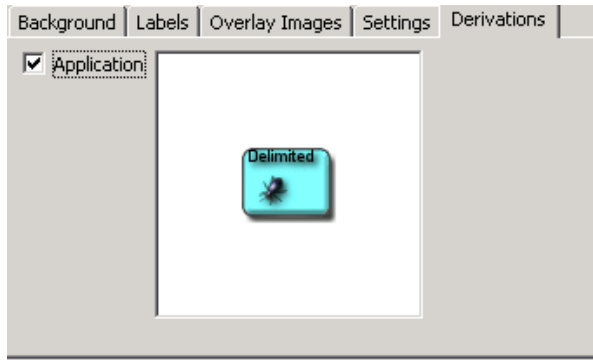
- 5 To add a background color or image to your icon, select *Background*.



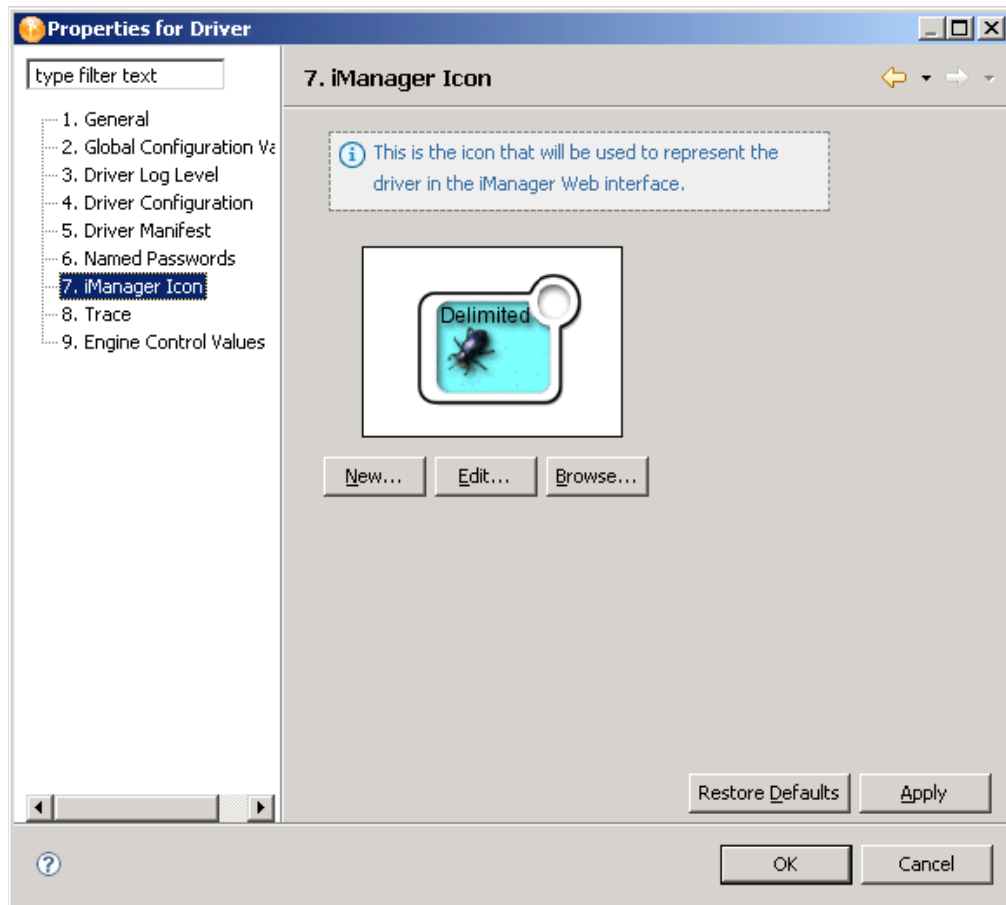
To change the background color, click the Browse button next to the *Background Color* field, select a color from the palette, then click *OK*. You can also click the Browse button next to the *Background Image* field, select a filename from one of the folders (for example, *gradients*), then click *Open*.

To change the background image, click the Browse button next to the *Background Image* field, select an image from the *applications* folder, then click *Open*.

6 To create a similar icon for iManager, select *Derivations*, then select *Driver* or *Application*.



This transfers the icon to the driver's or application's iManager properties. (See the iManager icon on the driver properties page.)



7 When you are finished editing the icon, click *Update*.

Setting Preferences

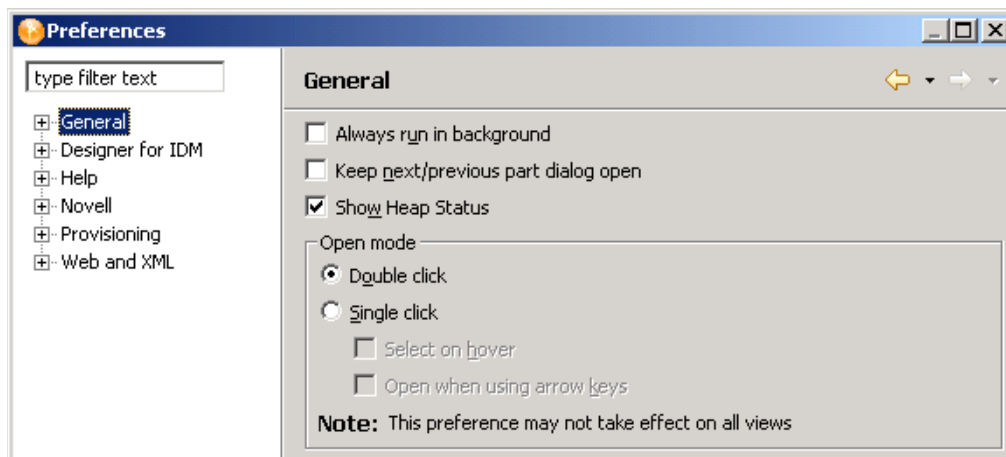
23

- ♦ Section 23.1, “Finding Preference Pages,” on page 441
- ♦ Section 23.2, “Preferences: General,” on page 441
- ♦ Section 23.3, “Preferences: Designer for IDM,” on page 448
- ♦ Section 23.4, “Preferences: Help,” on page 469
- ♦ Section 23.5, “Preferences: Novell,” on page 470
- ♦ Section 23.6, “Preferences: Provisioning,” on page 470
- ♦ Section 23.7, “Preferences: Web and XML,” on page 470

23.1 Finding Preference Pages

You customize Designer by setting options in Preferences.

- 1 From the main menu, select *Window > Preferences*.




- 2 Select a heading (for example, *Designer for IDM*) or navigate to a subheading.
- 3 Make changes, then click *Apply* or *OK*.

23.2 Preferences: General

- ♦ Section 23.2.1, “Appearance,” on page 442
- ♦ Section 23.2.2, “Capabilities,” on page 444
- ♦ Section 23.2.3, “Compare/Patch,” on page 444
- ♦ Section 23.2.4, “Content Types,” on page 444
- ♦ Section 23.2.5, “Editors,” on page 445
- ♦ Section 23.2.6, “Keys,” on page 447
- ♦ Section 23.2.7, “Perspectives,” on page 447
- ♦ Section 23.2.8, “Startup and Shutdown,” on page 447

- ◆ Section 23.2.9, “Web Browser,” on page 448
- ◆ Section 23.2.10, “Welcome,” on page 448

Table 23-1 Preferences: General

Setting	Description
	Searches all the preferences and shortens the tree view, depending upon what you type in the edit box.
<i>Always run in background</i>	Enables operations to run in the background without disturbing you.
<i>Keep next/previous part dialog open</i>	Keeps the editor and view dialog boxes open when an activation key is released. Normally, the dialog box closes as soon as the key combination is released.
<i>Show heap status</i>	Places a field in Designer’s bottom right corner and displays the amount of memory being used of total memory available.
<i>Open mode: Double click</i>	Opens a project when you double-click it.
<i>Single click: Select on hover</i>	Selects the setting when the cursor hovers there.
<i>Single click: Open when using arrow keys</i>	Opens the setting when you select it.

23.2.1 Appearance

Figure 23-1 Preferences: Appearance

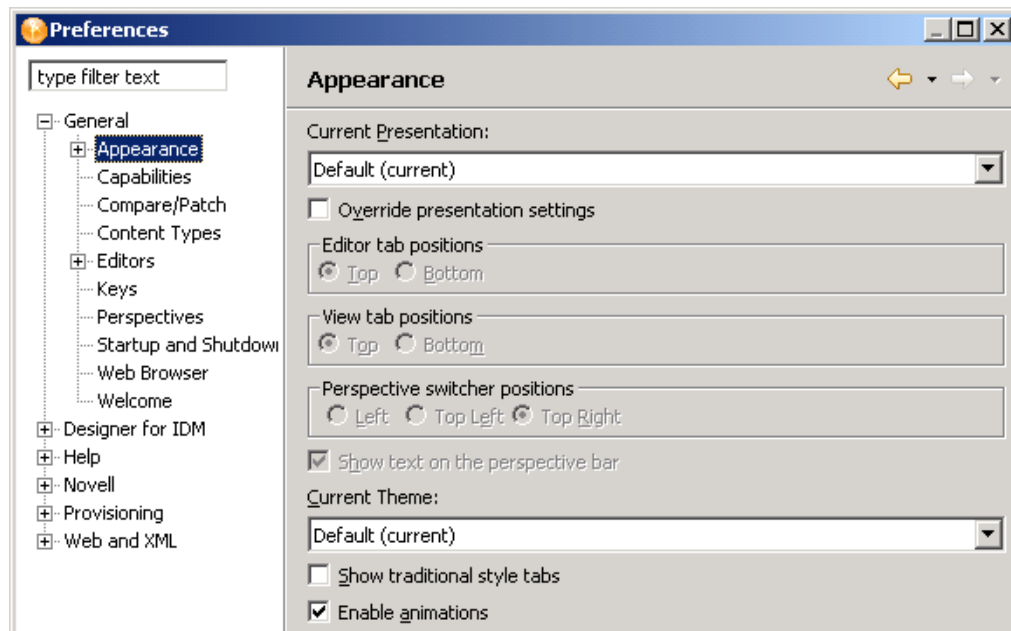



Table 23-2 Preferences: Appearance

Setting	Description
<i>Current Presentation: Override presentation settings</i>	Alters how the tabs and views appear in the workbench
<i>Editor tab positions</i>	Positions tabs on the Modeler, Novell® XML Editor, or Text Editor at the top or bottom.
<i>View tab positions</i>	Positions view tabs (for example, the Project view tab) at the top or bottom of views.
<i>Perspective switcher positions</i>	Positions the Perspective Switcher  at the left, top left, or top right of the workbench.
<i>Show text on the perspective bar</i>	Determines whether text (for example, Designer) displays next to the icons in the Perspective Switcher.
<i>Current theme</i>	The general theme (colors and fonts) that Designer uses.
<i>Show traditional style tabs</i>	Displays square Windows-style tabs. The alternative is rounded tabs.
<i>Enable animations</i>	Animates views (for example, Fast Views) and editors that you minimize, maximize, or restore. Reinforces tasks in Designer.

- ♦ [“Colors and Fonts” on page 443](#)

Colors and Fonts

To change a color:

- 1 Expand an option (for example, *Basic*).
- 2 Select an item (for example, *Active hyperlink text color*).
- 3 Click the color button.



- 4 Select a color from the Color palette, then click *OK*.

To change a font:

- 1 Expand an option (for example, *Basic*).
- 2 Select an item (for example, *Banner Font*).
- 3 Click *Change*.
- 4 Select a font, style, and size, then click *OK*.

23.2.2 Capabilities

Currently, this section relates to Eclipse functionality. Designer functionality will be implemented later.

- ♦ “Prompt When Enabling Capabilities” on page 444

Prompt When Enabling Capabilities

Eclipse lets you turn off capabilities. However, Eclipse sometimes detects that you need the capability, depending on how you use the product, and automatically turns the capability on again. This feature doesn't apply to Designer 2.1.

23.2.3 Compare/Patch

This Eclipse functionality customizes the behavior of the comparison editor. When you select to compare or synchronize two or more resources in the Workbench, one or more comparison editors usually open.

Table 23-3 Preferences: Compare/Patch

Setting	Description
<i>Open structure compare automatically</i>	Makes visible an additional information area that shows differences in the underlying structure of the resources being compared. This information might not be available for all comparisons. The default is <i>On</i> .
<i>Show additional compare information in the status line</i>	Causes the status line to display additional context information about the comparison. The default is <i>Off</i> .
<i>Ignore white space</i>	Causes the comparison to ignore differences that are white space characters (for example, spaces and tabs). Also causes differences in line terminators (LF versus CRLF) to be ignored. The default is <i>Off</i> .
<i>Automatically save dirty editors before patching</i>	Controls whether any unsaved changes are automatically saved before a patch is applied. The default is <i>Off</i> .

For information on *Text Compare*, refer to the Eclipse documentation.

23.2.4 Content Types

Table 23-4 Preferences: Content Types

Pane	Description
<i>Content types</i>	The type of content (for example, HTML or XML) that a file contains.

Pane	Description
<i>File associations</i>	<p>The file extension that is associated with a content type. For example, <code>.xml</code> is associated with a file that contains XML content. To add a file association:</p> <ol style="list-style-type: none"> 1. Select a content type. 2. Click <i>Add</i>. 3. Define a new file type, then click <i>OK</i>.

23.2.5 Editors

Table 23-5 *Preferences: Editors*

Setting	Description
<i>Size of recently opened files list</i>	The number of files to add to the file menu of recently opened files, which you can easily reopen.
<i>Show multiple editor tabs</i>	Displays tabs for all opened projects. If you deselect this option, only one editor tab displays. However, an abbreviated name displays on the editor tab.
<i>Close editors automatically</i>	Automatically closes the first-opened editor when you open additional editors. This option prevents displaying too many editors and cluttering the workbench.
<i>Number of opened editors before closing</i>	Determines how many editors can be open. For example, if you specify two and then open a third project, the first-opened project automatically closes.
<i>When all editors are dirty or pinned</i>	Prompts you to save unsaved components in the project that is about to automatically close, or to open an additional editor.

- ◆ [“File Associations” on page 445](#)
- ◆ [“Structured Text Editors” on page 446](#)
- ◆ [“Text Editors” on page 446](#)

File Associations

Enables you to associate editors (whether internally installed in the Designer, or an external application) with file types (extensions) so that you can edit files.

To find out which editor is associated with a file type, select the file type. For example, a `.docgen` file type is associated with Style Editor, but a `.scriptpolicy` file type is associated with Policy Builder.

To associate an additional editor with a file type:

- 1 Select the file type.
- 2 In the Associated Editors pane, click *Add*.
- 3 Select an additional editor, then click *OK* twice.

To add a file type:

- 1 In the *File Types* pane, click *Add*.
- 2 Type the extension (for example, `.doc`) for the file type, then click *OK*.
- 3 In the *Associated Editors* pane, click *Add*.
- 4 Select an editor for that file type, then click *OK* twice.

Structured Text Editors

For information on structured text editors, refer to the Eclipse documentation.

Text Editors

Table 23-6 Preferences: Text Editors

Setting	Description
<i>Displayed tab width</i>	The number of characters or spaces moved when you press the Tab key in a text editor. The default is 4. The maximum is 16.
<i>Undo history size</i>	Determines how many changes that you can undo in an editor. The default is 25 changes.
<i>Highlight current line</i>	Selects the line that has focus.
<i>Show print margin</i>	Controls whether the print margin is visible. A vertical line identifies the margin.
<i>Show line numbers</i>	Numbers each line in the editor.
<i>Show range indicator</i>	Controls whether an editor displays a range indicator.
<i>Support hyperlink style navigation</i>	Determines whether hyperlink-style navigation is supported. If you put a link (for example, <code>www.novell.com</code>) in a text editor, press Ctrl, and hover the cursor over the link, Designer launches the link and loads it into an HTML view inside Designer.
<i>Disable overwrite typing mode</i>	Controls whether Overwrite Typing mode is enabled. The default is <i>Off</i> .
<i>Appearance color options</i>	To change an option's color, select the option, click <i>Color</i> , select a color, then click <i>OK</i> .

For additional information on text editors, see the Eclipse documentation.

23.2.6 Keys

Enables you to view a table of all of the keyboard mappings, change those mappings, and add new mappings.

23.2.7 Perspectives

Table 23-7 Preferences: Perspectives

Setting	Description
<i>Open a new perspective</i>	<i>In the same window:</i> Places a new icon in the Perspective Switcher, so that you can toggle between perspectives in the same window. <i>In a new window:</i> Opens a new perspective in a different window. You can toggle between perspective windows by selecting icons on the taskbar.
<i>Open a new view</i>	<i>Within the perspective:</i> Opens the view so that it is contiguous to the Modeler. <i>As fast view:</i> Opens the view and places a Fast view in the bottom left corner of the perspective.
<i>Open the associated perspective when creating a new project</i>	Determines how and when you switch to an associated editor when you open a perspective.
<i>Available perspectives</i>	Designer is the default perspective. Use Designer.

23.2.8 Startup and Shutdown

Table 23-8 Preferences: Startup and Shutdown

Setting	Description
<i>Prompt for workspace on startup</i>	Prompts you for a workspace folder. You can have multiple workspace folders and can specify a folder on startup.
<i>Refresh workspace on startup</i>	Synchronizes the workspace with resources (for example, <code>myfile.xml</code>) on disk.
<i>Confirm exit when closing last window</i>	Displays an <i>Exit Designer?</i> prompt when you exit Designer.
<i>Plug-ins activated on startup</i>	Lists plug-ins that are automatically loaded and registered.

23.2.9 Web Browser

Table 23-9 Preferences: Web Browser

Setting	Description
<i>Use internal Web browser</i>	Enables you to use an internal Web browser.
<i>Use external Web browser</i>	Enables you to add and use an external browser (for example, Netscape*). If you enable this option, you must also enable Use External Browser in the Help section (also found in Preferences).
<i>External Web browsers</i>	Lists browsers. To add a browser: <ol style="list-style-type: none">1. Click <i>New</i>.2. Name the new browser.3. Scroll to and select an executable (for example, <code>netscp6.exe</code>).4. Specify a parameter, then click <i>OK</i>.

23.2.10 Welcome

Table 23-10 Preferences: Welcome

Setting	Description
<i>Home: Home Page Theme</i>	Enables you to select the theme that appears when you click <i>Help > Welcome</i> .
<i>Home: Root Pages</i>	Adds tabs (for example, Overview) on the Welcome properties page. You add functionality by customizing these tabs.

23.3 Preferences: Designer for IDM

- ◆ [Section 23.3.1, “Versions, Updates, Prompts, and Browsers,” on page 449](#)
- ◆ [Section 23.3.2, “Configuration,” on page 451](#)
- ◆ [Section 23.3.3, “Document Generation,” on page 455](#)
- ◆ [Section 23.3.4, “Entitlements,” on page 457](#)
- ◆ [Section 23.3.5, “iManager,” on page 458](#)
- ◆ [Section 23.3.6, “Import/Deploy,” on page 458](#)
- ◆ [Section 23.3.7, “Changing the Language Setting,” on page 460](#)
- ◆ [Section 23.3.8, “Modeler,” on page 461](#)
- ◆ [Section 23.3.9, “Policy Builder,” on page 465](#)
- ◆ [Section 23.3.10, “Project Checker,” on page 466](#)

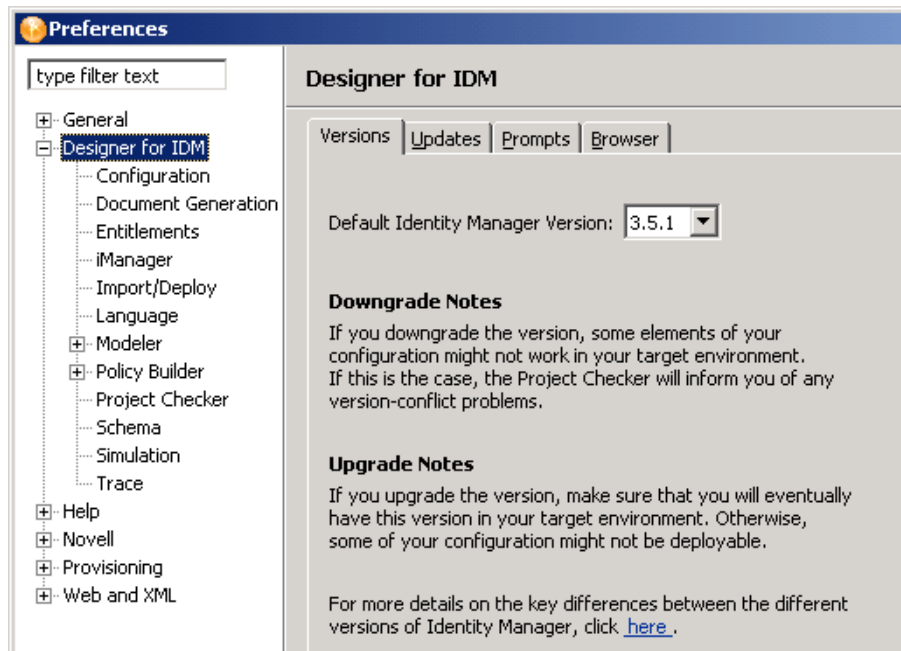
- ◆ Section 23.3.11, “Schema,” on page 466
- ◆ Section 23.3.12, “Simulation,” on page 466
- ◆ Section 23.3.13, “Trace,” on page 467

23.3.1 Versions, Updates, Prompts, and Browsers

- ◆ “Versions” on page 449
- ◆ “Updates” on page 450
- ◆ “Prompts” on page 450
- ◆ “Browsers” on page 451

Versions

Figure 23-2 Preferences: Versions



Specifies the Identity Manager version running on a server.

Updates

Figure 23-3 Preferences: Updates

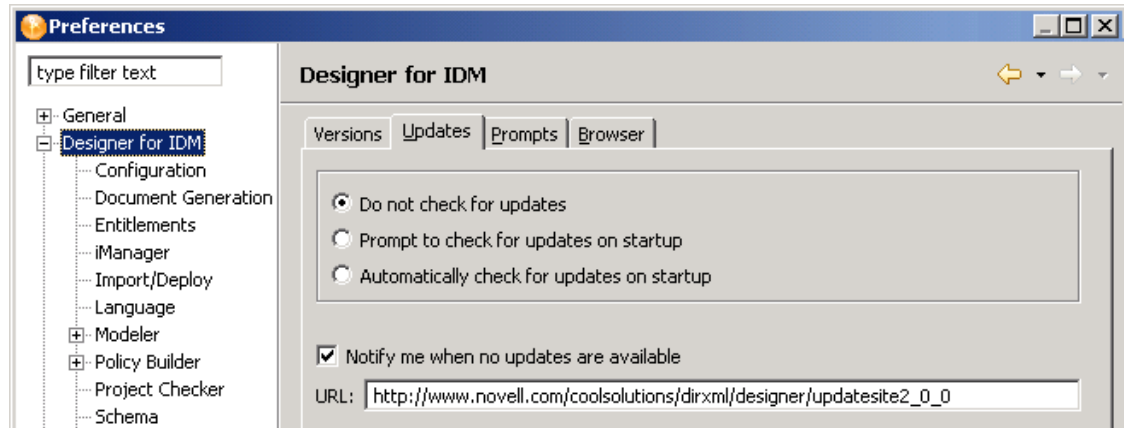


Table 23-11 Preferences: Automatic Updates

Settings	Description
<i>Do not check for updates</i>	Prevents Designer from checking for updates on startup. Hides the Designer Updates dialog box.
<i>Prompt to check for updates on startup</i>	Displays a prompt each time you run Designer. You can disable this prompt.
<i>Automatically check for updates on startup</i>	Always checks for updates. If you disable the prompts that appear on startup, select this option.
<i>Notify me when no updates are available</i>	Displays a <i>No New Updates</i> message when you select to check for updates.

Prompts

Table 23-12 Preferences: Prompts

Setting	Description
<i>Warn when downgrading server versions</i>	Prompts you when you select an earlier server version for a project. If you downgrade, some elements of your configuration might not work in your target environment.
<i>Warn when upgrading server versions</i>	Prompts you when you select a later server version for a project. If you upgrade, some of your configuration might not be deployable unless you have this later server version in your environment.
<i>Warn when another editor has updated files in the same project space</i>	Warns you that your project might be erased from your workspace. The prompt occurs when overwriting a file in the file system for notification templates and policies.

Setting	Description
<i>Warn when deleting items from the outline view</i>	Confirms that you want to delete the selected items.

Browsers

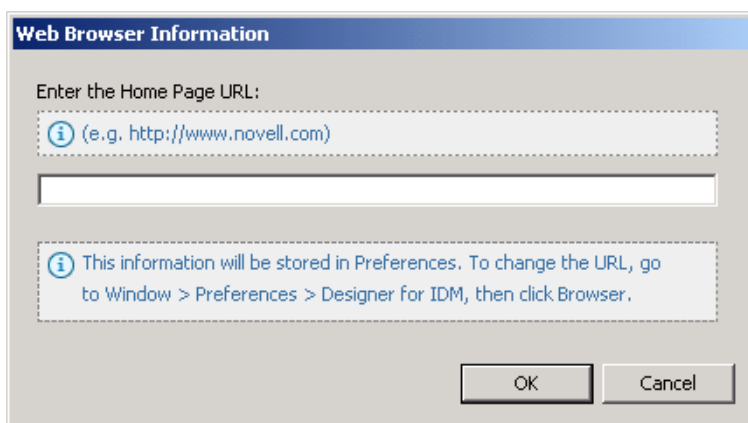
You can use Designer to open a Web browser. The Web browser button is available from the main toolbar.

Figure 23-4 *The Web Browser button*



When you first launch the browser, you are prompted for a home page.

Figure 23-5 *The Prompt for a Home Page*



After you enter the URL, Designer stores it.

To change the URL, type a new one in *Preferences*, then click *OK*.

23.3.2 Configuration

- ◆ “General” on page 451
- ◆ “eDir-to-eDir SSL/TLS” on page 452
- ◆ “Prompts” on page 454

Each driver has a startup parameter. If it is disabled, the driver never starts until you change the setting. By default, Identity Manager drivers are disabled when you create them in the Modeler or start Designer. You must start them manually.

For more information, see [Section 9.5, “Configuring Driver Sets,” on page 152](#).

General

These general settings specify how drivers start up and how their global configuration values (GCVs) act on specified target servers. The default state uses *Disabled* and *Merge GCVs*.

Figure 23-6 Configuring General Driver Settings

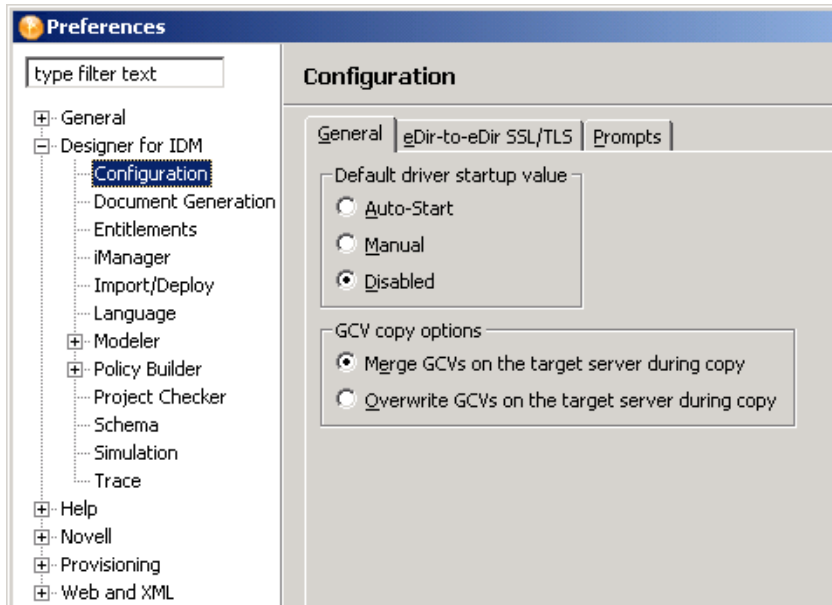


Table 23-13 Driver Settings: General

Setting	Description
<i>Auto-Start</i>	The driver automatically starts after you create it or whenever you start or load Designer.
<i>Manual</i>	You must start the driver manually.
<i>Disabled</i>	The driver never starts.
<i>Merge GCVs on the target server during copy</i>	Copies the GCVs from one driver/driver set to multiple targets of the same type. For example, you might configure GCVs on one driver and then copy them to multiple drivers. You also have the option of overwriting the target GCVs or merging your “source GCVs” with the existing target driver GCVs, if they exist.
<i>Overwrite GCVs on the target server during copy</i>	Overwrites existing GCVs when they are copied to the server.

eDir-to-eDir SSL/TLS

This setting configures how two eDirectory drivers communicate with each other over a secure channel. For more information, see [Section 21.6, “Configuring TLS for eDir-to-eDir Drivers,” on page 427](#).

Figure 23-7 Configuring eDir-to-eDir Drivers

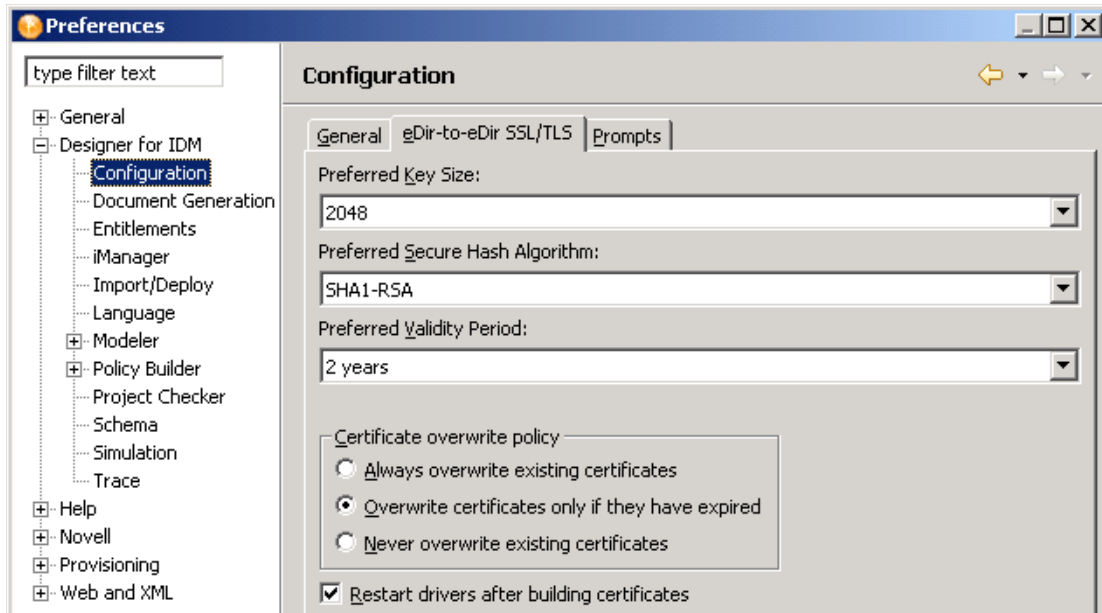


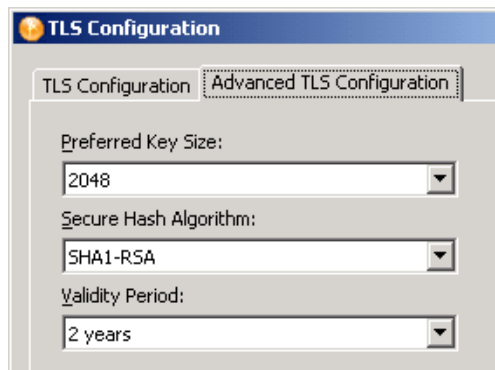
Table 23-14 eDir-to-eDir and SSL/TLS Settings

Setting	Description
<i>Preferred key size</i>	Specifies the preferred key size that is generated when drivers are encrypted and stored in eDirectory: 512, 768, 1024, or 2048 bytes.
<i>Preferred secure hash algorithm</i>	Specifies the preferred hash algorithm to use when encrypting drivers: SHA1-RSA, MD2-RSA, or MD5-RSA.
<i>Preferred validity period</i>	Specifies the validity period for a driver certificate ranging from 6 months to 10 years.
<i>Always overwrite existing certificates</i>	Specifies that existing driver certificates are overwritten with each deployment. If you select this option, Designer deletes existing certificates and creates new ones. The new certificates are then good for another two years (assuming the default value is two years, as defined in the <i>Preferred Validity Period</i> field.) If you select <i>Live > Create eDir-to-eDir Certificates</i> , Designer deletes old certificates and creates new ones
<i>Overwrite certificates only if they have expired</i>	Specifies that only expired driver certificates are overwritten with each deployment. This is the default setting. The default expiration length is two years. If a certificate expires, SSL/TLS stops working. If a certificate is expired, Designer deletes it and creates a new one.
<i>Never overwrite existing certificates</i>	Never overwrites driver certificates.

Setting	Description
<i>Restart drivers after building certificates</i>	Restarts drivers after certificates have been updated/created.

When you create certificates, Designer reads the preferences, including Preferred Key Size, Preferred Secure Hash Algorithm, and Preferred Validity Period. These options are also available through *Secure Connection Settings > Advanced TLS Configuration*.

Figure 23-8 *The Advanced TLS Configuration Dialog Box*



NOTE: Designer reads these preferences after you first set them. If you subsequently change the preferences by using the driver’s configuration page, those changes override the settings in Preferences.

After you change default settings and click *OK*, that configuration information is recorded. When you deploy the driver, Designer creates the certificates, or deletes and creates new certificates with a new time stamp.

Prompts

These settings specify how users are prompted to manage driver certificates on the target server. All are selected in the default state.

Figure 23-9 *Configuring Certificate Prompts*

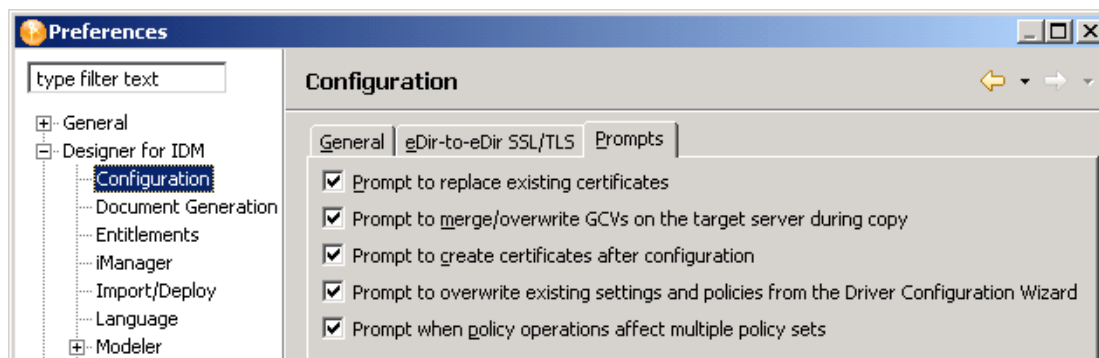


Table 23-15 *Configuration Prompts*

Setting	Description
<i>Prompt to replace existing certificates</i>	Prompts the user to provide new certificates.
<i>Prompt to merge/overwrite GCVs on target server during copy</i>	Prompts the user to merge or overwrite when copying GCVs to the target server.
<i>Prompt to create certificates after configuration</i>	Prompts the user to create certificates after configuring a secure connection.
<i>Prompt to overwrite existing settings and policies from the Driver Configuration Wizard</i>	In the Driver Configuration Wizard, prompts the user whether to reset (overwrite) all driver settings and policies.
<i>Prompt when policy operations affect multiple policy sets</i>	Turns on and off a warning dialog box associated with policy operations. The dialog box appears when you move policies in a pre-3.5 environment and the move operation affects multiple policy sets.

23.3.3 Document Generation

The Document Generator comes with the following settings:

Figure 23-10 Document Generation Settings

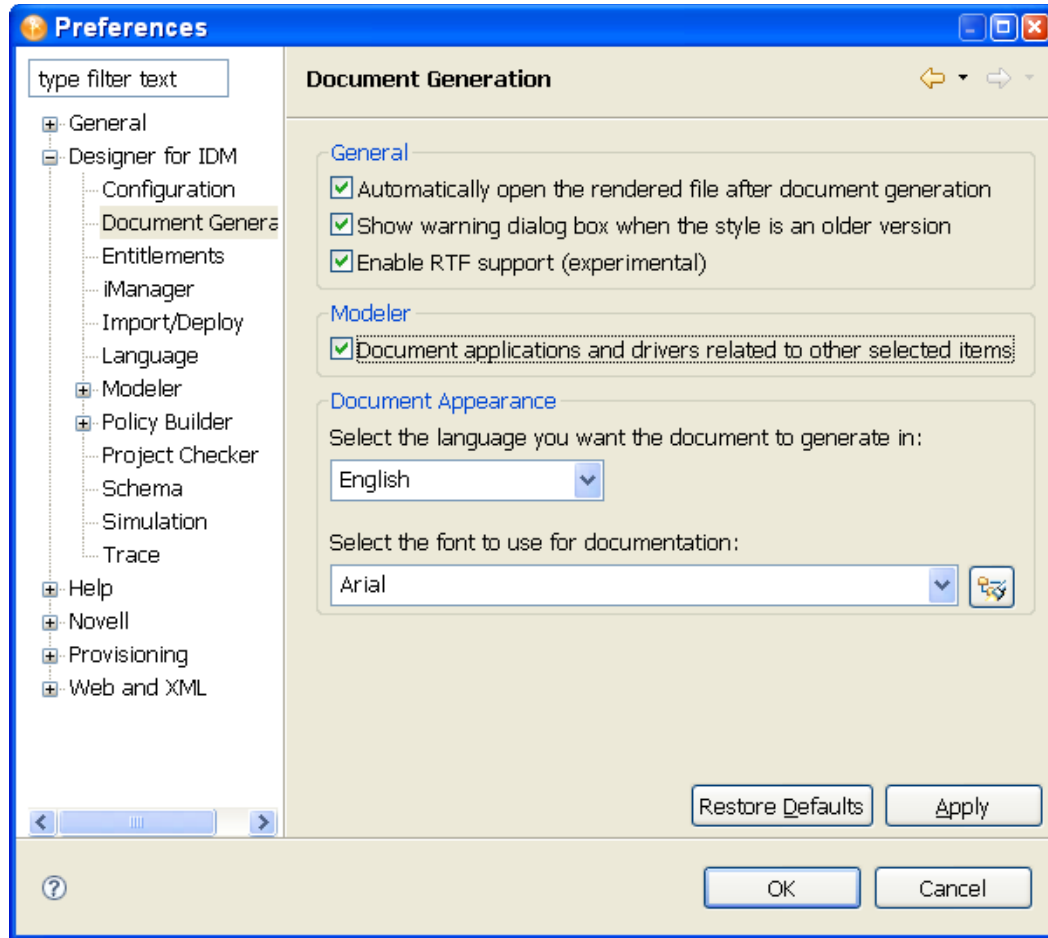


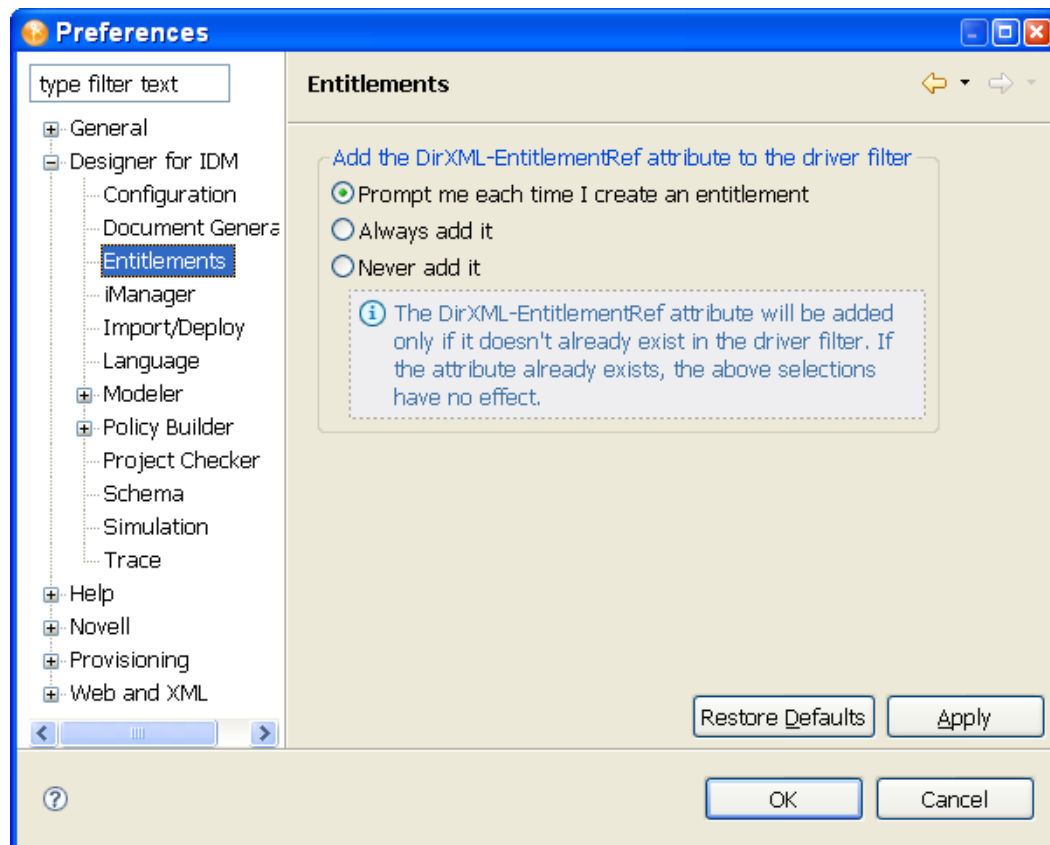
Table 23-16 Document Generation Preferences

Setting	Description
<i>Automatically open the rendered file after document generation.</i>	If you have a PDF reader installed on your workstation, the rendered file automatically opens in the reader. If you have enabled the RTF format and have a TRTF reader installed, the rendered file automatically opens in the reader. The default is On.
<i>Show warning dialog box when the style is an older version.</i>	Shows a warning when generating documents on out-of-version styles. The default is On.
<i>Enable RTF support (experimental).</i>	Allows you to save documents to RTF format. This is a pre-release version. The default is Off.
<i>Document applications and drivers related to other selected items.</i>	With this option selected, parent objects and direct child objects are included to give context to the document. Deselecting this option excludes direct children of the selected item. The default is On.

Setting	Description
<i>Document Language</i>	Allows you to select a language other than English in which to generate documents. Languages include Chinese Simplified, Chinese Traditional, Dutch, English, French, German, Italian, Japanese, Portuguese Brazil, and Spanish. The default is English.
<i>Font settings</i>	Allows you to select the font you want to use for document generation. This selection adds double-byte font support. The default is Arial font.

23.3.4 Entitlements

Figure 23-11 Entitlements Settings



Controls whether or how often you receive a prompt whenever you add the DirXML-EntitlementRef attribute to a driver filter. The default is *Prompt me*, but because this attribute is added only if it doesn't already exist on the driver filter, you can select *Always add it* to not see the pop-up window.

23.3.5 iManager

Table 23-17 Preferences: iManager

Setting	Description
<i>iManager URL</i>	. The IP address and port for the iManager server
<i>Show Novell iManager Information Dialog</i>	Prompts you for the URL to the iManager server after you select <i>Tools > iManager</i> . If the URL is missing or incorrect, iManager is unable to launch.

23.3.6 Import/Deploy

The Import/Deploy preference window contains five tabs: *Deploy*, *Import*, *Export*, *Prompts*, and *Trace*. The following table describes their options.

Table 23-18 Preferences: Deploy

Setting	Description
<i>Show the summary dialog prior to performing a deployment</i>	Displays a summary of what is going to be deployed. The default is On.
<i>Replace driver set/server associations when deploying a driver set</i>	If you want driver set and server associations when deploying, select this option. The default is Off.
<i>Always deploy both drivers of an eDir-to-eDir connection when SSL/TSL is enabled</i>	With this option selected, you are prompted to deploy both sides on the connection. With both drivers deployed, Deploy is integrated with the creation of eDir-to-eDir certificates, if the certificates are created in Designer. Deploy adheres to the settings set in <i>Preferences > Designer for IDM > Configuration > eDir-to-eDir</i> . The default is On. This is the recommended setting.
<i>Restart running drivers after deploying the driver</i>	Restarts the driver after it is deployed. The default is On.

Table 23-19 Preferences: Import

Setting	Description
<i>Perform prompt checking when running a driver configuration file</i>	Displays the <i>Do you wish to perform all mandatory and required prompt checking when running this Driver configuration file?</i> prompt. If you select <i>Yes</i> to the prompt, you must then enter information in required fields while configuring the driver. If you select <i>No</i> , you temporarily disable this setting and can skip required fields.

Setting	Description
<i>Include application schema when importing drivers</i>	Imports the eDirectory application schema when you select this option. You might not want to import all the associated data. The default is Off. See Section 15.4.3, "Importing a Schema," on page 288.
<i>Show the summary dialog prior to performing an import</i>	Allows you to view what's being imported in a summary screen. The default is On.

Table 23-20 *Preferences: Export*

Setting	Description
<i>Copy cross driver policy references into exported configuration files</i>	Selected by default, this option saves you the trouble of manually inputting cross-driver policy references.

Table 23-21 *Preferences: Prompts*

Setting	Description
<i>Show dialog to export cross driver policy references to configuration files</i>	Selected by default. If you do not want to see a dialog box about these references, deselect the option.
<i>Show a warning dialog when overwriting a driver set/server association</i>	Warns that the driver set being deployed has a different server association than the server that you are about to deploy to. The association in the deployed driver set overwrites the existing server association.
<i>Show the dialog box to deploy both drivers in an eDir-to-eDir system when SSL/TLS is enabled</i>	This is the default, and it is also the recommended setting. With this option selected, you are prompted to deploy both sides on the connection.
<i>Show the dialog box to restart drivers after a deployment</i>	Selected by default. If you do not want to see a dialog box about these references, deselect the option.

Table 23-22 *Preferences: Trace*

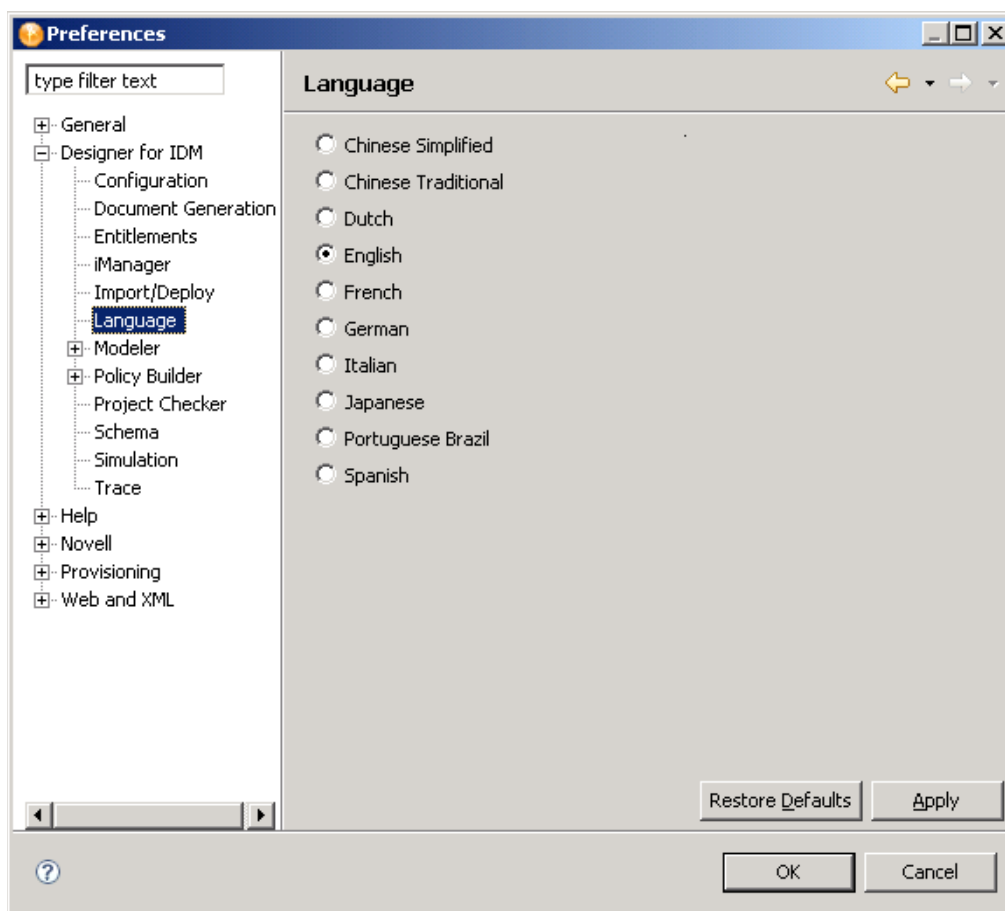
Setting	Description
<i>Trace import and deploy event information</i>	Deselected by default. If you need to troubleshoot an import or a deploy, select this option, then open the Trace view to inspect the import or deploy.
<i>Generate debug messages for the Driver configuration prompt dialog box</i>	Deselected by default. If you need to generate debug messages, select this option.
<i>Show verbose debug messages</i>	Deselected by default. If you need to generate verbose debug messages, select this option.

Setting	Description
<i>Time import and deploy operations</i>	Deselected by default. If you need to time how long it takes to import or deploy an object, select this option.

23.3.7 Changing the Language Setting

When you installed Designer, you selected a language to display Designer's UI. This setting enables you to change the language.

- 1 Select *Window > Preferences > Designer for IDM > Language*.



- 2 Select a language, then click *OK*.

The *Apply* and *OK* buttons set the language and prompt you to restart Designer.

Restore Defaults reads the `config.ini` file, detects the previous language setting, and then defaults to that setting. When the changed property is written back to the `.ini` file, all comments are removed from the file. To preserve these comments, Designer copies the original `config.ini` to `config.ini.bak` and uses the backup to determine the default setting.

- 3 Restart Designer.

23.3.8 Modeler

- ♦ “Modeler Tabs” on page 461
- ♦ “Dataflow” on page 464
- ♦ “Palette” on page 465

Modeler Tabs

- ♦ “Behaviors” on page 461
- ♦ “Display” on page 461
- ♦ “Guidance” on page 462
- ♦ “Layouts” on page 462
- ♦ “Pages” on page 463
- ♦ “Prompts” on page 464
- ♦ “Themes” on page 464



Behaviors

Table 23-23 Preferences: Modeler Behaviors

Setting	Description
<i>Auto-create servers when connecting a driver to a different driver set</i>	Automatically creates a server for a driver set when you connect a driver to a different driver set.
Launch the driver Properties dialog box	Launches the driver’s Properties page.
Show the driver’s Policy Flow view	Displays the driver’s Policy Flow diagram in the Outline view.

Display

Table 23-24 Preferences: Modeler Display

Setting	Description
<i>Show labels by Applications and Identity Vaults (Architect. mode)</i>	Shows labels below applications (in both modes) and above Identity Vaults (Architect mode only).
<i>Show driver icons in Developer mode</i>	Displays a driver icon  on the line that represents a driver in the Modeler.
<i>Show password icons in Developer mode</i>	Displays a password sync icon  below a driver icon in the Modeler.
<i>Auto-expand Identity Vaults to fit contents</i>	Causes Identity Vaults to expand to accommodate objects that you place in them.
<i>Auto-shrink Identity Vaults to fit contents</i>	Causes Identity Vaults to shrink when you remove objects from them.

Setting	Description
<i>Auto-size Identity Vaults to fit their titles</i>	Enables vaults to expand horizontally, to accommodate long titles. Otherwise, the titles concatenate after approximately 20 characters.
<i>Grid Width</i>	Increases or decreases cells in the Modeler's grid. To access the grid, select the Modeler, then click <i>View > Grid</i> .

Guidance

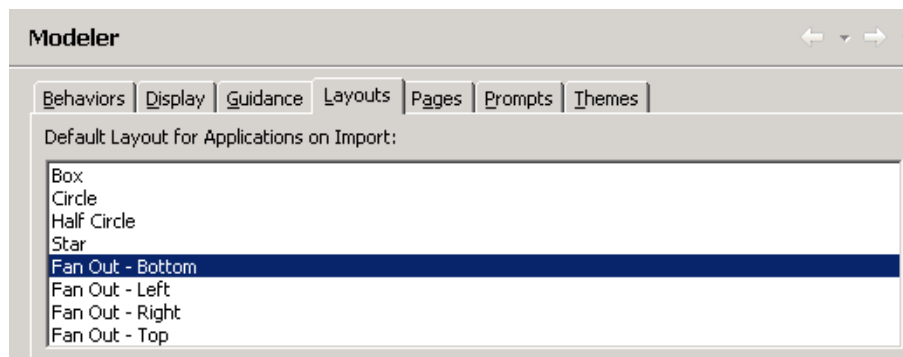
Table 23-25 *Preferences: Modeler Guidance*

Setting	Description
<i>If an Identity Vault doesn't already exist, one will be created when you drop the application</i>	Creates an Identity Vault when you drag or drop an application from the palette into the Modeler.
<i>eDir-to-eDir connection tip, when you've connected the same eDir app to two driver sets</i>	Prompts you to connect a line directly between the end driver sets when you set up an eDir-to-eDir relationship.
<i>Setting data flows in architect mode will default all policy and schema settings</i>	Sets policy and schema settings to defaults when you set data flows in architect mode. To edit the settings, use the Developer mode.
<i>Saving Dataflow to disk will first force a project save</i>	Requires you to save a project before you can save Dataflow to disk.

Layouts

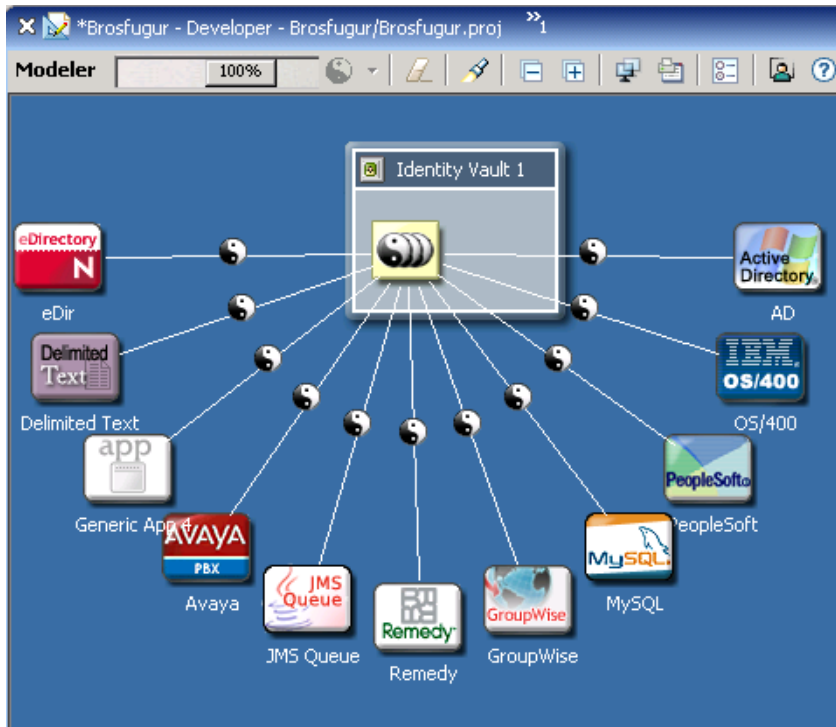
Layout settings determine how applications are displayed in the Modeler. To set a default layout when you import projects, select an option (for example, *Circle*).

Figure 23-12 *Preferences: Modeler Layouts*



The following figure illustrates a half-circle layout:

Figure 23-13 An Example Layout

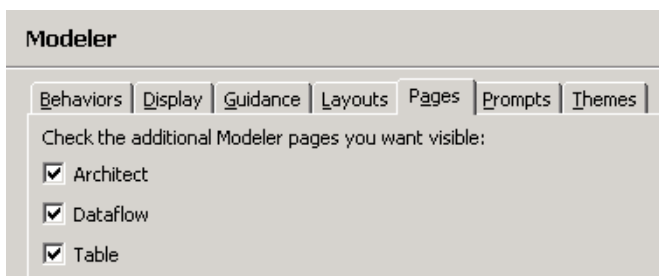


To arrange existing applications in a particular layout:

- 1 Right-click a Driver Set object in the Modeler.
- 2 Select *Arrange Applications*.
- 3 Select a layout.

Pages

Figure 23-14 The Pages Properties Page



Determines whether the Architect, Dataflow, and Table pages display as tabs at the bottom of the Modeler. By default, the Developer page (mode) appears in the Modeler. Displaying the additional tabs enables you to quickly toggle among pages.

Prompts

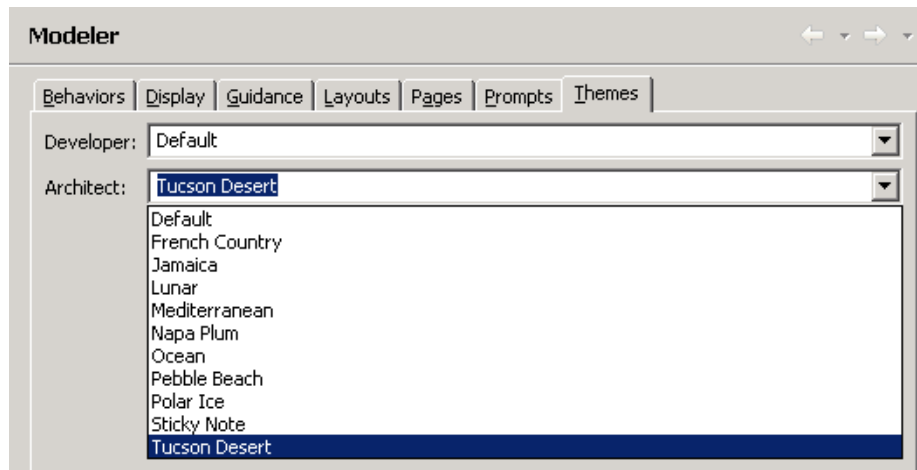
Table 23-26 Preferences: Modeler Prompts

Setting	Description
Show the Driver Config Wizard at connection time	Launches the Driver Configuration Wizard when you drag or drop an application in the Modeler.
Confirm when a driver is being deleted	Provides a Yes/No prompt for you to choose whether you want to delete the driver and its policies.

Themes

Defines themes for the Developer and Architect modes. Customizes the colors of the background, text, line, Domain Group background, and Domain Group title in the Modeler.

Figure 23-15 The Themes Property Page

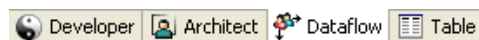


Dataflow

Specifies the number of columns per page that the Dataflow editor saves in the HTML reports.

To view or use the Dataflow editor, select the *Dataflow* tab in the Modeler.

Figure 23-16 The Dataflow Tab



Palette

Table 23-27 Preferences: Palette Settings

Setting	Description
<i>Arrange applications in folders</i>	Displays folders (for example, Database) in the palette and places applications in appropriate folders.
<i>Arrange applications in an alphabetical list</i>	Places all applications into one folder in the palette, and lists the applications alphabetically.

23.3.9 Policy Builder

Policy Builder creates and manages the policies in the drivers.

Table 23-28 Preferences: Policy Builder

Setting	Description
<i>Expand all rules when the Policy Builder is loaded</i>	Automatically expands rules in the Rules pane when you open the Policy Builder.
<i>Show version/author/last changed information</i>	Displays fields in the Rule Inline Editor when you double-click a rule. When you enter data in the fields, the comments are added to the policy.

logonHours: Convert to Login Allowed Time Map form

Name

Description

Author

Version Last Changed

Policy Description

The *Policy Description* allows you to add comments about the policy.

Figure 23-17 Preferences: Policy Description

Policy Description

Convert selected attributes to a form most commonly used in the Identity Vault.

Policy Builder Source

Table 23-29 Preferences: Policy Description Settings

Setting	Description
<i>Expand the Policy Description field</i>	Automatically expands the <i>Policy Description</i> field. You can hide the field by selecting the check box.
<i>Number of rows of text to display</i>	Determines how many rows to display in the <i>Policy Description</i> field. The default is 3.
<i>Policy Description position on the page</i>	Places the <i>Policy Description</i> field above or below the Rules pane.

23.3.10 Project Checker

Allows you to validate your project.

Table 23-30 Preferences: Project Checker

Setting	Description
<i>Limit Visible Items to</i>	Allows you to limit the number of items displayed in the Project Checker. The default value is 100.
<i>Prompt me to save the editor before running Project Checker</i>	Allows you to receive a prompt asking you to save your project before running the Project Checker. By default, this is enabled.

23.3.11 Schema

Allows you to manage the Identity Vault and connected system's schema.

Table 23-31 Preferences: Policy Builder > Schema

Setting	Description
<i>Warn when LDAP names are different from eDirectory names during .ldif import/export</i>	Allows you to turn off this warning prompt, which appears during the import or export of the schema.
<i>Warn when exporting base classes to .ldif</i>	Allows you to turn off this warning prompt, which appears during the export of the schema.
<i>Show the information message for the Manage Application Schema context menu</i>	Allows you to turn off the information message that appears when managing the application schema.

23.3.12 Simulation

The Simulation setting enables you to test a policy.

Table 23-32 Preferences: Simulation

Setting	Description
<i>Directories: Input</i>	<p>The directory that stores the input document. To simulate (test), you must have or generate an input document. Designer provides several predefined input documents. They are in subdirectories (for example, Add) in the <code>Designer\ eclipse\ plugins\ com.novell.designer.idm.policy_version</code> directory.</p> <p>You can specify a different location to store the input documents.</p>
<i>Directories: Output</i>	<p>The directory that stores the output document. When you select <i>Save As</i> while testing a document, Designer stores the file in an output directory. By default it is the <code>Designer\ eclipse\ plugins\ com.novell.designer.idm.simulation</code> directory.</p> <p>You can specify a different location to store the output documents.</p>
<i>Directories: Java Extensions</i>	<p>Enables you to simulate policies that contain references to external Java extensions. Specify the directory where the jar file is located, then copy the jar file containing the Java class to that directory.</p> <p>You can specify multiple Java extensions.</p>
<i>Options: Clear the policy simulation log file prior to performing a simulation</i>	<p>Automatically clears the log file. If you don't enable this setting, Designer displays a <i>Clear Log</i> button that you can use. If you do many simulations in succession, you might want to disable this option. The log file then captures and displays the events of all the tests, until you click <i>Clear Log</i>.</p>
<i>Options: Show the information prompt when a query is generated</i>	<p>Displays a prompt when the simulator generates a query. It simulates what the engine would do when a query is required to process the policy.</p>
<i>Options: Enable unsupported and experimental pre-release functionality</i>	<p>Enables a live version of the Policy Simulator.</p> <hr/> <p>WARNING: This feature is not supported and it is not documented.</p> <hr/>

23.3.13 Trace

The Trace view is useful in the following situations:

- ◆ To trace internal errors and messages, so that you can find out why something might not work as expected.
- ◆ To provide information for Novell Support, engineers, or other consulting resources.

All Designer-specific trace messages go to the Trace view if this view is open. Otherwise, no trace messages are sent.

Warnings and error messages are sent to the `.log` file, found in the run-time workspace metadata directory. Use the Error view to view this information.

Table 23-33 *Parameters: Trace*

Setting	Description
<i>Enable tracing</i>	Writes events to the Trace view. By default, tracing is off. To increase performance, disable tracing when you don't need it.
<i>Include stack traces</i>	Provides separate traces. Dumps the entire stack where an internal exception occurs, so that you can see in the code where the internal exception is failing.
<i>Include XML processor traces</i>	Provides separate traces that detail all of the processing of XML documents. This trace can become quite verbose.
<i>Show plug-in names in the trace</i>	In the Trace view, displays names of plug-ins where tracing has occurred. This is useful if you are tracing more than one plug-in.
<i>Show view when tracing</i>	Automatically brings up the Trace view if a trace is trying to be logged. By default, this setting is On.
<i>Trace buffer size</i>	Increases the buffer to show more characters. As the buffer increases in size gets higher, performance might degrade, depending on your system.
<i>Plug-Ins to Trace</i>	Lists all Designer plug-ins (in their simple name form). Select plug-ins that you want to trace.
<i>Select All</i>	Enables tracing in all Designer plug-ins.
<i>Deselect All</i>	Disables tracing in all Designer plug-ins.

23.4 Preferences: Help

Table 23-34 Preferences: Help

Setting	Description
<i>Specify how help information is displayed: Use external browser</i>	If an embedded Web browser is supported on your system, the help window uses that browser to display help contents. To force help to use an external browser, enable this option. To select an external browser, use the Web Browser preference page.
<i>Open window context help</i>	Determines whether the window context help opens in a dynamic Help view or in a pop-up window.
<i>Open dialog context help</i>	Determines whether the dialog box context help opens in a dynamic help section of the Help view or in a pop-up window.
<i>Help view document open mode</i>	Determines whether the documents selected in the Help view open in place or in the editor area.
<i>Search: Show all potential hits/Show only actual hits</i>	Determines how quickly the search is completed.

- ◆ [Section 23.4.1, “Help Server,” on page 469](#)

23.4.1 Help Server

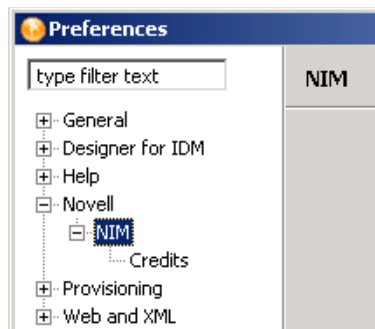
The Help system includes an internal server that serves help contents to the browser. Use the following settings to change the interface and the port that the server users. Change these settings only if you experience problems and can't view help by using the default settings.

Table 23-35 Help Server Settings

Setting	Description
<i>Host name or IP address</i>	The name or address of a local IP interface that the server uses. Either leave this field blank or point it to your own IP address.
<i>Server port</i>	The IP port for the server to listen on. If you don't assign a port, the operating system assigns one.

23.5 Preferences: Novell

Figure 23-18 Novell NIM



Provides legal notices for Novell Integration Manager (formerly exteNd Composer™), parts of which are included in a Designer plug-in.

23.6 Preferences: Provisioning

For information on provisioning settings, see [Setting Provisioning View Preferences \(http://www.novell.com/documentation/idm35/dgpro/index.html?page=/documentation/idm35/dgpro/data/palprovprefs.html\)](http://www.novell.com/documentation/idm35/dgpro/index.html?page=/documentation/idm35/dgpro/data/palprovprefs.html).

23.7 Preferences: Web and XML

The Web and XML preferences let you specify how Designer should handle editing and creation of CSS, HTML, and JavaScript* files.

- ◆ [Section 23.7.1, “CSS Files,” on page 470](#)
- ◆ [Section 23.7.2, “HTML Files,” on page 474](#)
- ◆ [Section 23.7.3, “JavaScript Files,” on page 479](#)
- ◆ [Section 23.7.4, “Task Tags,” on page 482](#)
- ◆ [Section 23.7.5, “XML Catalog,” on page 482](#)
- ◆ [Section 23.7.6, “XML Files,” on page 484](#)

23.7.1 CSS Files

The CSS Files preferences allow you to specify how Designer displays and manages CSS files.

Figure 23-19 CSS Files

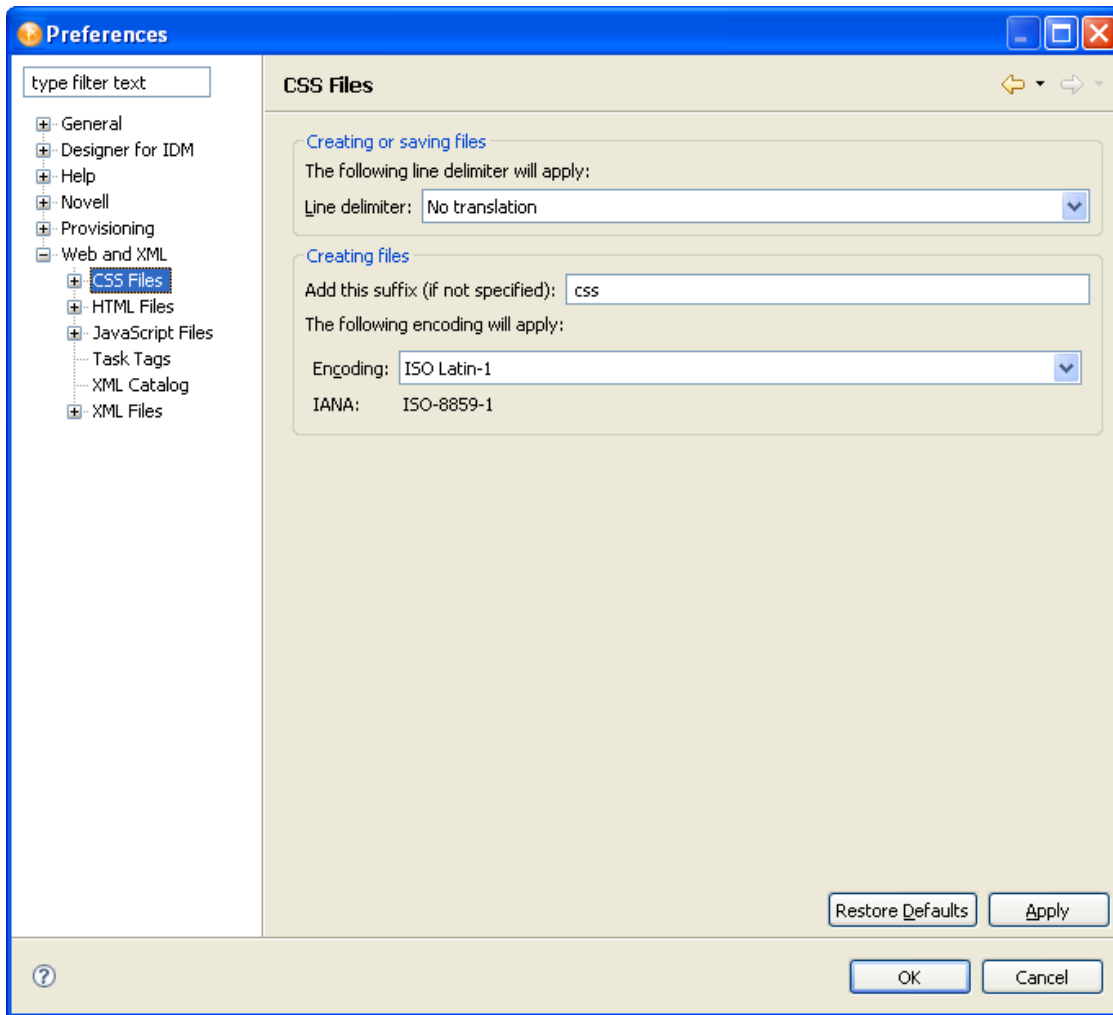


Table 23-36 CSS Files

Preference	Description
<i>Creating or saving files: Line Delimiter</i>	Sets the type of line delimiter applied to newly created files. The options are: <ul style="list-style-type: none"> ◆ Windows ◆ Unix ◆ Mac ◆ No translation
<i>Creating files: Add this suffix</i>	The file suffix the editor adds when creating files. The default is CSS.
<i>Creating files: Encoding</i>	The encoding used for new files.

- ◆ “CSS Source” on page 472

- ◆ “CSS Styles” on page 473

CSS Source

Figure 23-20 CSS Source

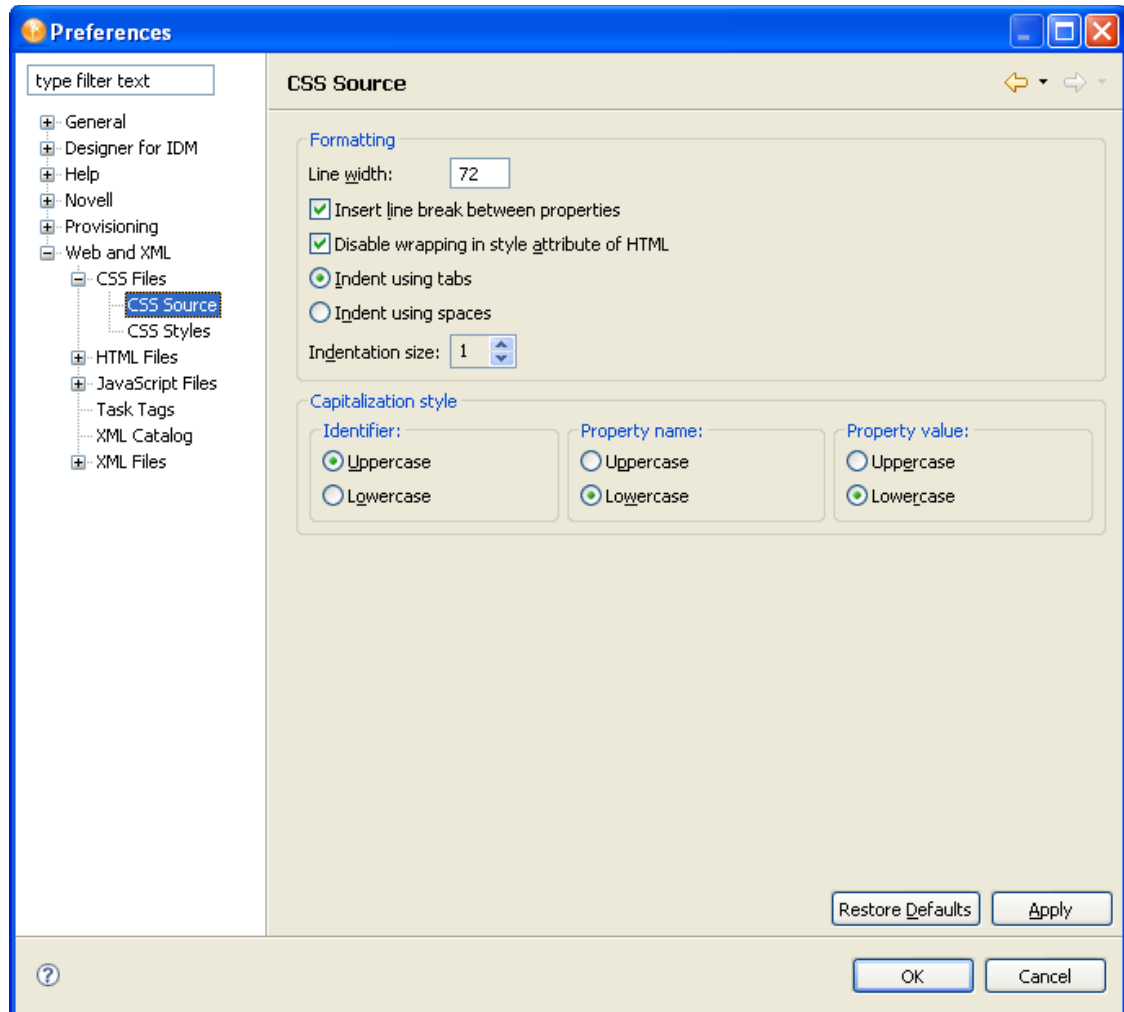


Table 23-37 CSS Source

Setting	Description
<i>Formatting: Line width</i>	Specifies the number of characters in a line.
<i>Formatting: Insert line between properties</i>	Specifies whether the editor should insert a line between the CSS properties.
<i>Formatting: Disable wrapping in style attribute of HTML</i>	Specifies whether the HTML editor (used in the e-mail notification template editor) should allow wrapping of the value of a style attribute.

Setting	Description
<i>Formatting: Indent using tabs or Formatting Indent using spaces and</i>	Specifies how the first line of text indents.
<i>Formatting: Indentation size</i>	Specifies the size of the indent.
<i>Formatting: Capitalization style</i>	Specifies the default case for identifiers, property names, and property values.

CSS Styles

Figure 23-21 CSS Styles

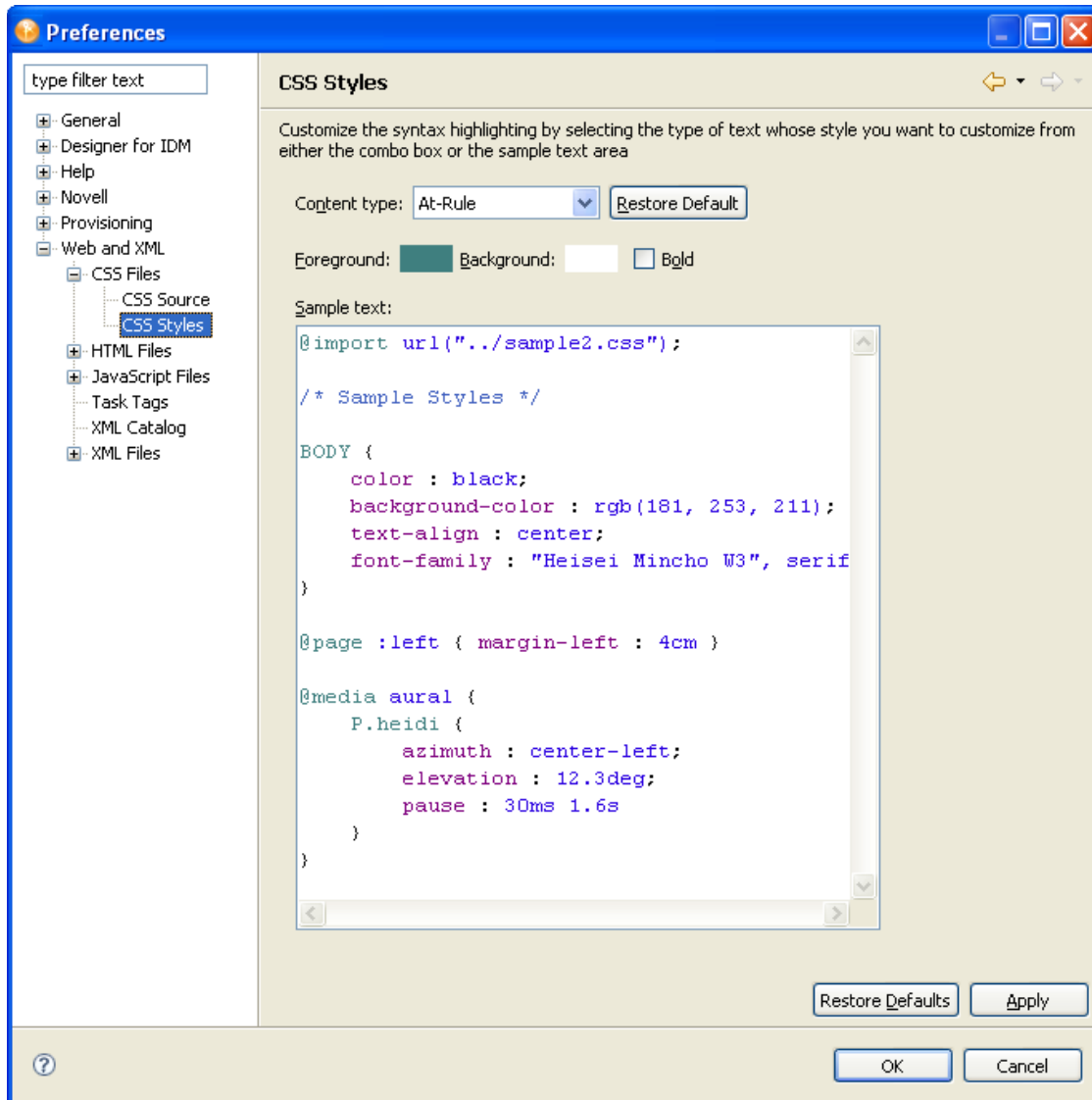


Table 23-38 CSS Styles

Setting	Description
Content type	Choose the content type for which you want to define a style.
Foreground/Background/Bold	Specifies the syntax highlighting (foreground and background color) and whether the text should be bold for individual CSS constructs.

23.7.2 HTML Files

The HTML Files preferences allow you to specify how Designer displays and manages HTML files and content.

Figure 23-22 HTML Files

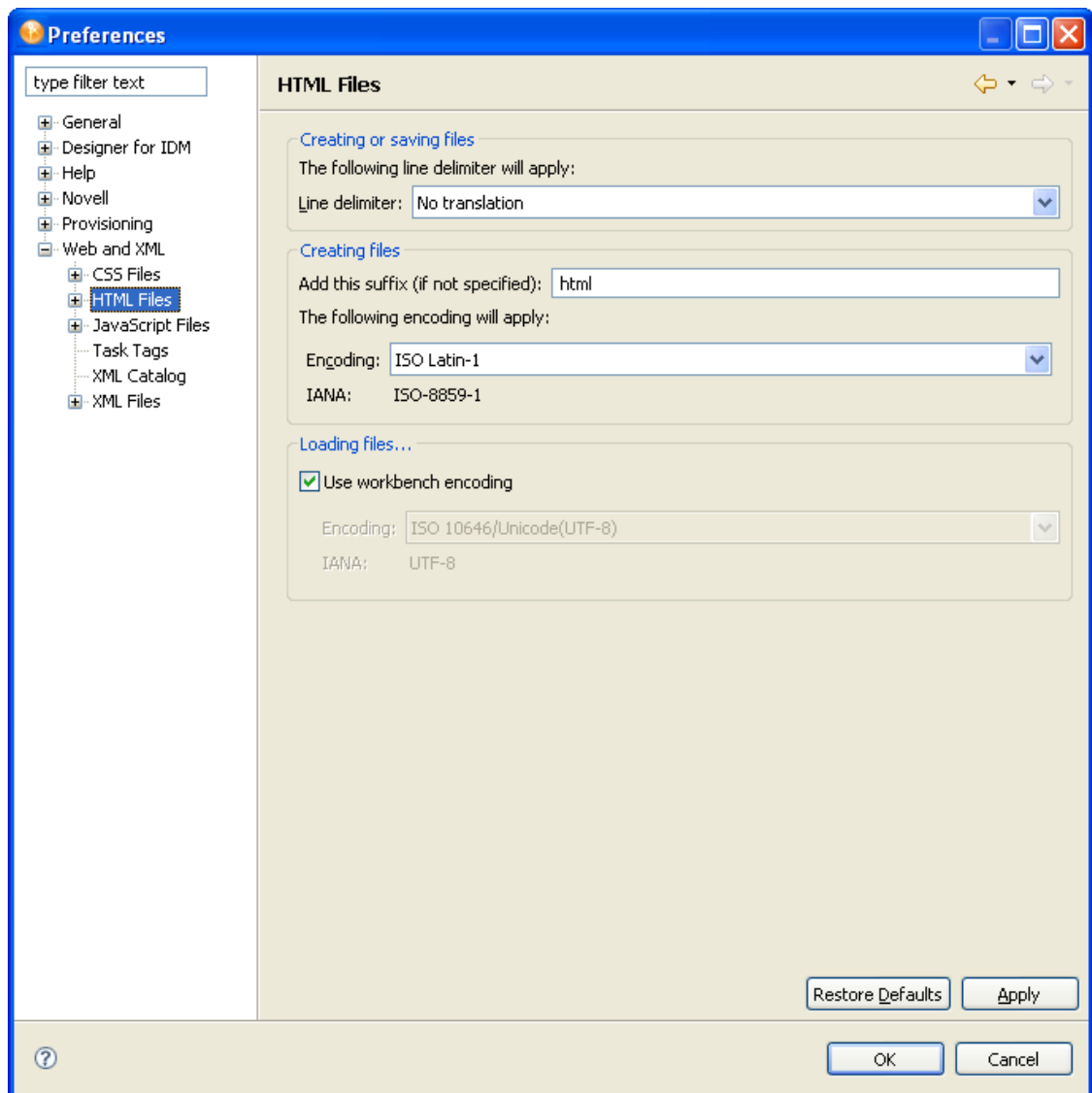


Table 23-39 *HTML Files*

Preference	Description
<i>Creating or saving files: Line Delimiter</i>	Choices are: <ul style="list-style-type: none">◆ <i>Windows</i>◆ <i>Unix</i>◆ <i>Mac</i>◆ <i>No translation</i>
<i>Creating files: Add this suffix</i>	Specifies the file suffix the editor should add when creating a new file. The default is html.
<i>Creating files: Encoding</i>	Specifies the editor's encoding for new files.
<i>Loading files</i>	Choose the encoding for files opened in the editor. Click <i>Use workbench encoding</i> to accept the default UTF-8, or select one from the list.

- ◆ [“HTML Source” on page 476](#)
- ◆ [“HTML Styles” on page 477](#)
- ◆ [“HTML Templates” on page 478](#)

HTML Source

Figure 23-23 HTML Source

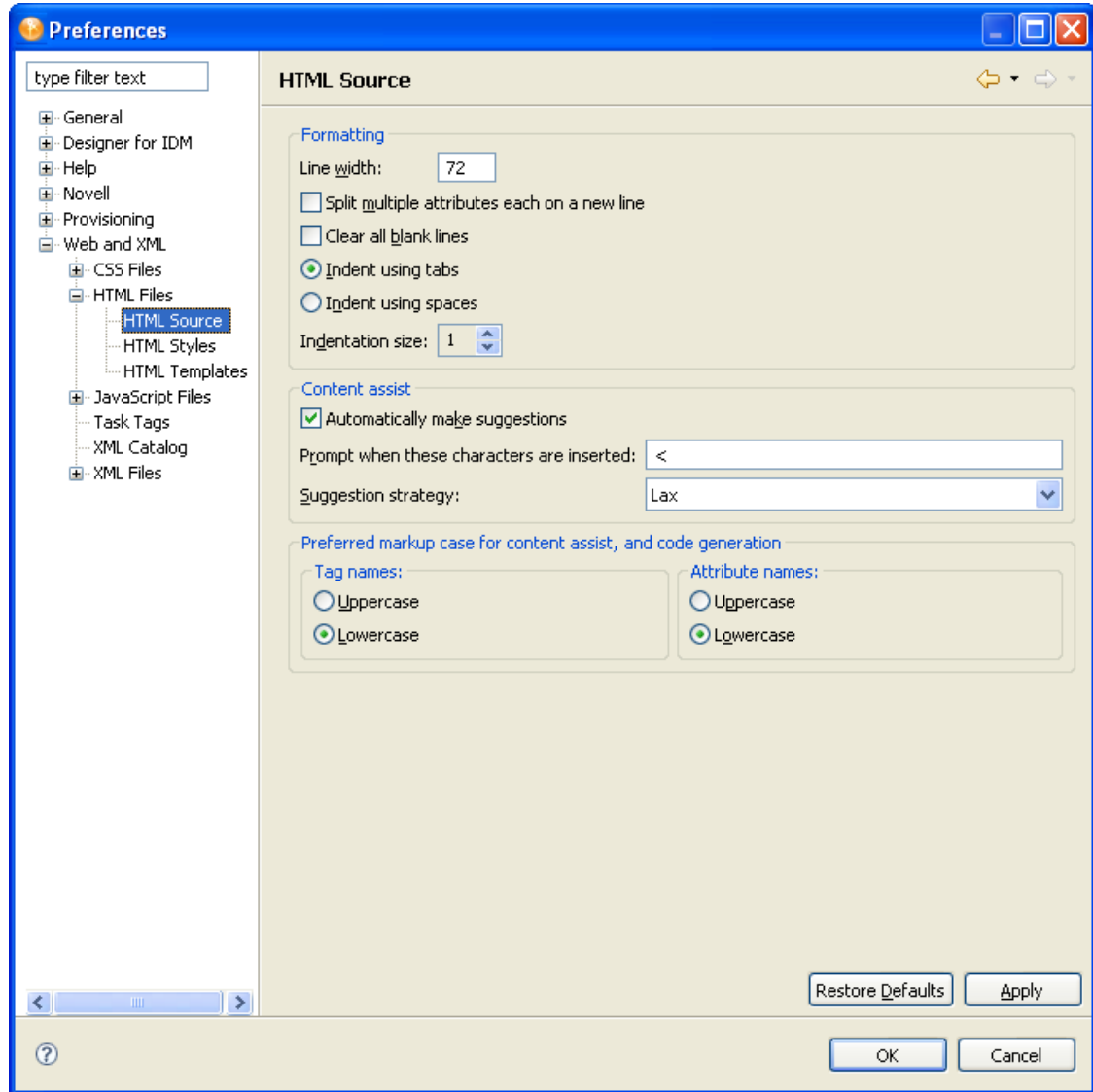


Table 23-40 HTML Source

Setting	Description
<i>Formatting: Line Width</i>	Specifies the number of characters for each line.
<i>Formatting: Clear all blank lines</i>	Specifies what the editor should do with blank lines
<i>Formatting: Split multiple attributes each on a new line</i>	Specifies what the editor should do with multiple attributes.
<i>Formatting: Indent</i>	Specifies whether the indent should be using tabs or spaces and also the indentation size.

Setting	Description
<i>Content assist: Automatically make suggestions</i>	Specifies whether to do automatic code completion.
<i>Content assist: Prompt when these characters are inserted</i>	Specifies the characters that initiate the content assist.
<i>Content assist: Suggestion strategy</i>	Specifies whether to use Lax or Strict grammar when making suggestions
<i>Preferred markup: Tag Names/Attribute Names</i>	Specifies if the editor's suggestions should be in uppercase or lowercase.

HTML Styles

Figure 23-24 HTML Styles

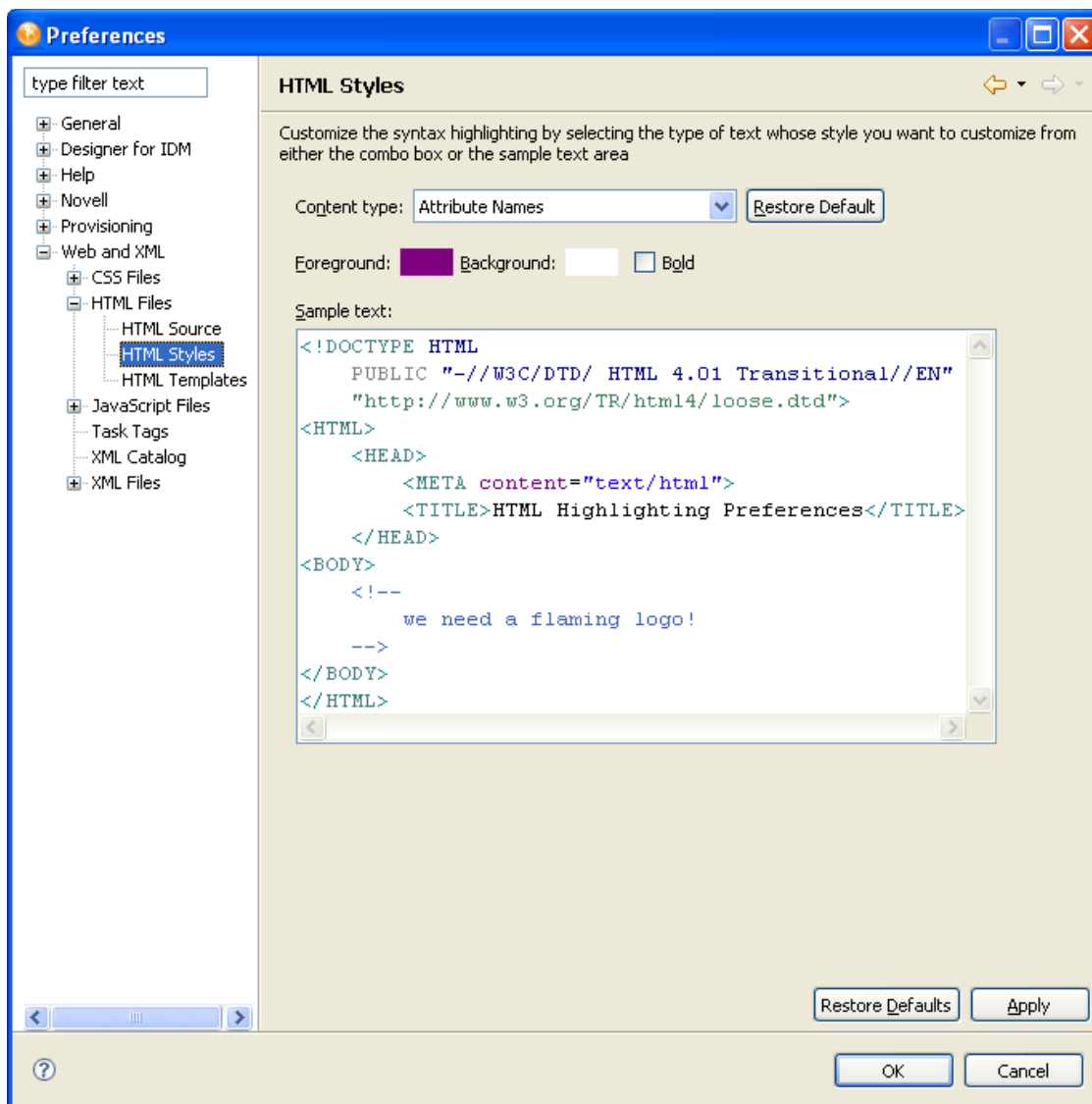


Table 23-41 HTML Styles

Setting	Description
Content type/Foreground/Background/Bold	Specifies how the editor performs syntax highlighting (foreground and background color) and bolding for the selected HTML content type.

HTML Templates

Figure 23-25 HTML Templates

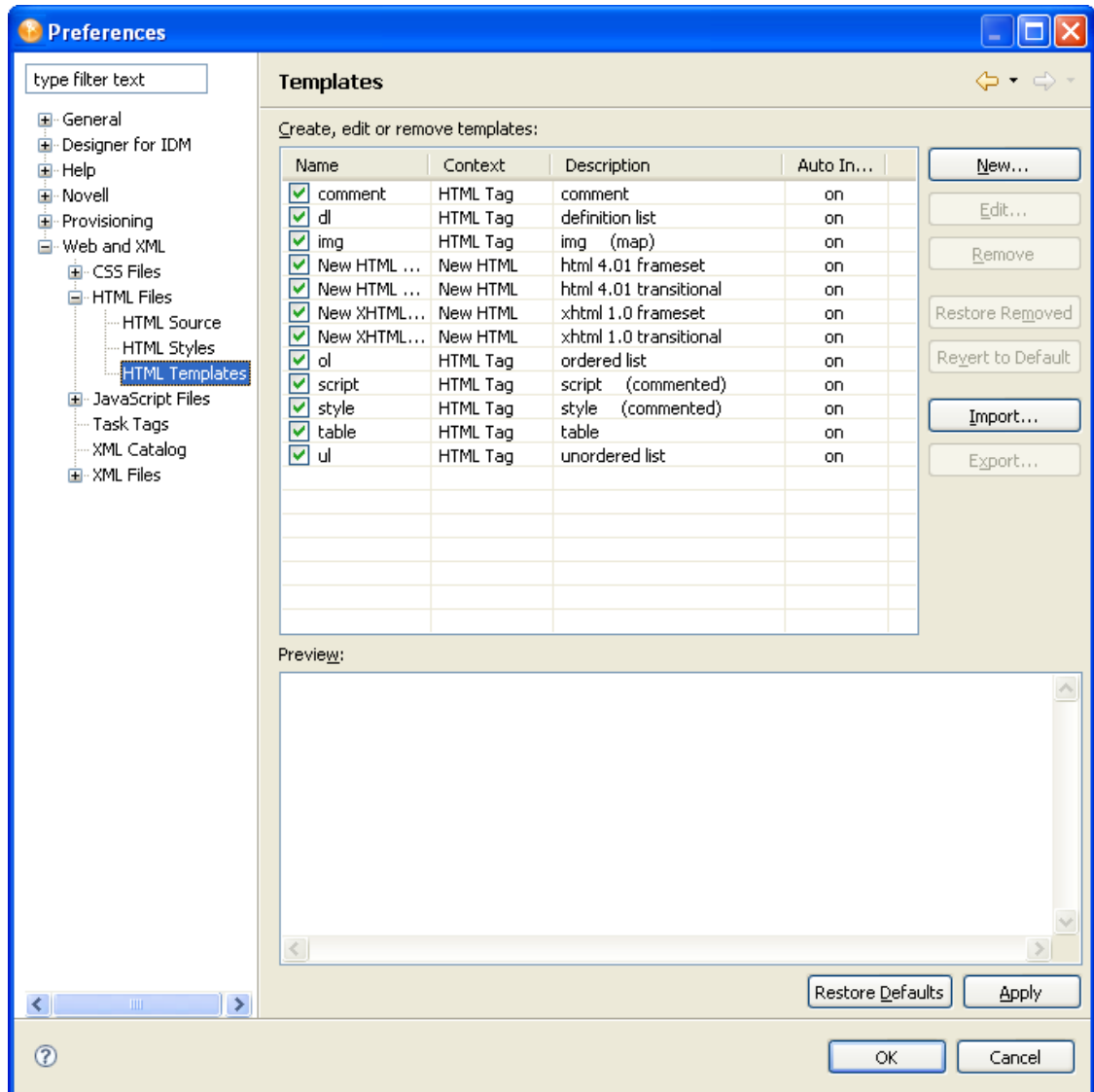


Table 23-42 HTML Templates

Setting	Description
Templates	The templates are used in the code completion in source editor. Use this preference to add, remove or edit templates.

23.7.3 JavaScript Files

Changes how JavaScript files are created.

Figure 23-26 JavaScript Files

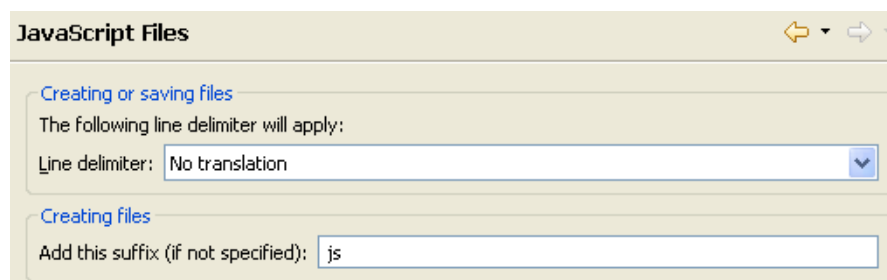


Table 23-43 JavaScript Files

Setting	Description
Creating or saving files: Line delimiter	Sets what type of line delimiter is applied to the file. The options are: <ul style="list-style-type: none">◆ No translation◆ UNIX◆ Mac◆ Windows
Creating files: Add this suffix (if not specified)	Sets the suffix for the file. The default value is <code>js</code> . It can be set to any value.

- ◆ [“JavaScript Source” on page 479](#)
- ◆ [“JavaScript Styles” on page 480](#)
- ◆ [“JavaScript Validation” on page 482](#)

JavaScript Source

Changes the formatting for the JavaScript files.

Figure 23-27 JavaScript Source

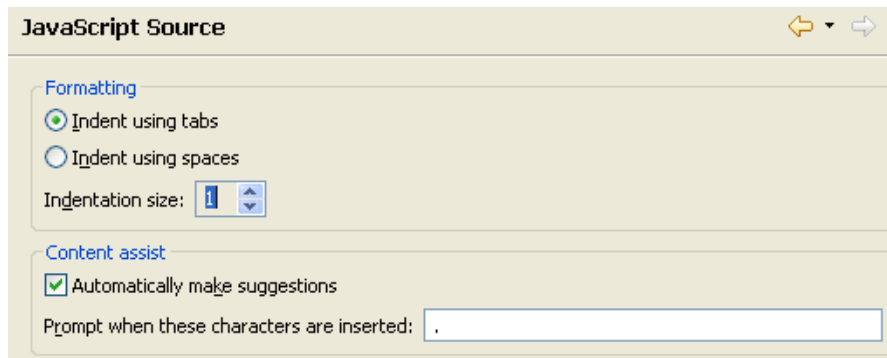


Table 23-44 JavaScript Source

Setting	Description
<i>Formatting</i>	Sets the formatting for the editor. The options are: <ul style="list-style-type: none">◆ <i>Indent using tabs</i>◆ <i>Indent using spaces</i>◆ <i>Indentation size</i>: Changes the indentation size by setting a numeric value.
<i>Content assist</i>	Helps with prompts when creating the files. <i>Automatically make suggestions</i> : Can be enabled or disabled if the check box is selected. <i>Prompt when these characters are inserted</i> : Allows you to receive prompts when the specified characters are entered.

JavaScript Styles

Customizes the content of the JavaScript files.

Figure 23-28 JavaScript Styles

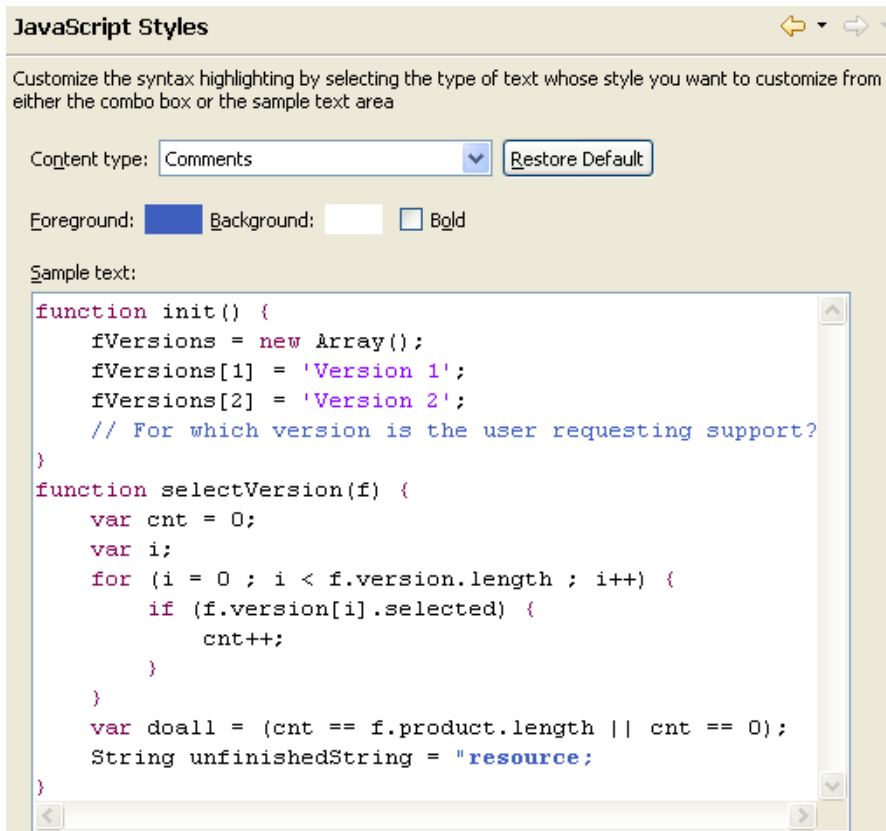


Table 23-45 JavaScript Styles

Setting	Description
<i>Content type</i>	<p>Selects the content to be customized. You can change each element independently. The sections that can be changed are:</p> <ul style="list-style-type: none"> ◆ <i>Comments</i> ◆ <i>Default Code</i> ◆ <i>Keywords</i> ◆ <i>Literal Strings</i> ◆ <i>Unfinished Strings and Comments</i> <p>If you select the element in the sample text field, the content type changes to what is selected.</p>
<i>Foreground</i>	<p>Displays the color that is set for the foreground. Double-click the color field to select another color.</p>
<i>Background</i>	<p>Displays the color that is set for the background. Double-click the color field to select another color.</p>
<i>Bold</i>	<p>Allows you to bold an element. Select the element, then click the <i>Bold</i> check box.</p>

Setting	Description
<i>Sample text</i>	Displays a sample file to see the changes.

JavaScript Validation

Allows the editor to validate the JavaScript as it is entered. To automatically validate the scripts, select *Automatically validate scripts*. Otherwise, the JavaScript is not validated.

23.7.4 Task Tags

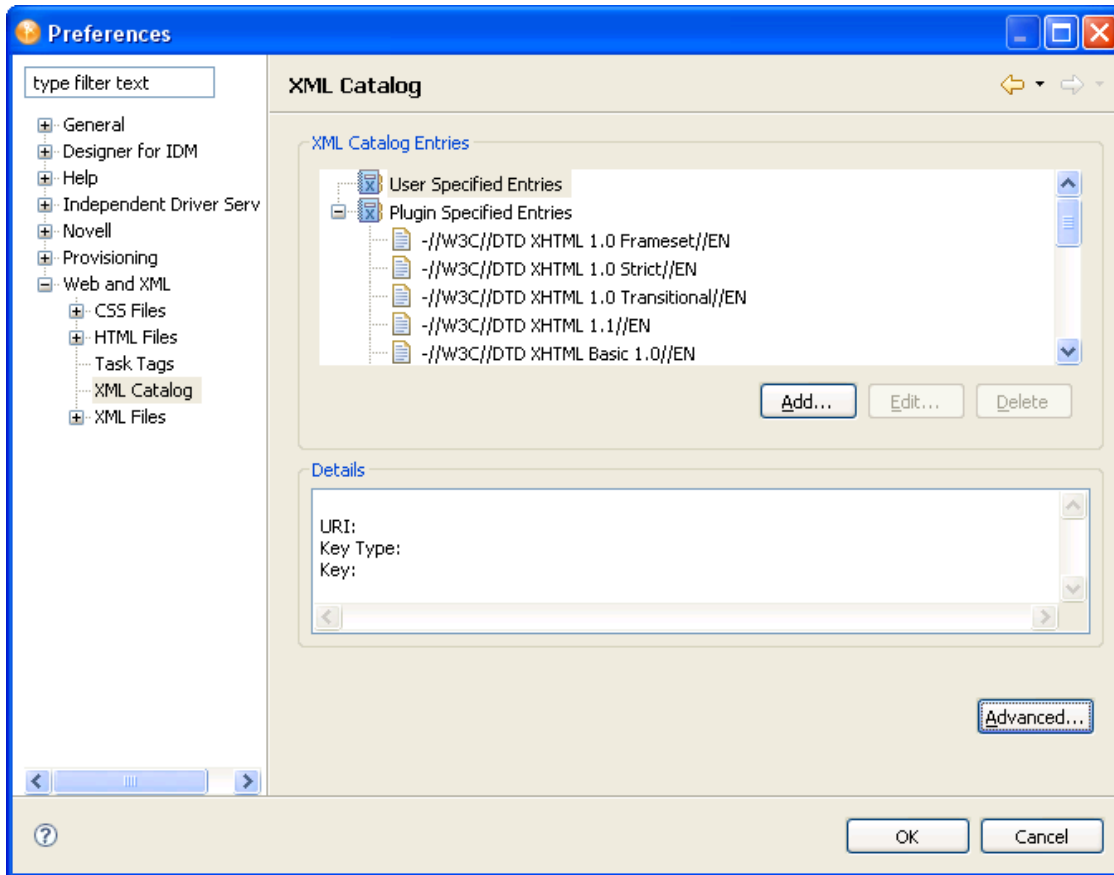
The Task Tag Preferences allow you to specify whether Designer should search for task tags and which task tags to display in the Task view. If *Enable searching for Task Tags* is not selected, Designer does not search for task tags. If you do not use the Task view, make sure this option is not selected because the search can affect Designer's performance.

The *TODO* and *FIXME* task tags are defined by default, but you can specify additional tags by selecting *New* and completing the Task Tag dialog box. Use the *Filters* tab to specify the file types that Designer should search for the selected task tags.

23.7.5 XML Catalog

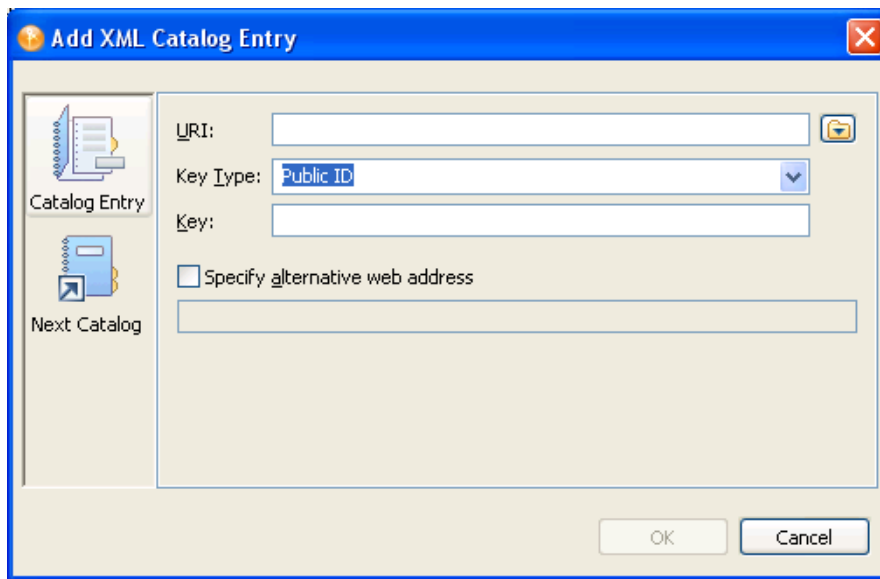
The XML Catalog Preferences allow you to manage the WST XML Catalog implementation. You can add, edit, or delete user-specified catalogs. You cannot use this preference to manage the plug-in specified entries. The XML editor uses the WST XML catalogs implementation to resolve XML schema and DTD references for associating URLs, system and public identifiers with URLs.

Figure 23-29 XML Catalog Preferences



To add a user-specified entry:

- 1 Click Add.



2 Fill in the fields as follows:

Field	Description
<i>URI</i>	Specify a location on disk or a URL of the schema or DTD.
<i>Key Type</i>	Specify the key type. Values are public identifiers for DTDs or URIs for XML schemas.
<i>Key</i>	Specify a unique key
<i>Specify alternative web address</i>	Optionally, specify an alternative Web address for locating the schema or DTD.

3 Click *OK* to save.

23.7.6 XML Files

You can set the following general XML File preferences:

Figure 23-30 XML File Preferences

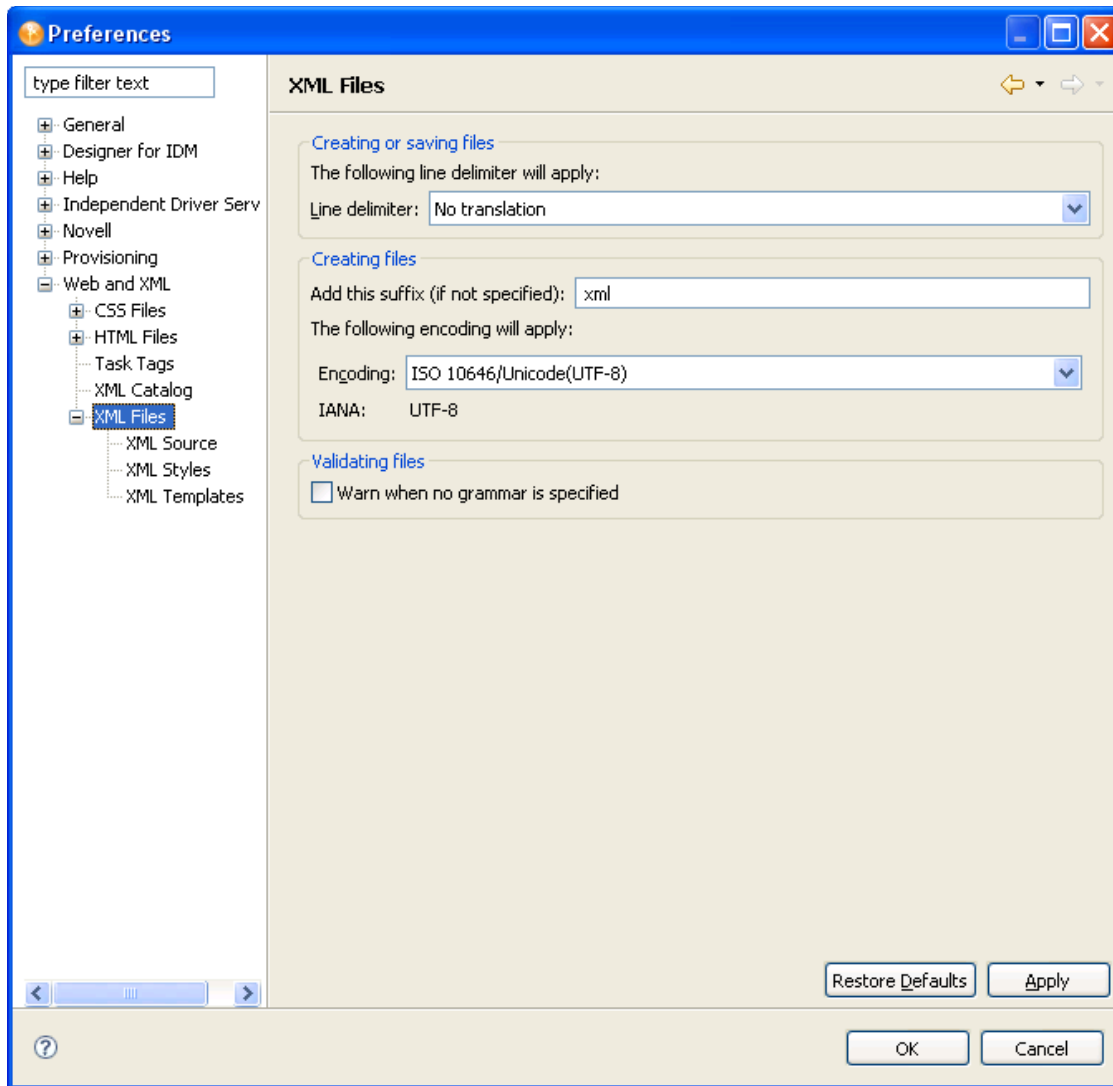


Table 23-46 XML Files Preferences

Category	Preference	Description
Creating or saving files	Line delimiter	Choices are:
		◆ <i>Windows</i>
		◆ <i>Unix</i>
		◆ <i>Mac</i>
◆ <i>No translation</i>		
Creating files	<i>Add this suffix</i>	The file suffix. The default is XML.
	<i>Encoding</i>	The encoding used by the editor
Validating files	<i>Warn when no grammar is specified</i>	Specifies whether to display a warning when no grammar (such as XML Schema or DTD) is associated with the XML document.

- ◆ [“XML Source” on page 486](#)
- ◆ [“XML Styles” on page 488](#)
- ◆ [“XML Templates” on page 488](#)

XML Source

You can set the following XML Source preferences:

Figure 23-31 XML Source

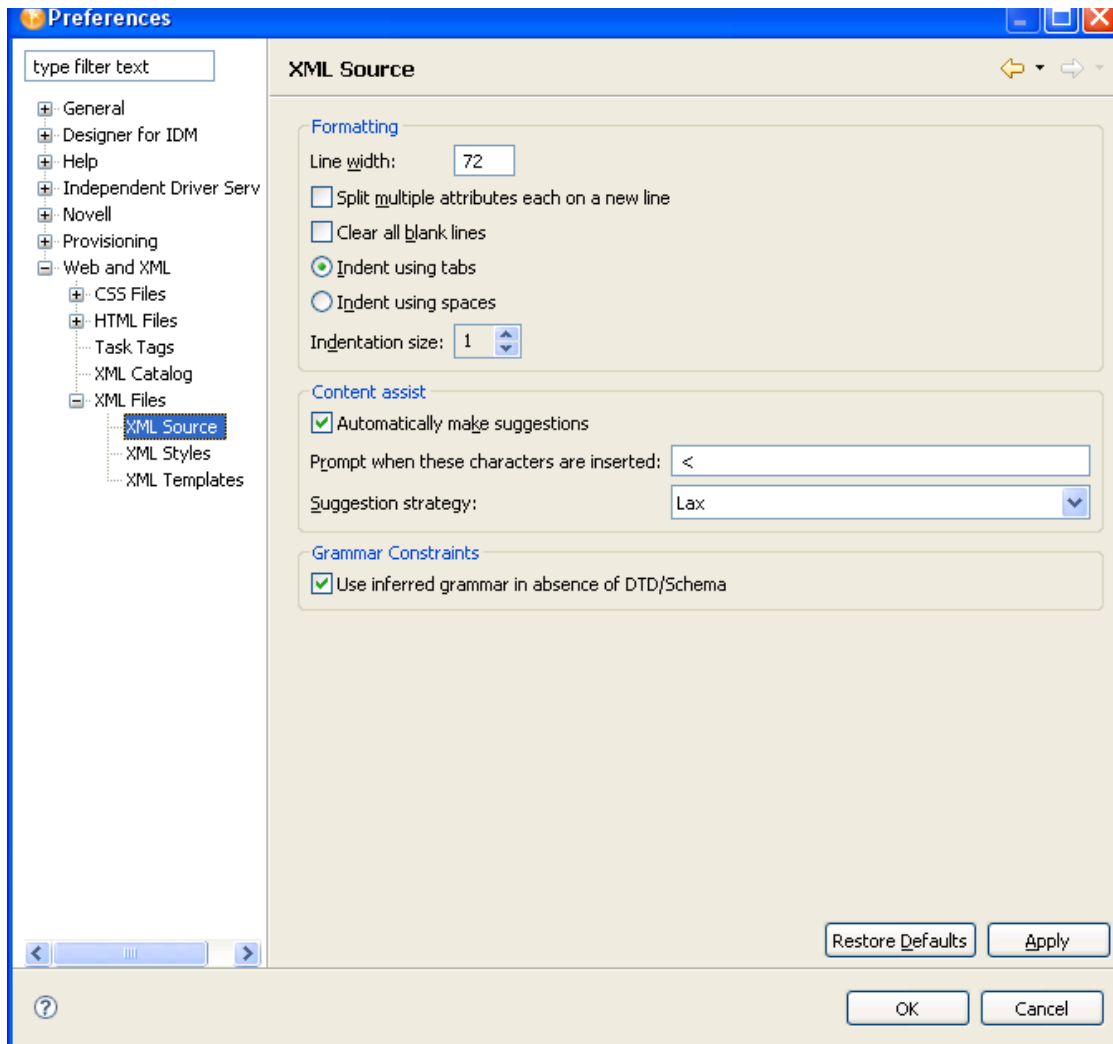


Table 23-47 XML Source Preferences

Category	Preference	Description
Formatting	Line width	Specifies the number of characters in a line.
	Split multiple attributes each on a new line	Specifies how attributes are formatted (whether to show each attribute on a separate line).
	Clear all blank lines	Specifies whether blank lines are removed when formatting.
	Indent using tabs/Indent using spaces, Indentation size	Specifies whether to use tabs or spaces as indentation and indentation size.

Category	Preference	Description
<i>Content Assist</i>	<i>Automatically make suggestions</i>	Specifies whether to do automatic code completion.
	<i>Prompt when these characters are inserted</i>	The list of characters that initiate code completion.
	<i>Suggestion strategy</i>	Specifies whether to use Lax or Strict grammar when making suggestions
<i>Grammar constraints</i>	<i>Use inferred grammar in absence of DTD/Schema</i>	Specifies whether to display code completion suggestions based on existing content of the XML document.

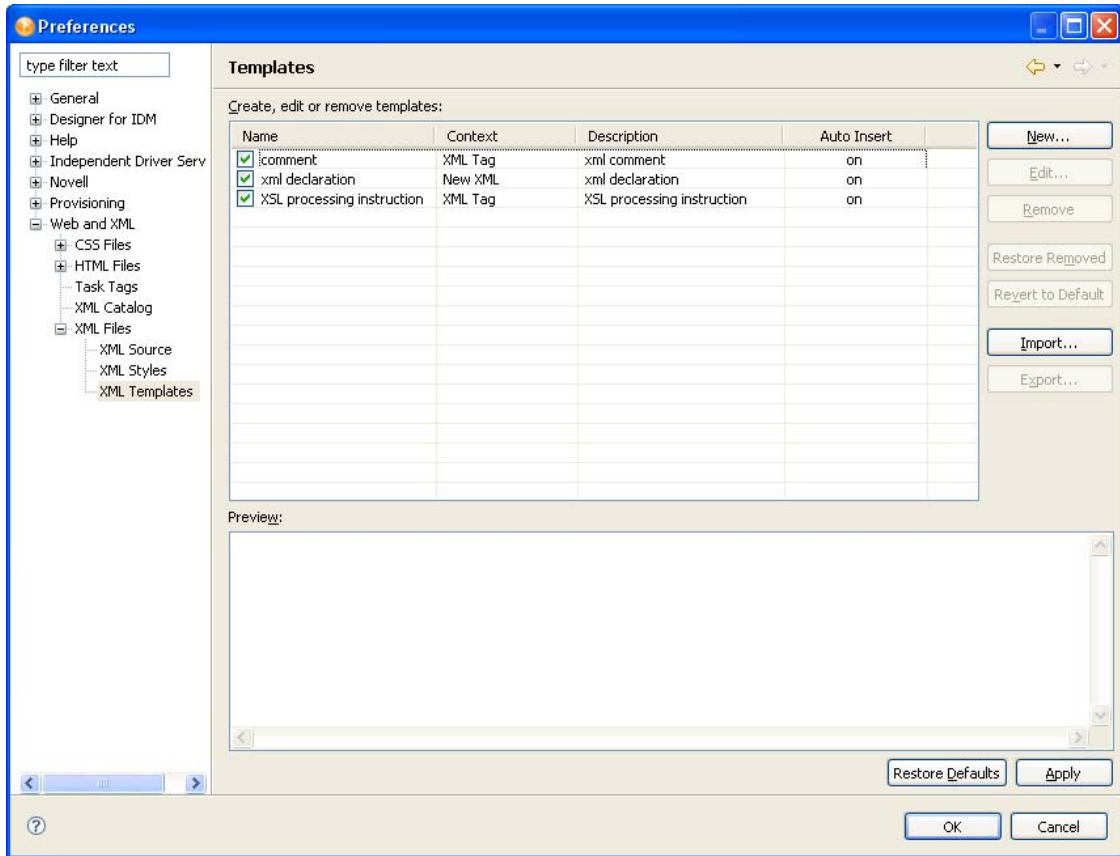
XML Styles

The XML Styles preferences allow you to specify the syntax highlighting (foreground and background color) and whether the text should be bold for individual XML constructs.

XML Templates

Use the XML Templates preference page to define XML templates. The templates are used in the code completion in the XML Source Editor. For example, selecting the `XSL Processing Instruction` template in the code completion inserts `<?xml-stylesheet type="text/xsl" href=""?>` in the source editor and places the cursor in the href value.

Figure 23-32 XML Templates Preference



- ♦ Section 24.1, “Viewing the Error Log,” on page 491
- ♦ Section 24.2, “Turning on Trace Messages,” on page 492
- ♦ Section 24.3, “Checking Loaded Plug-Ins,” on page 494
- ♦ Section 24.4, “Deploying Identity Manager Objects,” on page 495
- ♦ Section 24.5, “Issues When Generating Documents,” on page 509
- ♦ Section 24.6, “No F1 Help in Maximized Editors,” on page 513
- ♦ Section 24.7, “Freeing Heap Memory,” on page 513
- ♦ Section 24.8, “Reporting Bugs and Giving Feedback,” on page 514

24.1 Viewing the Error Log

The Error Log view can help you troubleshoot. If something isn't working, messages written to the error log might help you. The log is named `.log`. It is a hidden file.

To view the error log, you can use menus or browse the file system.

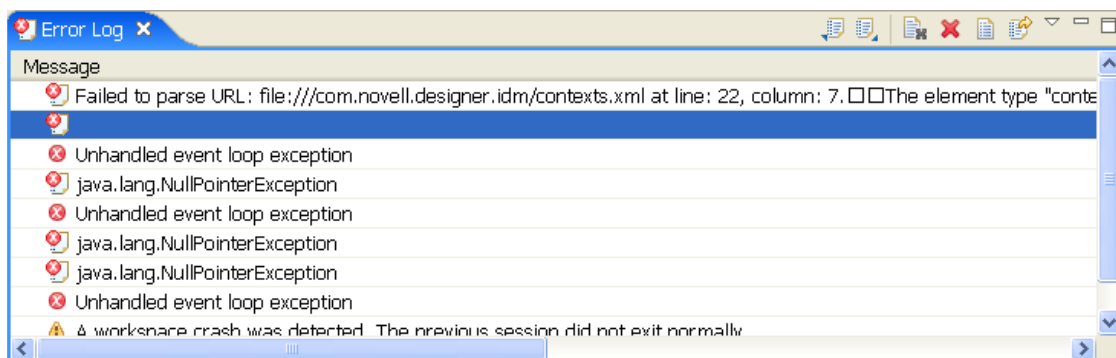
- ♦ Section 24.1.1, “Using Menus,” on page 491
- ♦ Section 24.1.2, “Browsing the File System,” on page 492

24.1.1 Using Menus

- 1 Select *Window > Show View > Other > PDE Runtime > Error Log*.
- 2 Click *OK*.

If you view the log through the application, a list of messages displays.

Figure 24-1 The Error Log



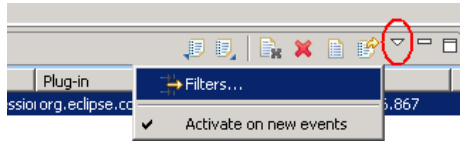
For details, double-click a message. (You can also right-click, then select *Event Details*.)

To sort messages, click the appropriate header bar.

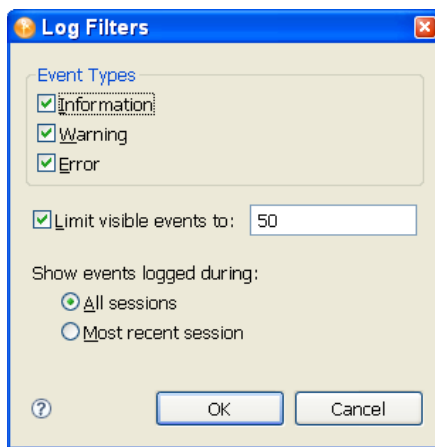
To use the Error Log view toolbar, right-click a message. The toolbar allows you to copy the log, clear the log viewer, delete, open, restore, export, import, and view the event's details.

To set filters for the types (and number) of messages to see:

- 1 On the Error Log view toolbar, click the filter icon.



- 2 Select *Filters*.
- 3 Customize the filter settings, then click *OK*.



24.1.2 Browsing the File System

- 1 Browse to your Designer workspace.

In Windows, the log file is typically in subfolders in the `/eclipse/workspace/.metadata` directory.

In Linux, the log file is typically in the Home directory, in the `/eclipse/workspace/.metadata` directory.

- 2 Open the log file.

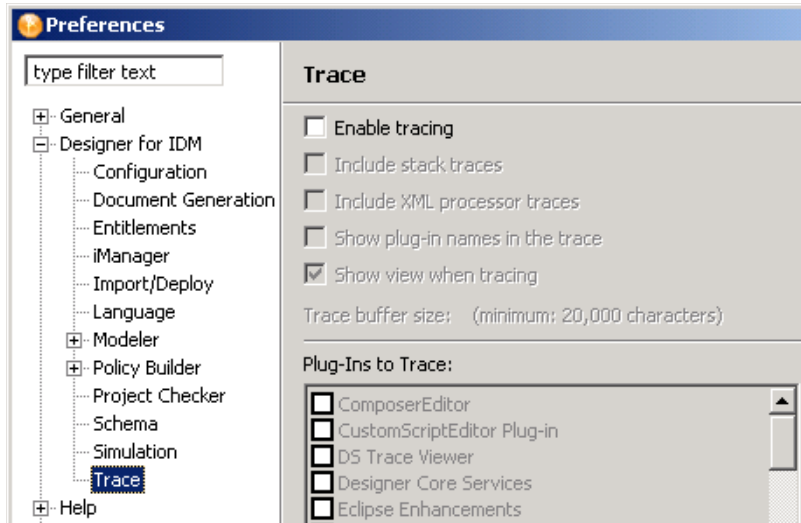
24.2 Turning on Trace Messages

You might want to send trace messages to the error log so that the messages are captured in a file. You can then easily e-mail the trace message to Novell[®] Support or others.

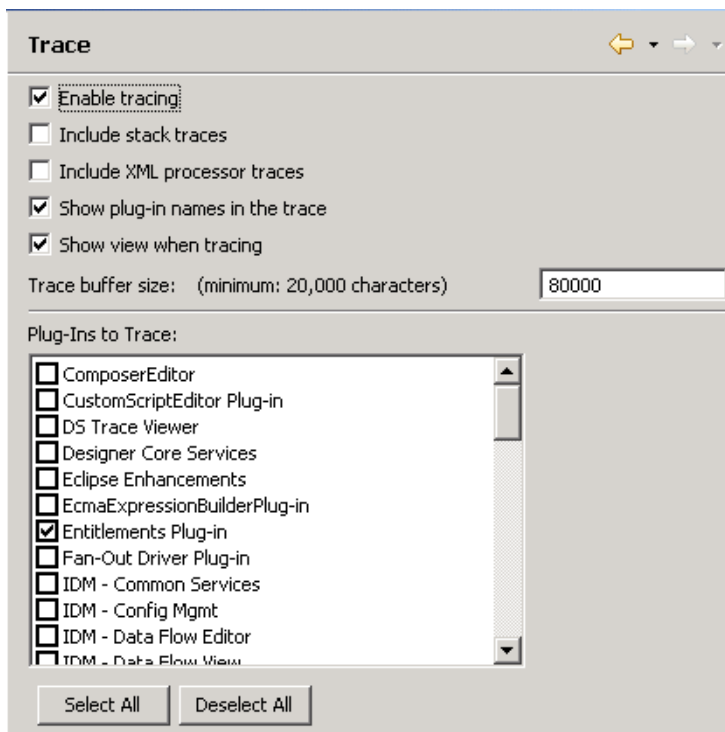
Programmers sometimes place hidden messages in their code so that if you are having problems, you can turn on the trace functionality and get additional insight. Even if you don't understand the hidden messages, they can help Novell Support diagnose the problem.

To get trace messages:

- 1 Click *Window > Preferences > Designer for IDM*.



- 2 Expand *Designer for IDM*, then click *Trace*.
- 3 Select *Enable tracing*, then select the options that you want to include or show.

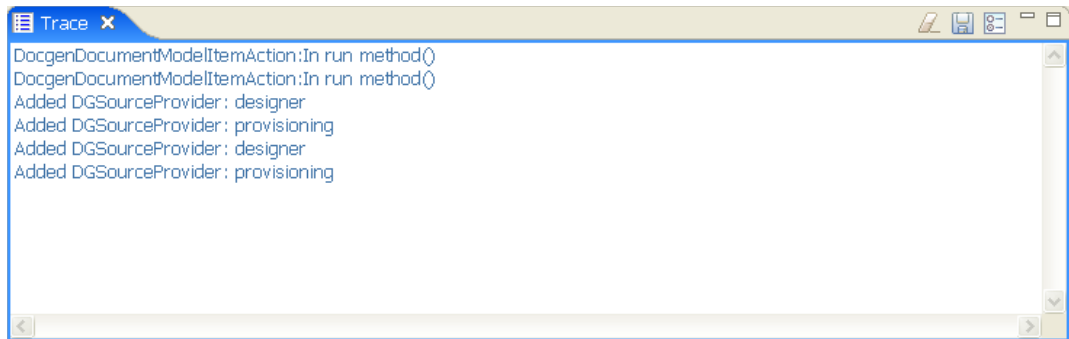



- 4 Select the plug-ins that you want to trace, then click *OK*.

To view the results of traces:

- 1 Select *Window > Show View > Trace*.

- 2 View data in the Trace view.

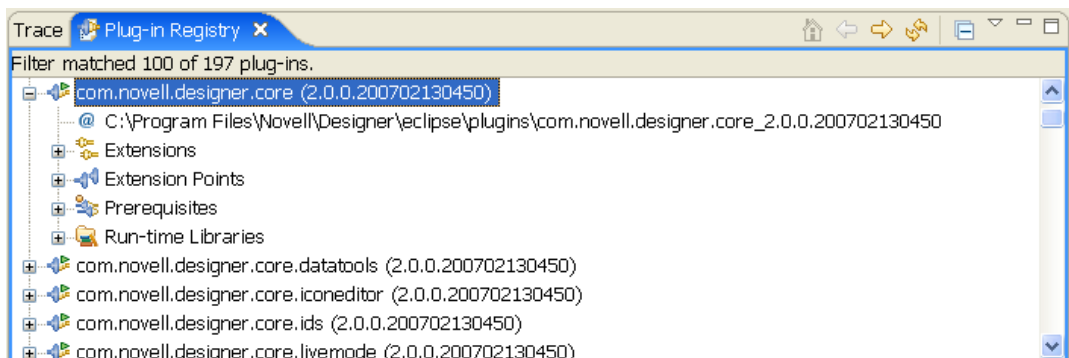


- 3 You can also turn on trace options from the Trace view by clicking the Preferences icon  in the Trace view.
- 4 Use the *Clear Trace* icon to clear the Trace view.
- 5 Use the *Save* icon to save Trace view contents to a .txt file.

24.3 Checking Loaded Plug-Ins

A problem can occur if a plug-in fails to load. To see which plug-ins are loaded:

- 1 Select *Window > Show View > Other*.
- 2 Open the PDE Runtime folder.
- 3 Click *Plug-in Registry > OK*.



The Plug-in Registry page lists the Designer plug-ins, which have a green triangle in the plug-in icon.

- 4 Use the *Home* icon to bring you to the top of the plug-in list.
- 5 Select a plug-in and use the *Right Arrow* icon to drill into the plug-in, and use the *Left Arrow* icon to return.
- 6 Use the *Refresh* icon to refresh the Plug-In Registry view.
- 7 Use the Plug-In Registry view toolbar to select *Show Active Plug-Ins Only*.

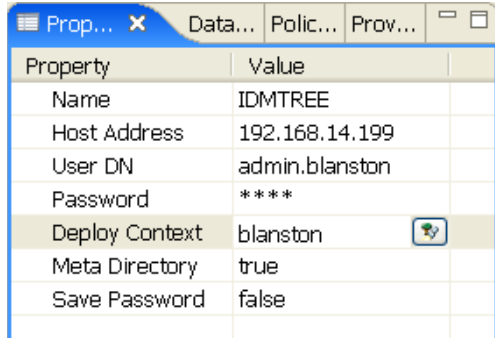
24.4 Deploying Identity Manager Objects

When you see an error message in Designer, the error message corresponds to the place where Designer could not complete the task, and indicates the best place to start troubleshooting. This section discusses the common problems you face when deploying Identity Manager objects into an eDirectory™ tree. To see error messages and possible solutions, see [Section 24.4.3, “Error Messages and Solutions,”](#) on page 498.

Deployment Considerations

- ◆ Ensure that the Metadirectory server meets the system requirements necessary to run Identity Manager. See Overview chapter in the *Identity Manager 3.5.1 Installation Guide* for requirements.
- ◆ Ensure that the Metadirectory server you are deploying to has Identity Manager installed and holds a real copy of the objects to which you want to synchronize. The server running eDirectory must have a Master Read-Write or a Filtered Read-Write replica.
- ◆ Ensure that the Java software installed on the server is running correctly, because Identity Manager is dependent on Java. If Java is corrupted, you might be able to deploy to a Metadirectory server but not run the Identity Manager drivers.
- ◆ To deploy an Identity Manager-based project or an object in a project, you must have access to the eDirectory tree that is associated with the Identity Vault you are designing. Select the Identity Vault you want to deploy, then look in the Properties view below the Project/Outline view.

Figure 24-2 *The Properties View*



Property	Value
Name	IDMTREE
Host Address	192.168.14.199
User DN	admin.blanston
Password	****
Deploy Context	blanston
Meta Directory	true
Save Password	false

In the Properties view, ensure that the Identity Vault’s Name, Host Address, User DN, Password, Deploy Context’s Distinguished Name (DN), and Metadirectory information is complete and accurate. (You can click the *Browse* button to find the Deploy Context’s DN on an existing tree if the other information is accurate and Designer can attach to the eDirectory tree.) You need this information to deploy anything, even a policy, into an existing eDirectory tree running the Metadirectory engine.

- ◆ Use the Deploy feature only after you have thoroughly tested the rules and policies that make up your drivers. To test a policy, use the Policy Simulator (right-click a policy and select *Simulate*, then click *Start* to see the simulation results of the policy that is being tested). For policy design, see the Policy Builder Help topics within the Designer utility.

You can use the Import feature to import a driver, a channel, or a policy. You can then modify the object or objects, run the Policy Simulator to ensure that the object is working correctly, then deploy the object back into the test tree for further analysis. You can also run the Compare

feature to see the differences between your modified driver and the driver that is currently running on an Identity Vault server.

- ◆ In the Outline view in the Project Group view, right-click the driver object in question (you can also double-click the driver object). Use the Properties window to make most changes to drivers. Properties are unique to each driver.

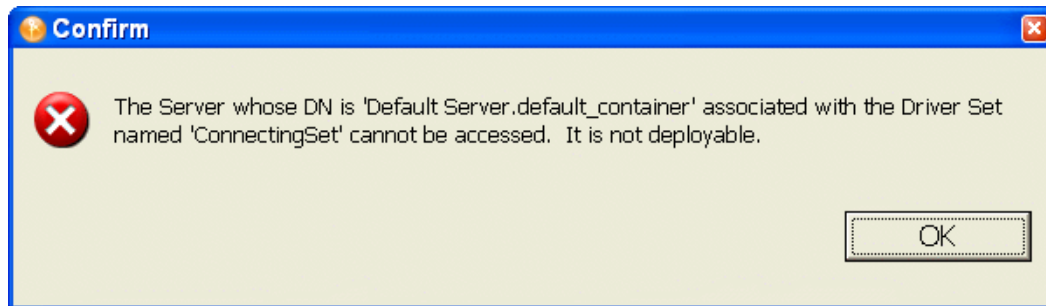
A simple driver problem is specifying the incorrect context (Distinguished DN) for an eDirectory tree. For example, the context of a user object in eDirectory is shown with the slash notation (for example, *Blanston\Sales\Users*) on the Properties of the Identity Manager driver or when you import the driver. However, different drivers can use formats other than the slash notation. For example, Active Directory and LDAP drivers use comma-delimited format (*OU=Users,OU=Sales,O=Blanston*). See the driver guides for further details on the drivers you are deploying.

- ◆ [Section 24.4.1, “An Example Deploy Error,” on page 496](#)
- ◆ [Section 24.4.2, “Deploying a Driver with an Empty Server List,” on page 497](#)
- ◆ [Section 24.4.3, “Error Messages and Solutions,” on page 498](#)

24.4.1 An Example Deploy Error

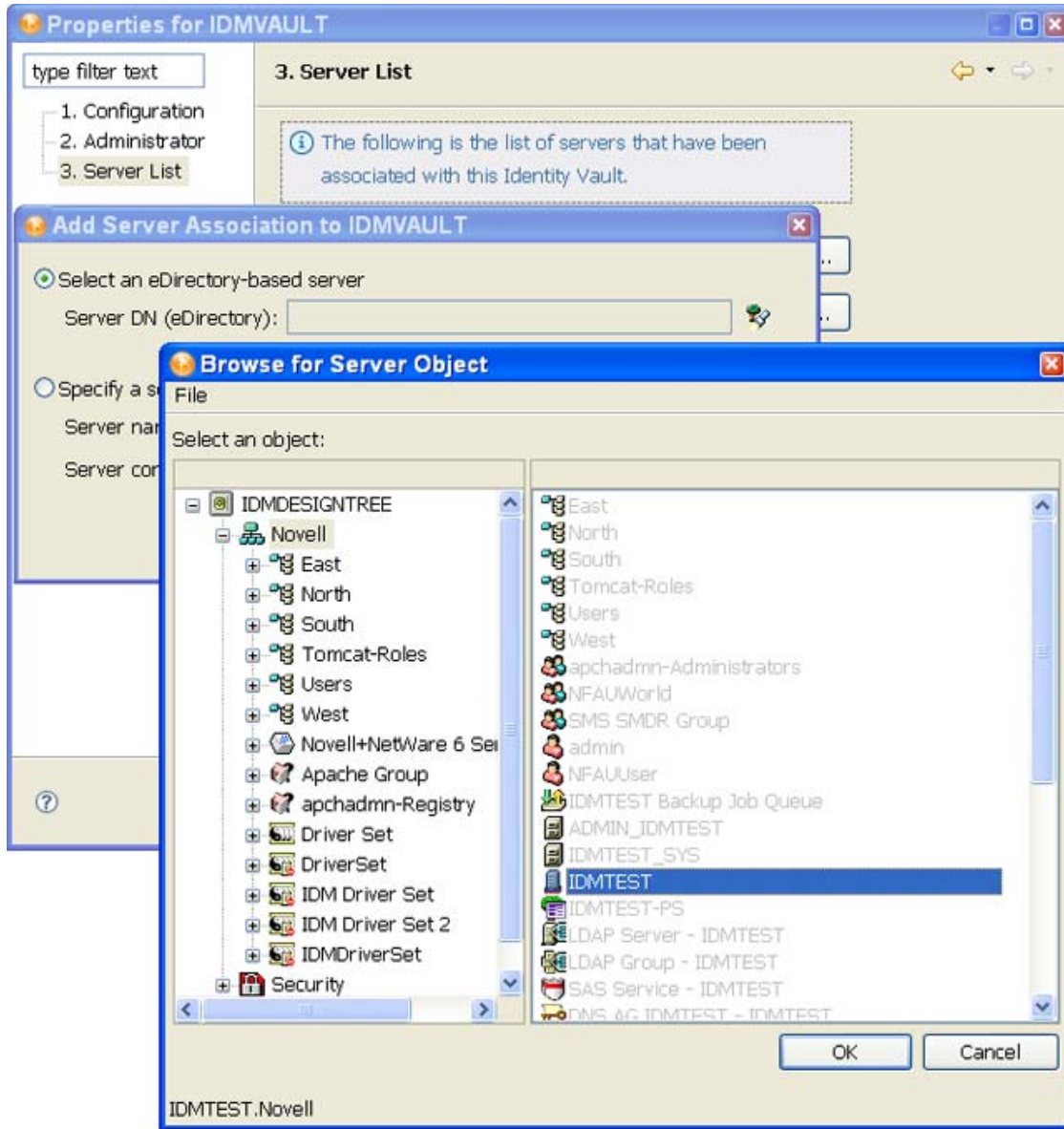
When deploying an Identity Vault for the first time, there are several common sources for errors, from incorrectly typing information to not completing the driver set templates.

Figure 24-3 *Default Server Container Message*



In this message, the server cannot be accessed because the server information is incorrect. Right-click the Identity Vault in the Modeler view, select *Properties > Server List*, then click the *Edit* button to edit the server information.

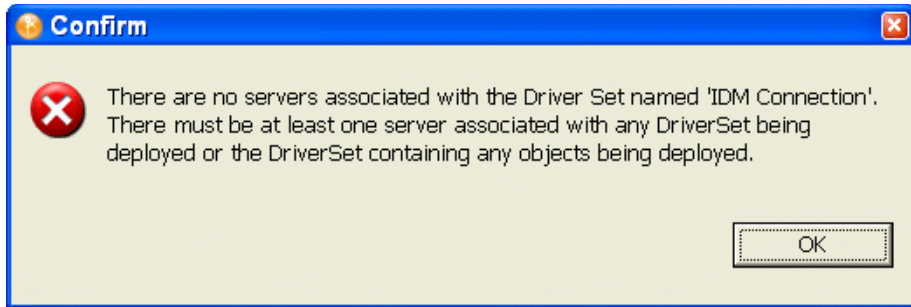
Figure 24-4 Correcting a Server Name Problem



24.4.2 Deploying a Driver with an Empty Server List

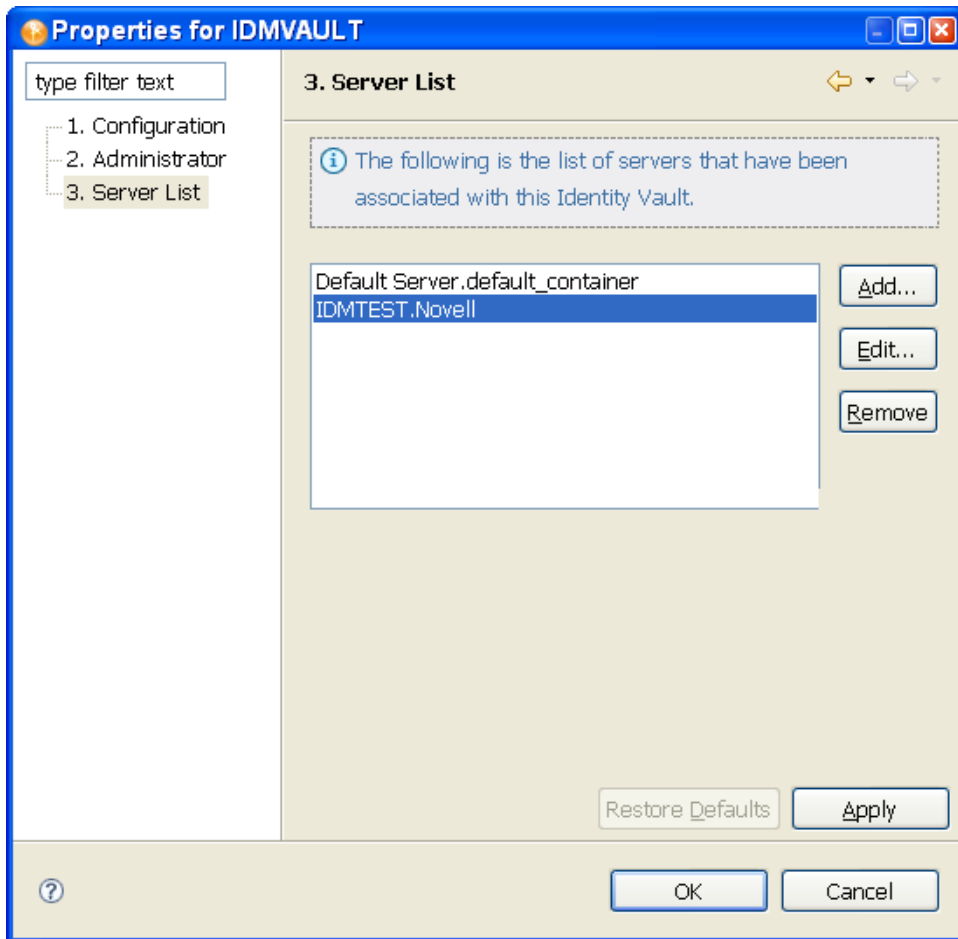
When deploying a driver set with an empty server list, you see the following error message:

Figure 24-5 No Server Association



To correct this problem, make sure the server list information is set correctly on the Identity Vault and driver set's Properties page.

Figure 24-6 Available Servers



24.4.3 Error Messages and Solutions

When you see an error message in Designer, the error message corresponds to the place where Designer could not complete the task and indicates the best place to start troubleshooting. This

section discusses the error messages you might see when deploying Identity Manager objects into an eDirectory tree, followed by their cause and possible solutions.

- ♦ “Identity Vault Configuration Errors” on page 499
- ♦ “Driver Configuration Errors” on page 499
- ♦ “Internal Designer Errors” on page 501
- ♦ “eDirectory Access Errors” on page 501
- ♦ “eDirectory Object/Attribute Creation Errors” on page 502
- ♦ “Warnings” on page 504

Identity Vault Configuration Errors

Cannot connect to host [Identity Vault Host]; verify the address is correct and that the server is running.

Possible Cause: The address listed in the Identity Vault properties is incorrect or the server is not running.

Solution: Verify that the server address is correct and that the server is up and running.

[User] could not be authenticated to [Identity Vault Host]. Cannot proceed.

Possible Cause: The user name or password listed in the Identity Vault properties is incorrect.

Solution: Verify the user name specified in the Identity Vault properties and reenter the user’s password.

Driver Configuration Errors

The driver configuration file [Driver Config File] is not a valid XML document: [Error Message].

Cause: The Driver Configuration file being imported from the file system does not contain a valid XML document.

Solution: Fix the Driver Configuration file format.

The XML contained the file named [Driver Config File] is not a driver configuration file. The file cannot be imported.

Cause: The Driver Configuration file being imported from the file system is a valid XML document but is not a valid driver configuration file.

Solution: Import a driver configuration file.

The following 'XML DOM Exception' was thrown.
[ExceptionInfo]

Cause: The Driver Configuration XML document is incorrectly formatted. This is probably an internal error because driver configuration files are dynamically generated by Designer for deployment.

Solution: Turn on trace for Designer. To do this, select *Window > Preferences > Designer for IDM Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*. Attempt to deploy again and send the trace file to Novell Support.

The following 'Number Format Exception' was thrown.
[ExceptionInfo]

Cause: An integer value in the driver configuration file being deployed is invalid. All integer fields in Designer should validate the content when it is set.

Solution: Turn on trace for Designer. To do this, select *Window > Preferences > Designer for IDM Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*. Deploy again and analyze the generated driver configuration file to see if all integer attribute values are correct. Identify the incorrect parameter in Designer, correct the setting, and redeploy. The specified driver configuration file does not contain a valid driver configuration.

Cause: Designer attempted to process a dynamically generated driver configuration file with an invalid format.

Solution: Turn on XML tracing for the Import/Deploy plug-in. To do this, select *Window > Preferences > Designer for IDM Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*. Deploy again and send the trace to Novell Support. Otherwise, edit and correct the configuration file being imported.

Tree population is not supported from a Driver Set configuration. Tree population components will be ignored.

Cause: The driver configuration file being processed has a `<ds-object>` element under a `<driver-set-configuration>` element, which is not permitted.

Solution: If this is a dynamically generated configuration file, contact Novell Support; otherwise, move the `<ds-object>` element under a `<driver-configuration>` element.

The following Driver Set based global variables could not be resolved:
[Global Variable List]

These variables exist in both the source and target Driver Sets. The two definitions, however, have different types.

Cause: The driver configuration file being processed has global variable definitions that could not be resolved.

Solution: If this is a dynamically generated configuration file, contact Novell Support. If it is a driver configuration file on disk, check the global variable definitions.

The driver configuration file being processed does not contain a valid driver configuration.

Cause: The driver configuration file being processed does not contain a `<driver-configuration>` element.

Solution: If this is a dynamically created configuration file, turn on XML tracing for the Import/Deploy plug-in. To do this, select *Window > Preferences > Designer for IDM Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*. Deploy again and send the trace to Novell Support. Otherwise, edit and correct the configuration file that is being imported.

The specified driver configuration file was only intended to be imported from a ConsoleOne command line.

Cause: The driver configuration file being processed is not a valid document.

Internal Designer Errors

An internal error has occurred in the Designer Data Model: The policy named [Policy Name] does not know its container.

Cause: The policy being deployed is not contained in a Channel or Driver object. This is an abnormal error, indicating that the Designer model has become corrupted.

Solution: Contact Novell Support.

eDirectory Access Errors

The following 'Component Creation Exception' occurred while trying to access eDirectory.

[Exception Info]

Cause: A value contained in the driver configuration file being deployed could not be successfully created in eDirectory. This is probably an internal error because driver configuration files are dynamically generated by Designer for deployment. However, if the Driver in Designer was created by importing a driver configuration file from the file system and that configuration file contained a Tree Population Segment, a value within a <ds-object> element might be invalid.

Solution: Turn on trace for Designer. To do this, select *Window > Preferences > Designer for IDM Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*. Deploy again and analyze the generated driver configuration file to see if any <ds-object> elements exist. If they do, verify that all attribute values are correct. If no <ds-object> elements exist or if all values seem to be correct, contact Novell Support.

The following 'IO Exception' occurred while trying to access eDirectory.

[ExceptionInfo]

Cause: This is a Java exception indicating that Designer could not perform the requested input or output operation.

Solution: Contact Novell Support.

DSAccessException:

[ExceptionInfo]

Cause: Designer could not connect to the target deployment server.

Solution: Verify that the server information specified in the Identity Vault properties page is correct and that the eDirectory server is up and running.

The following 'Namespace Exception' occurred while trying to access eDirectory. ({})

Cause: This is a namespace exception indicating that there is a problem with the eDirectory schema, such as a missing attribute or class.

Solution: Verify that the eDirectory schema being imported from or deployed to is correct. If the driver being deployed contains Tree Population segments, verify that the objects being created are valid for the target eDirectory schema.

An exception occurred during the deployment. Cannot perform the operation.

Cause: An unknown exception was encountered.

Solution: Contact Novell Support.

The following 'Snapin Exception' occurred while trying to access eDirectory.
[ExceptionInfo]

Cause: Snap-in exceptions can be thrown in certain methods to report exceptions or errors during import/deploy. Subclasses of a snap-in exception include:

- ◆ NotAContainerException: There was a call to get the children of an eDirectory object that is not a container.
- ◆ ObjectNotFoundException: The object being resolved cannot be found in eDirectory.
- ◆ SPIException: Unable to connect to the eDirectory tree.

Solution: The exception might include the name of the object that caused the exception. Verify that the eDirectory tree being imported or deployed to is up and running and that it has Identity Manager installed.

The following exception occurred but was not handled. ({0})

Cause: An unexpected error occurred while resolving an object in eDirectory.

Solution: Contact Novell Support.

eDirectory Object/Attribute Creation Errors

The driver could not be created.

Cause: Designer attempted to create a driver in eDirectory, but the process failed.

Solution: Verify that the target eDirectory server has Identity Manager installed.

A [ObjectClass] object named [ObjectName] could not be created.

Cause: Designer attempted to create a Publisher, Subscriber, or Policy object in eDirectory, but the process failed.

Solution: Verify that the target eDirectory server has Identity Manager installed.

The driver password could not be saved.

Cause: Designer attempted to set the Driver password in eDirectory, but the request failed.

Solution: Verify that the target eDirectory server has Identity Manager installed.

The password named '{0}' could not be saved.

Cause: Designer attempted to set a named password in eDirectory, but the request failed.

Solution: Turn on stack tracing for the Import/Deploy plug-in to get details of the exception. To do this, select *Window > Preferences > Designer for IDM Trace > Enable Tracing*. In the Trace window, select the check box for *Include Stack Traces*.

The value for the attribute named [Attribute Name] could not be stored on the object named [Object name].

Cause: Designer attempted to add an attribute to an object in eDirectory, but the request failed. The error message should contain information about the attribute and object.

Solution: Verify that the attribute and value are valid for the given eDirectory object type.

The value for the attribute named '{0}' could not be updated using the XSLT on the object named '{1}'.

Cause: Unable to export shim configuration information.

Solution: Contact Novell Support.

An exception was thrown updating the value of the [Attribute Name] attribute on the [Item Type] object named [Object Name].
[Exception Info]

Cause: Unable to deploy the Identity Manager object and attributes to eDirectory. The error message should contain details of the exception.

Solution: Contact Novell Support.

A [Object Class] object could not be created. The name is missing.

Cause: An eDirectory object could not be created for the given object class because a name was not provided.

Solution: Contact Novell Support.

The policy named [Policy Name] contains a cycle in its next transformation list.

Cause: This is a warning message generated when Designer encounters a circular loop in the policy chain.

Solution: Remove the policy loop by correcting the next policy in the Policy Set view.

The policies named [Policy name] contain cycles in their next transformation lists.

Cause: This is a warning message generated when Designer encounters a circular loop in the policy chain.

Solution: Remove the policy loop by correcting the next policy in the Policy Set view.

Driver [Driver name] could not be restarted for the deployed changes to be in effect.

Cause: Designer was unable to restart a driver after a deployment.

Solution: Turn on DSTRACE screen in eDirectory to identify the error preventing the driver from starting.

Driver '[Driver Name]' is disabled and could not be restarted for the deployed changes to be in effect.

Cause: Designer was unable to restart a driver after a deployment because its Driver Start option is set to *Disabled*.

Solution: Change the Driver Start option to *Manual* or *Auto-start* under the driver properties and then deploy the driver.

Driver '[Driver Name]' could not be stopped for the deployed changes to be in effect.

Cause: Designer was unable to stop a running driver after a deployment.

Solution: Turn on DSTRACE screen in eDirectory to identify the error preventing the driver from stopping.

An invalid request to set up security on an exported driver was made, no Driver objects were provided. The request cannot be processed.

Cause: The code to set up the security equivalence for a deployed driver was passed an invalid parameter.

Solution: Contact Novell Support.

Warnings

The version of Identity Manager running on the server named '[Server Name]' does not support all the features of Designer. Although you can import a configuration from that server, changes may not work if the configuration is deployed back to it.

Cause: An import or deploy action was made to an eDirectory server running an unsupported version.

Solution: The server must be upgraded for deployments.

An internal error has occurred. The parameters passed into the importer were invalid.

Cause: The code that performs the import was passed an invalid parameter.

Solution: Contact Novell Support.

The '[Attribute Name]' attribute of '[Object Name]' refers to a policy that does not exist or cannot be accessed.

Cause: The driver configuration file being processed contains a DN attribute that cannot be resolved in eDirectory.

Solution: Verify or correct the DN attribute value on the specified object in eDirectory.

An external reference to '[Object Name]' was not handled.

Cause: The driver configuration file being processed contains a DN attribute that cannot be resolved in eDirectory.

Solution: Contact Novell Support.

The XML for the policy named '[Object Name]' contained in the [Policy Type] named '[Policy Name]' does not contain valid XML for a policy. '[Root Node]' is not recognized as the root node for policy XML. The policy is being ignored.

Cause: The policy being imported does not contain a valid XML document.

Solution: Correct the content of the policy in eDirectory.

A [Item Type] can only be imported into a [Item Type].

A [Item Type] can only be imported into a [Item Type] or [Item Type].

Cause: An attempt was made to import an Identity Manager object into an invalid parent object. For example, policies might not be imported into a Driver Set. The code should prevent this from happening, but this error identifies scenarios that were not caught.

Solution: Contact Novell Support.

An unhandled import request was encountered in DeployImporter_Import method [Object DN].

Cause: An attempt was made to import an unknown object or attribute from eDirectory. The code should prevent this from happening, but this error identifies scenarios that were not caught.

Solution: Contact Novell Support.

Could not access the driver configuration file named '[File Name]'.

Cause: Designer could not open or parse the given driver configuration file.

Solution: Contact Novell Support.

The driver filter could not be read from the driver named '[Driver Name]'.

Cause: Designer could not import the Driver filter.

Solution: Turn on the DSTRACE in eDirectory to determine the error, then contact Novell Support.

An error was encountered processing the driver configuration file. The variable named [Variable Name] is defined more than once.

Cause: The driver configuration file has a variable that is being defined multiple times.

Solution: If you are importing a driver configuration file from a file, edit the file and remove multiple declarations for the specified variable. If this is a dynamically generated configuration file (import/deploy to eDirectory), turn on XML tracing for import/deploy to get a trace of the generated configuration file, then contact Novell Support. To turn on trace for Designer, select *Window > Preferences > Identity Manager > Application > Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*.

An error was encountered processing the driver configuration file. The declaration of the Node variable named [Variable Name] is invalid. The [Attribute name] attribute is missing.

Cause: The driver configuration file being processed has an invalid variable declaration.

Solution: If you are importing a driver configuration file from a file, edit the driver configuration file and correct the variable declaration. If this is a dynamically generated configuration file (import/deploy to eDirectory), turn on XML tracing for import/deploy to get a trace of the generated configuration file, then contact Novell Support. To turn on trace for Designer, select *Window > Preferences > Identity Manager > Application > Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*.

An error was encountered processing the driver configuration file. Flexible prompting requires a 'use-when-value' when a 'use-when-var' is specified.

Cause: The driver configuration file being processed has an error.

Solution: If you are importing a driver configuration file from a file, edit the driver configuration file and add a use-when-value for the specified use-when-var. If this is a dynamically generated configuration file (import/deploy to eDirectory), turn on XML tracing for import/deploy to get a trace of the generated configuration file, then contact Novell Support. To turn on trace for Designer, select *Window > Preferences > Identity Manager > Application > Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*.

An error was encountered processing the driver configuration file. Flexible prompting requires a 'use-when-var' when a 'use-when-value' is specified.

Cause: The driver configuration file being processed has an error.

Solution: If you are importing a driver configuration file from a file, edit the file and add a use-when-var for the specified use-when-value. If this is a dynamically generated configuration file (import/deploy to eDirectory), turn on XML tracing for import/deploy to get a trace of the generated configuration file, then contact Novell Support. To turn on trace for Designer, select *Window > Preferences > Identity Manager > Application > Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*.

The variable named [Variable Name] has been referred to but not defined in the driver configuration file being processed.

Cause: The driver configuration file has a variable that is being referenced but has not been defined.

Solution: If you are importing a driver configuration file from a file, edit the driver configuration file and add a declaration for the specified variable. If this is a dynamically generated configuration file (import/deploy to eDirectory), turn on XML tracing for import/deploy to get a trace of the generated configuration file, then contact Novell Support. To turn on trace for Designer, select *Window > Preferences > Identity Manager > Application > Trace > Enable Tracing*. In the Trace window, select the check box for *Include XML Processor Traces*.

An error was encountered processing the driver configuration file. Built-in variables cannot be used as a flexible prompting control variable. The reference to the variable named '[Variable Name]' is invalid.

Cause: The driver configuration file being processed contains an invalid reference to a variable.

Solution: If this is a dynamically created configuration file generated during an import/deploy action, contact Novell Support. If this is a driver configuration file being imported from disk, edit and correct the configuration file for the variable specified.

An error was encountered processing the driver configuration file. There was a non-checkbox reference to the checkbox variable named '[Check Box Variable name]'.

Cause: The driver configuration file being processed contains an invalid reference to a check box variable.

Solution: If this is a dynamically created configuration file that is generated during an import/deploy action, contact Novell Support. If this is a driver configuration file being imported from disk, edit and correct the configuration file for the check box variable specified.

An error was encountered processing the driver configuration file. An unhandled import prompt was encountered.

Cause: The driver configuration file being processed contains an invalid prompt type.

Solution: If this is a dynamically created configuration file that is generated during an import/deploy action, contact Novell Support. If this is a driver configuration file being imported from disk, edit and correct the configuration file.

The eDirectory tree corresponding to the Identity Vault named '[Identity Vault Name]' cannot be accessed. Directory browsing cannot be performed.

Cause: Designer attempted to access eDirectory through an eDirectory browse button in the Driver Configuration Wizard, but the connection could not be created.

Solution: Cancel out of the Driver Configuration Wizard, set up the connection parameters in Identity Vault, and run the Driver Configuration Wizard again.

The partition could not be created on the '{0}' object. The problem may be that it has not replicated to the master yet. You can try creating the partition manually later.

Cause: Designer attempted to create a partition when deploying a driver set and the partition operation failed.

Solution: Turn on the eDirectory tracing options for partitioning to determine why the eDirectory partitioning operation failed.

The Driver Set was created but did not replicate to all the servers in the replica ring. The deployment cannot proceed.

Cause: Designer cannot deploy per-server attributes until the driver set has replicated to the eDirectory server.

Solution: Turn on the eDirectory tracing options for replication and determine why eDirectory replication is not occurring.

There are no servers associated with the Driver Set named '{0}'. There must be at least one server associated with any Driver Set being deployed or the Driver Set containing any objects being deployed.

Cause: Designer cannot deploy an Identity Vault or driver set with an empty server list.

Solution: Edit the properties of the Identity Vault and the driver set to add a server to the server lists. The Identity Vault name '[Identity Vault Name]' does not contain any Driver Set objects to deploy.

Cause: You cannot deploy an Identity Vault that does not contain at least one driver set.

Solution: Add a Driver Set to the Identity Vault.

'[User Name]' could not be authenticated to '[Host Name]'. Cannot proceed.

Cause: Designer could not authenticate to the eDirectory tree.

Solution: Verify that the host name, user, and password for the Identity Vault are correct in the Identity Vault properties.

The Identity Vault named '[Identity Vault Name]' does not contain the eDirectory tree to access. Cannot proceed.

Cause: The Identity Vault does not contain a host address or DNS name for authentication.

Solution: Specify the host address or DNS name for the Identity Vault in the Properties view or Properties page.

Deploy_Util_NoIdentityVault=The {2} named '{1}' is not contained in an {0}. Cannot proceed.

The Identity Vault named '[Identity Vault name]' does not contain the DN of the user to authenticate to the target eDirectory tree with. Cannot proceed.

Cause: The Identity Vault does not contain a user for authentication.

Solution: Specify the user for the Identity Vault in the Properties view or Properties page.

The server list on the parent Driver Set for the following eDirectory Driver is empty. We were unable to import the connected eDirectory Driver:

Cause: Designer uses the per-server Shim Auth Server attribute of an eDirectory driver to identify the tree and connected eDirectory driver to import. Because the server list is empty, the connected eDirectory driver cannot be imported.

Solution: Fix the server list on the driver set for the eDirectory driver and the Drivers Shim Auth Server attribute in eDirectory or import the connected eDirectory driver separately.

The Shim Auth Server parameter for the eDirectory Driver '[Driver Name]' on server '[Server Name]' is empty. We were unable to import the connected eDirectory Driver.

Cause: Designer uses the Shim Auth Server parameter of an eDirectory driver to identify the tree and connected eDirectory driver to import. If this parameter is empty, the connected eDirectory driver cannot be imported.

Solution: Fix the Shim Auth Server parameter on the eDirectory driver or import the connected eDirectory driver separately.

Unable to save Driver Configuration to file '[File Name]'.

Cause: Designer was unable to save an exported driver configuration file.

Solution: Try to save the file to a different directory or filename.

Unable to clear contents of Driver Configuration file '[File Name]'.

Cause: Designer was unable to clear the contents of a driver configuration file that is being overwritten.

Solution: Delete the configuration file being overwritten.

Setting up the Security Equals and Excluded objects may only be performed on a Driver object.

Cause: An invalid object was selected in the Modeler or Outline view.

Solution: Select a single Driver object to set up security equivalences or excluded users.

Only one Driver object can be selected to set up the Security Equivalences and Excluded objects.

Cause: Multiple Driver objects are selected in the modeler or outline view.

Solution: Select a single Driver object to set up security equivalences or excluded users.

The selected Driver '{0}' has not been deployed or cannot be found in the eDirectory '{1}'.

Cause: Designer cannot resolve to the Driver object in eDirectory to set up the security equivalences or excluded user list.

Solution: Deploy the driver to eDirectory before setting up the security equivalences or excluded users.

The eDirectory tree corresponding to the Identity Vault named '[Tree Name]' cannot be accessed. Setting up the Driver Security Equivalence/ Excluded Users cannot be performed.

Cause: Designer cannot connect or authenticate to the eDirectory tree to set up a driver's security equivalences or excluded user list.

Solution: Verify that the eDirectory parameters specified on the Identity Vault are correct and that the eDirectory server is running.

The Identity Vault named '[Identity Vault Name]' has no deployment DN specified. It is not deployable.

Cause: A deployment context is not specified on the Identity Vault or driver set being deployed.

Solution: Add a deploy DN (context) to the properties of the Identity Vault or Driver Set object in Designer.

24.5 Issues When Generating Documents

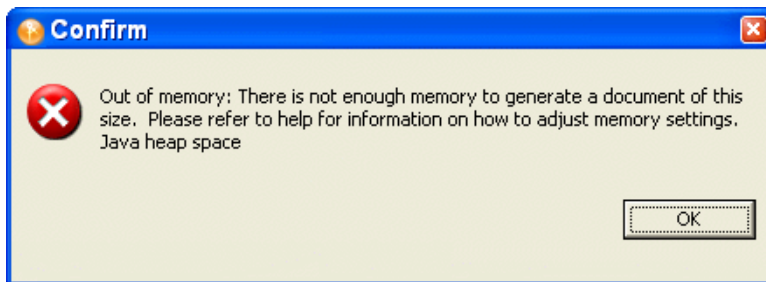
There are two main issues that occur when generating documents:

- ◆ [Section 24.5.1, “Memory Issues,” on page 509](#)
- ◆ [Section 24.5.2, “Editing Extra Identity Vault Schema Items from Your Report,” on page 511](#)

24.5.1 Memory Issues

When generating documentation for a project, Designer puts all model/project information into the document. If the Designer’s memory parameters are not being used or if your computer does not have enough memory available to run the document generation process, you see the following error message:

Figure 24-7 Out of Memory Message

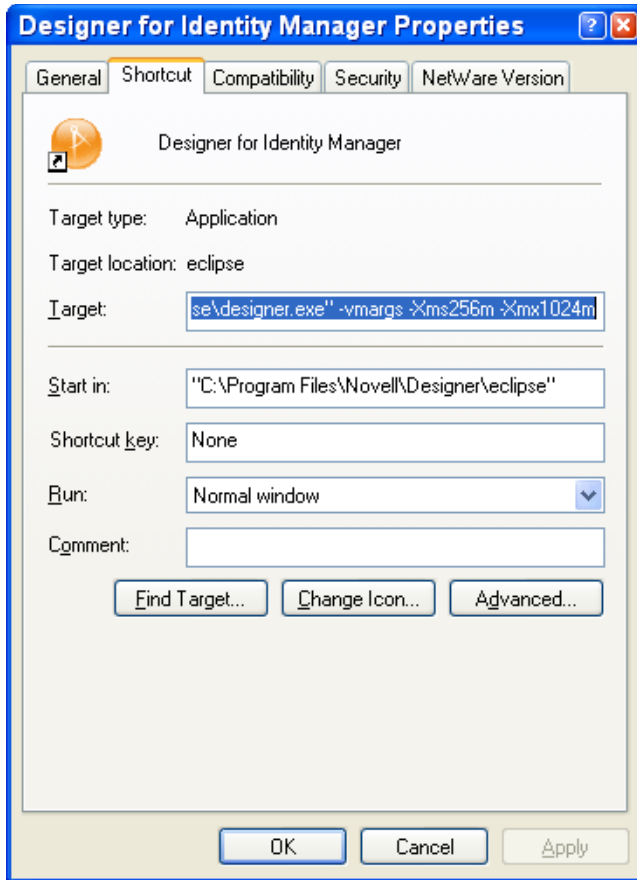


- ◆ [“Solution 1: Ensure that Memory Parameters Are in Place” on page 509](#)
- ◆ [“Solution 2: Change Memory Parameters” on page 510](#)
- ◆ [“Solution 3: Disable Schema Generation” on page 511](#)

Solution 1: Ensure that Memory Parameters Are in Place

The amount of memory that the default JVM* instance of Designer uses is defined in the memory parameters `-vmargs -Xms256m -Xmx512m`. The `-Xms256m` parameter is the minimum amount of memory (in megabytes) that is set aside for Designer, and the `-Xmx512m` parameter is the maximum setting. These are the only parameters that need to be adjusted. In Windows, these settings are included as part of the Shortcut Properties and are implemented when Designer loads.

Figure 24-8 Windows Shortcut



You can run into memory issues if you have deleted the parameters in your Designer shortcut and forgot to include them again. Or you might be clicking the Designer executable file directly, so Designer won't use these designated memory parameters. To ensure that the memory parameters are in your Windows shortcut, right-click the desktop shortcut for Designer and select *Properties*. On the *Target* line in the Shortcut window, you should have the following:

```
C:\Program Files\Novell\designer\eclipse\designer.exe" -vmargs -Xms256m -Xmx512m
```

For Linux workstations, the parameters are located in the `/designer/eclipse/StartDesigner.sh` file. This file should contain the following line:

```
/home/user/designer/eclipse/Designer -vmargs -Xms256m -Xmx512m
```

Solution 2: Change Memory Parameters

The default memory parameter should be adequate for most document generation. However, if you have enough available memory, change the `-Xmx512m` parameter to `-Xmx768m` or `-Xmx1024m`. This changes the maximum setting.

NOTE: These settings are adjusted in megabytes.

Solution 3: Disable Schema Generation

If you are using the maximum amount of memory you can on this workstation and are still receiving Out Of Memory errors when generating documents, disable the schema when generating documents. To do this, click on the *5. Appendix B: Schema* heading in the Style Editor.

Under the *Appendix B: Schema* heading, deselect the *Include this section in the final document*. When deselected, *Appendix B:* icons under the Style Editor view change to reflect the new settings. You should now be able to generate a document without memory errors.

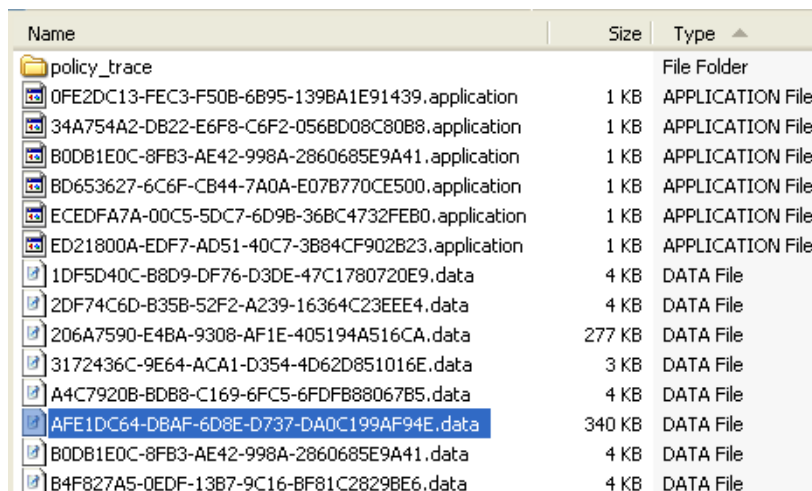
NOTE: This issue is being addressed in later versions of Designer.

24.5.2 Editing Extra Identity Vault Schema Items from Your Report

If you import a project with the entire Identity Vault schema and run document generation on that project, you can have an extra 600 pages or more of schema information. This happens because you have all of the Identity Vault's base schemas included in the document. Follow these steps if you do not want to document all of the Identity Vault's base schemas:

- 1 Go into the workspace folder for this project and delete the `.data` file.

For example, on a Windows workstation, the `Blanston1_files` directory is found at `C:\Documents and Settings\Novell User\workspace\Blanston1\Blanston1_files`. By clicking *Type*, you will see all of the `.data` files shown together. The largest `.data` file holds the base Identity Vault schema.



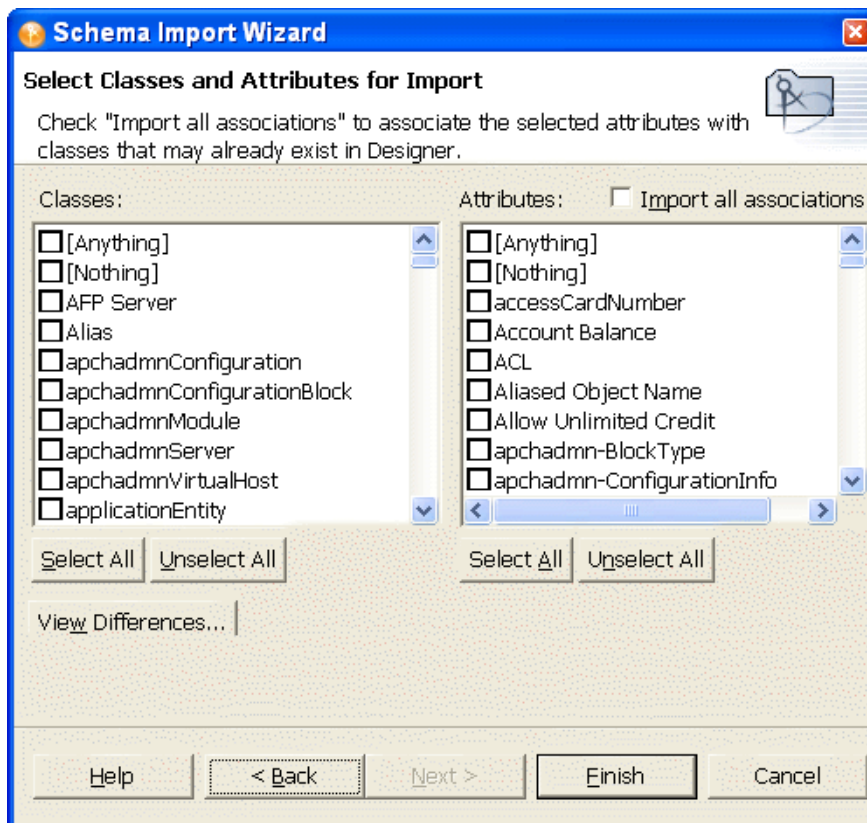
Name	Size	Type
policy_trace		File Folder
0FE2DC13-FEC3-F50B-6B95-139BA1E91439.application	1 KB	APPLICATION File
34A754A2-DB22-E6F8-C6F2-056BD08C80B8.application	1 KB	APPLICATION File
B0DB1E0C-8FB3-AE42-998A-2860685E9A41.application	1 KB	APPLICATION File
BD653627-6C6F-CB44-7A0A-E07B770CE500.application	1 KB	APPLICATION File
ECEDFA7A-00C5-5DC7-6D9B-36BC4732FEB0.application	1 KB	APPLICATION File
ED21800A-EDF7-AD51-40C7-3B84CF902B23.application	1 KB	APPLICATION File
1DF5D40C-B8D9-DF76-D3DE-47C1780720E9.data	4 KB	DATA File
2DF74C6D-B35B-52F2-A239-16364C23EEE4.data	4 KB	DATA File
206A7590-E4BA-9308-AF1E-405194A516CA.data	277 KB	DATA File
3172436C-9E64-ACA1-D354-4D62D851016E.data	3 KB	DATA File
A4C7920B-BDB8-C169-6FC5-6FDFB88067B5.data	4 KB	DATA File
AFE1DC64-DBAF-6D8E-D737-DA0C199AF94E.data	340 KB	DATA File
B0DB1E0C-8FB3-AE42-998A-2860685E9A41.data	4 KB	DATA File
B4F827A5-0EDF-13B7-9C16-BF81C2829BE6.data	4 KB	DATA File

If you have more than one Identity Vault in your project, you see more than one large `.data` file in the project's folder.

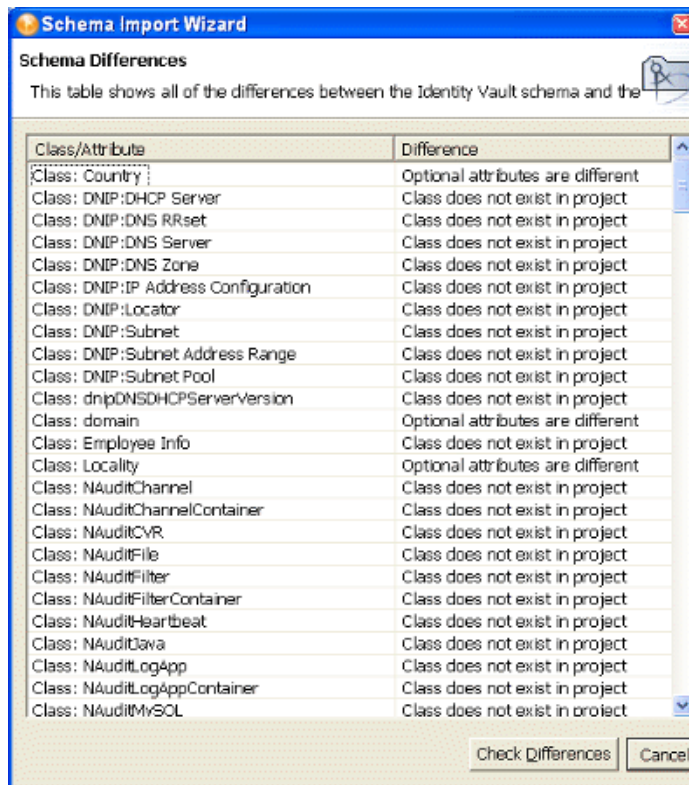
- 2 Open the .data file in a text-based editor (in this instance, Wordpad) to make sure you have the schema file. You see <schema-def> at the top of the file:

```
<schema-def>
<class-def class-name="AFP Server">
  <attr-def attr-name="Account Balance" multi-value
  <attr-def attr-name="ACL" type="structured"/>
  <attr-def attr-name="Allow Unlimited Credit" mult
  <attr-def attr-name="Audit:File Link" multi-value
  <attr-def attr-name="Authority Revocation" multi-
  <attr-def attr-name="auxClassCompatibility" read-
  <attr-def attr-name="Back Link" read-only="true"
```

- 3 Assume that you don't want the Identity Vault's entire base schema, only the schema differences between your project and the Identity Vault. To get the schema differences, delete any large .data files and re-import the project's schema, using the following steps:
 - 3a Bring up the project in Designer's Modeler view. Right-click the Identity Vault and select *Live Operations > Import Schema*.
 - 3b On the Select Source for Import page, select *Import from eDirectory* if you are connected to an actual Identity Vault, or select *Import from .sch File* if you have saved the schema files locally.
 - 3c On *Import from eDirectory*, type the host name, user name, and password connection information. Click *Next*.
 - 3d On the Select Classes and Attributes for Import page, click *View Differences*.



- 3e** On the Schema Differences page, you see the schema differences between Identity Vault and your project.



- 3f** Select *Check Differences*. This brings you back to the Select Classes and Attributes for Import page with the differences now selected under the Classes and Attributes headings. By default the *Import All Associations* box is also selected. Leave it selected and click *Finish*.

- 3g** On the Import Summary page, click *OK*.

When you run document generation on this project, you only document the schema differences that you imported, not the whole Identity Vault schema. Your present choices are to import the entire Identity Vault schema or just the schema differences between the Identity Vault and the project. You can also choose to not include the schema in your document by deselecting *Include This Section in the Final Document*, as described in [“Solution 3: Disable Schema Generation” on page 511](#). This document generation of the schema a known problem and will be fixed in the future.

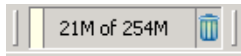
24.6 No F1 Help in Maximized Editors

Context-sensitive help is available when you press F1. However, if you maximize an editor (for example, the Modeler), help topics do not display when you press F1. To view the help, minimize the editor.

24.7 Freeing Heap Memory

A status field at the bottom of Designer displays heap memory used and heap memory available for an application or other item in Designer.

Figure 24-9 *The Heap Memory Display*



The information varies, depending on which item (for an example, an application) you click in the Modeler, Outline view, or other editors.

To free unused heap memory at any time, click the *Run Garbage Collector* icon.

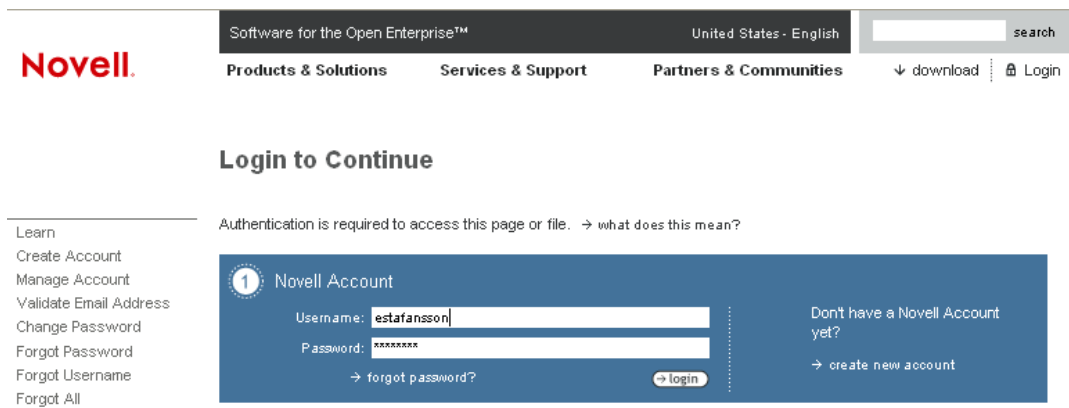
Figure 24-10 *The Run Garbage Collector Icon*



24.8 Reporting Bugs and Giving Feedback

Gathering bugs and getting your ideas are keys to improving the performance of Designer and making Designer a tool of choice for you. To send us your feedback, select *Help > Report a Bug or Feedback*. We encourage you to try it!

- 1 Select *Help > Report a Bug or Give Feedback*.
- 2 Log in to Bugzilla.



If you don't have an account, you can easily create one.

3 Select the component in the product that you are reporting on.

Enter Bug: Identity Manager Designer

This page lets you enter a new bug into Bugzilla.

» Before reporting a bug, please do the following

- [Search for the bug](#)
- [Look at the list of most frequently reported bugs](#)
- [Read the bug writing guidelines](#)

» Report Bug Found in - Identity Manager Designer

Required Field*

Found in Version:*	<input type="text" value="1.0.0 Designer"/> <input type="text" value="1.1 Designer"/> <input type="text" value="2.0.0 Designer"/>	
Component:*	<input type="text" value="Builds and download site"/> <input type="text" value="Common Widgets"/> <input type="text" value="Configuration Management"/> <input type="text" value="Dataflow Editor"/> <input type="text" value="Dataflow View"/> <input type="text" value="Denlow"/>	
Hardware:	<input type="text" value="Other"/>	OS: <input type="text" value="Other"/>
Severity:	<input type="text" value="Normal"/>	
Found By:	<input type="text" value="Other"/>	Partner ID: <input type="text"/>
NTS Priority:	<input type="text"/>	NTS Support #: <input type="text"/>

» Bug Details

Summary:*	<input type="text"/>
Description:*	<input type="text"/>

The Designer 2.1 product is preselected. If you don't know which component you are reporting on, select your best guess (for example, Modeler).

4 In the *Summary* field, summarize the problem or your request for an enhancement.

5 In the *Description* field, describe the bug or enhancement.

If you are reporting a bug, provide clear steps on how to reproduce the problem.

Designer: An Effective Tool



Designer has abundant, easy-to-use features.

- ◆ Section A.1, “Core Application Services and Framework,” on page 517
- ◆ Section A.2, “Installation,” on page 518
- ◆ Section A.3, “Enterprise Modeler,” on page 518
- ◆ Section A.4, “Project View,” on page 519
- ◆ Section A.5, “Outline View,” on page 519
- ◆ Section A.6, “Import and Deploy,” on page 519
- ◆ Section A.7, “XML Editor,” on page 520
- ◆ Section A.8, “Configuration Management,” on page 520
- ◆ Section A.9, “Policy Management,” on page 520
- ◆ Section A.10, “Policy Simulation,” on page 522
- ◆ Section A.11, “Dataflow View and Editor,” on page 522
- ◆ Section A.12, “Entitlements,” on page 522
- ◆ Section A.13, “E-Mail Notification Templates,” on page 523
- ◆ Section A.14, “Document Generator,” on page 523
- ◆ Section A.15, “Directory Tools,” on page 523
- ◆ Section A.16, “Version Handling,” on page 524
- ◆ Section A.17, “Project Converter,” on page 524
- ◆ Section A.18, “Project Checker,” on page 524

A.1 Core Application Services and Framework

- ◆ Basic views: Error Log, Outline, Project, Navigator, Tasks
- ◆ A fully indexed and searchable Help system
- ◆ General undo/redo capabilities
- ◆ Internal common services: utilities, APIs, widgets, and resource caches
- ◆ Log and trace files
- ◆ Perspective management: open close, copy, and reset
- ◆ Preferences: general appearance, fonts, colors, styles, file associations, and history
- ◆ Text editor: CUA keyboard, tabs, print margins, line numbers, colors, preferences, find/replace, bookmarks, go-to line, print
- ◆ Toolbar, menu bar, welcome bar, perspective bar, and fast view bar
- ◆ Trace view with preferences for controlling the output
- ◆ View and editor management
- ◆ Welcome page: what’s new, updates, and Flash demos

- ◆ Localized software for international customers

A.2 Installation

- ◆ Ability to install newer plug-ins, manage installed plug-ins, and back-rev if necessary
- ◆ Option to place the Designer icon on the desktop
- ◆ Option to run Designer directly from the install
- ◆ Automatic (or manual) checking for updates, with a setting to turn updates on or off
- ◆ Language selection for localized installs

A.3 Enterprise Modeler

- ◆ Alignment capabilities: grid, snap, ruler, distribution, and hints
- ◆ Auto-layouts: star, circle, half-circle, and expand or contract in and out
- ◆ Copy/paste within editors and across editors
- ◆ Color themes
- ◆ Delete (with multiselect support)
- ◆ Driver Configuration Wizard integration
- ◆ A graphical Dataflow view and the ability to manipulate data flow arrows: update filters, schema, and mapping policies visually
- ◆ Decoration support: change vaults to trees, mark drivers as remote or communicating across firewalls, resource icons (help desk, log, and audit)
- ◆ Domain group management: add to group, ungroup, nest groups, clear, minimize/maximize, expand/collapse all, restore, auto-size, icon library
- ◆ Drag and drop Identity Vaults, driver sets, applications, domain groups, and resources
- ◆ Integration among editors
- ◆ Modeling Palette: drag and drop, folders, configurable in Preferences
- ◆ Graphic snapshot services
- ◆ Help: context-sensitive, search, browse, and demos
- ◆ Direct link from the help menu to the Identity Manager forums
- ◆ High-level architect mode and low-level developer modes
- ◆ Inline editing of titles and labels
- ◆ Integrated hierarchical Outline view
- ◆ Keyboard support
- ◆ Edit icons
- ◆ Manage line connections: auto-connecting, auto-highlight, mouse gravitation, collapsing lines when groups collapse, line straightening, reconnect, eDir-to-eDir connections, multidriver connections
- ◆ Preferences: many preferences for display behaviors, prompts, and auto-wizards
- ◆ Print support with various fit-to-page options
- ◆ Search a model: case sensitivity, integrated editors, and searching by class

- ◆ Manage selections: marquee and selection tools, synchronized selections with external views, select all, and extend or remove from selection
- ◆ Integrated services: start or stop drivers, driver information, and import or deploy
- ◆ Subapplication modeling with applications that contain applications
- ◆ Table editor: sort columns and launch editors
- ◆ Dataflow editor
- ◆ Server creation requests version information
- ◆ Zoom, pan, and scan (in the Thumbnail view)

A.4 Project View

- ◆ Copy projects
- ◆ Delete projects
- ◆ Create projects with the New Identity Manager Project Wizard
- ◆ Expand/collapse all
- ◆ Help-me-get-started process
- ◆ Import from directory or file system
- ◆ Move project
- ◆ Multiproject support
- ◆ Project documentation folder and content management

A.5 Outline View

- ◆ Create new items: vaults, drivers, driver sets, and applications
- ◆ Delete a selected item
- ◆ Integrated launching of editors
- ◆ Expand/collapse all
- ◆ Export to a driver configuration file
- ◆ Filter the view to show only certain objects
- ◆ List all project items hierarchically
- ◆ Host the Modeler Thumbnail view and the Policy Flow view
- ◆ Import/deploy integration
- ◆ Save and restore the expansion state and selection
- ◆ Synchronize selections with the Modeler
- ◆ Undo/redo

A.6 Import and Deploy

- ◆ Deploy groups, identity vaults, driver sets, drivers, policies, channels, and e-mail templates
- ◆ Import an existing Identity Manager solution from the directory into a project
- ◆ Import driver sets, drivers, policies, and channels

- ◆ Import both sides of eDir-to-eDir drivers
- ◆ Import and deploy driver configuration files
- ◆ Handle multiserver environments
- ◆ Set up driver security
- ◆ Easily protect overwriting during conflicts
- ◆ View changes before you deploy

A.7 XML Editor

- ◆ Used in policy management and configuration editing
 - For notification templates, you can use an HTML editor.
- ◆ Outline view support
- ◆ Find and replace
- ◆ Highlight syntax
- ◆ Auto suggest
- ◆ Auto-insert end tags

A.8 Configuration Management

- ◆ Edit applications: name, type, icon, environment, and administrator notes
- ◆ Edit drivers: Global Communication Values (GCVs), log level, module, authentication, driver parameters, startup option, driver manifest, named passwords, icon, and trace
- ◆ Edit driver sets: GCVs, servers, trace, and log level
- ◆ Edit e-mail templates: collections, e-mail notification server, and all template content with token support
- ◆ Edit Identity Vaults: credentials, administrator notes, and server list
- ◆ Edit resources: name, type, icon, and administrator notes
- ◆ Edit servers: name, context, host, DNS name, notes, and contact information
- ◆ Properties view: column resizing, hierarchical groups, inline editors and editor launching, and support for simultaneously editing selected items

A.9 Policy Management

- ◆ [Section A.9.1, “Policy Flow View,” on page 520](#)
- ◆ [Section A.9.2, “Filter Editor,” on page 521](#)
- ◆ [Section A.9.3, “Schema Map Editor,” on page 521](#)
- ◆ [Section A.9.4, “Policy Builder,” on page 521](#)

A.9.1 Policy Flow View

- ◆ Reorder, add, and delete policies (Policy Script or XSLT)
- ◆ Policy editor integration

- ◆ View policy flow between an application and the Identity Vault
- ◆ View Policy Set details in the Policy Set view
- ◆ In the Outline view, view a hierarchical organization of policies in the Policy Set view

A.9.2 Filter Editor

- ◆ Add classes and attributes
- ◆ Copy the filter
- ◆ Delete
- ◆ Multiselect support
- ◆ Keyboard support
- ◆ Refresh the schema
- ◆ Expand/collapse all
- ◆ Configure default values
- ◆ Create a home directory
- ◆ Track member templates
- ◆ Merge authority
- ◆ Optimize modifications
- ◆ Undo/redo
- ◆ XML editor integration

A.9.3 Schema Map Editor

- ◆ Add classes and attributes
- ◆ Delete
- ◆ Multiselect support
- ◆ Keyboard support
- ◆ Refresh the schema
- ◆ Expand/collapse all
- ◆ Non-class specific handling
- ◆ Integrated XML editor

A.9.4 Policy Builder

- ◆ Create and edit policy scripts
- ◆ Define rules: conditions, actions, groups
- ◆ Support for predefined rules
- ◆ Emphasis on productivity
- ◆ Nested inline editors
- ◆ Keyboard support

- ◆ Copy/paste within an editor and across editors
- ◆ Drag and drop
- ◆ Integrated Help
- ◆ Policy creation wizards
- ◆ Import/export
- ◆ Namespace editing
- ◆ Expand/collapse all
- ◆ Reordering
- ◆ Integrated e-mail templates
- ◆ Integrated Global Configuration Values
- ◆ Integrated schema
- ◆ Various display preferences
- ◆ Argument builder and token editor
- ◆ Undo/redo
- ◆ Validation
- ◆ Integrated XML editor

A.10 Policy Simulation

- ◆ Launch a simulation wizard from the Policy Flow view, policy editors, and the Policy Set view
- ◆ Load sample input documents: user add, delete, query, modify, and association
- ◆ Edit input documents by using an inline XML editor
- ◆ Query document input
- ◆ Render output documents
- ◆ See the Metadirectory engine log of event processing
- ◆ Clear logs
- ◆ Select a policy set you want to simulate
- ◆ Chain simulation from one policy to the next

A.11 Dataflow View and Editor

- ◆ View the filters that allow the data to flow between all of the systems and the Identity Vaults
- ◆ View how the passwords flow through the different systems
- ◆ Add, delete and edit driver sets, drivers, filters and Identity Vaults from one location
- ◆ Generate HTML reports of the data flow and password flow

A.12 Entitlements

- ◆ Set up criteria that, once met, initiate an event to grant or revoke access to business resources
- ◆ Use Role-Based entitlements (automated) and workflow provisioning entitlements (approval required)

- ◆ Use the Entitlement Wizard graphical interface to create the types of entitlements that you can use
- ◆ For a number of drivers, use preconfigurations that already contain entitlements
- ◆ Manage entitlements by using either of two packages: iManager through Role-Based Entitlement Policies, or the User Application for workflow entitlements

A.13 E-Mail Notification Templates

- ◆ Provide users with hints when they forget passwords
- ◆ E-mail new passwords
- ◆ Explain why a password wasn't reset or didn't synchronize
- ◆ Add or import templates
- ◆ Customize e-mail messages

A.14 Document Generator

- ◆ Click a button and generate a PDF document of your project
- ◆ Edit the content of the document: title, name, legal disclaimer, executive summary, and overview
- ◆ Edit the Document Style (template) to format the style and content any way you want
- ◆ Generate a linkable appendix and a table of contents
- ◆ Generate graphical filter policy tables with a legend
- ◆ Generate Policy Builder rule tables
- ◆ Generate detailed information on all vaults, driver sets, drivers, and policies
- ◆ Generate schema tables
- ◆ Generate server configuration tables
- ◆ Include model diagrams
- ◆ Manage and view multiple styles in the Project view
- ◆ Number pages

A.15 Directory Tools

- ◆ Directory browser for selecting objects: filtered on type; switch to another vault
- ◆ Import and deploy the schema to a directory or .sch file
- ◆ Use available eDirectory™ schemas for offline-from-scratch modeling
- ◆ Manage schema attributes: add, rename, delete, set constraints, syntax, size, see what classes are used, define hierarchy, and specify mandatory/naming/optional attributes
- ◆ Manage schema classes: add, rename, delete, set class flags, define hierarchy, specify mandatory/naming/optional attributes
- ◆ Schema management filter to see only changes that you've made in a project
- ◆ Remotely control desktop applications and resources
- ◆ Launch and use iManager

- ◆ Manage directory objects

A.16 Version Handling

- ◆ Most Designer components are now sensitive to the version of Identity Manager that the piece of configuration is going to run on
- ◆ Prevents incompatible configurations by adapting the UI to show only available features for the selected version.
- ◆ The Add Server Wizard lets you specify the Identity Manager version. Easily change to another version.
- ◆ Loads the Legacy Policy Builder loads if your Identity Manager version earlier than Identity Manager 3.5. Loads the new Policy Builder loads if your Identity Manager version is 3.5.
- ◆ Warning or preventing deployment to a legacy server if an incompatible configuration is detected during import or deployment.
- ◆ Driver configuration files report the minimum required engine version.

A.17 Project Converter

- ◆ Is a framework of chained converters
- ◆ Converts earlier projects to Identity Manager 3.5 formats
- ◆ Provides a backup of an earlier project in case you need to use the earlier version
- ◆ Launches automatically

A.18 Project Checker

- ◆ Checks project for validity and completeness before deploying
- ◆ In manual mode by default, but can be changed to run at every save
- ◆ Checks whether global variables exist as GCVs when referenced from conditions or actions

Document Generator Core Support Templates

B

- ◆ [Section B.1, “dgSection.xsl,” on page 525](#)
- ◆ [Section B.2, “dgFormat.xsl,” on page 526](#)
- ◆ [Section B.3, “idmConfig.xsl,” on page 528](#)
- ◆ [Section B.4, “idmUtil.xsl,” on page 529](#)

B.1 dgSection.xsl

Import: dgFormat.xsl

Table B-1 *dgSection.xsl*

Template	Description
match “/”	Main template that invokes all sub-templates. Users can override this template to create their own template behavior; however it is recommended that you override the Section.Content, Section.Body or Section.Main.
Section.Sequence	Main template that invokes all sub-templates. Users can override this template to create their own template behavior; however it is recommended that you override the Section.Content, Section.Body or Section.Main.
Section.Main	This section includes Section.Content and Section.Children.
Section.Content	This section includes Section.Title and Section.Body.
Section.Body	The body content of the section.
Section.ShowStyleAttributes	Describes the default way to display attributes (when no template is defined). Parameters: border - border used for tables. The default value is 0.5pt solid black.
Section.Children	Inserts the child sections that are passed as a parameter into this template.
Section.PageLayout	Formats the page layout including paper size, headers, page numbering, and so forth. The Section.Main template is called to insert the document into this layout.
Section.staticContent	This template formats the page layout, including paper size, headers, page numbering, and so forth. The Section.Main template is called to insert the document into this layout.
Section.Title	Creates a title block containing the appropriate title text and link.
Section.TitleText	Gets the text that should be displayed for this section.

B.2 dgFormat.xsl

Table B-2 *dgFormat.xsl*

Template	Description
Format.Title	<p>This template takes care of all the details involved in formatting a title block.</p> <p>Parameters:</p> <p>text - Text to display. id - Id for linking to this title (such as from the table of contents). font - Font size to use. image - Image to show as a bullet. The auto value tries to determine an image based on the current element.</p>
Format.FigureTitle	<p>This template takes care of all the details involved in formatting a figure title block.</p> <p>Parameters:</p> <p>title - Title text. description - Description text.</p>
Format.OutputTextArea	<p>Formats parameter information returned from a text area control that can contain HTML tags. If there is no HTML prefix, then line breaks are inserted.</p> <p>Parameters:</p> <p>value - Value of the textarea to output.</p>
Format.EnabledStatus	<p>Shows enabled image if value is True, disabled image shows only when parameter showDisabled is set to True.</p> <p>Parameters:</p> <p>value - Enabled, True/False. showDisabled - Set to True if the disabled image should show when the value is False. The default value is False.</p>
Format.Checkbox	<p>Shows a check box image if the value is True, or an empty check box image otherwise.</p> <p>Parameters:</p> <p>value - Checked, True/False. default - Default value if "", False or some other value besides True exists The default value is False.</p>
Format.PropertyRow	<p>Shows a table property row with two columns, one for the name and one for the value.</p> <p>Parameters:</p> <p>propertyName - The property name. propertyValue - The property value. border - Border used for table. The default value is 0.5pt solid black. disable-output-escaping - Disable output escaping on the output value, used so you can pass escaped FO content. The default value is False. showEmpty - Show empty values. The default value is False.</p>

Template	Description
Format.ContextRow	<p>Shows a contextual row with related attribute. Use this inside a table.</p> <p>Parameters: text - Text to display. level - Level or indent. The default value is 1. href - Href value to link to another portion of the document. image - Image to show as a bullet. The auto value will try to determine an image based on the current element. show-page-ref - Show page reference; True/False. The default value is True.</p>
Format.ShowBulletImage	<p>Show a bullet image.</p> <p>Parameters: image - Image to show as a bullet. The auto value will try to determine an image based on the current element.</p>
Format.XMLFigure	<p>This template takes care of all the details involved in formatting a figure that shows XML content.</p> <p>Parameters: title - Title text. description - Description text. xml - XML data to show in the figure in text. You can also use a "." to get the current node and children simple-format - If True, this shows the XML without text selecting. This can also be preferred if name space attributes need to be included or if the XML is not well-formed. The default value is False.</p>
match "node()" mode "xml-to-text"	<p>XML-to-text formatting function.</p> <p>Parameters: attr-name-color, attr-value-color</p>
match "@*" mode "xml-to-text"	<p>XML-to-text formatting function.</p> <p>Parameters: attr-name-color, attr-value-color</p>
match "text()" mode "xml-to-text"	<p>XML-to-text formatting function.</p>
match "comment()" mode "xml-to-text"	<p>XML-to-text formatting function.</p> <p>Parameters: comment-color</p>
Format.ImageFigure	<p>This template formats a figure that shows an image for its content.</p> <p>Parameters: title - Title text. description - Description text.</p>
Format.PageBreak	<p>Inserts a page break.</p>

Template	Description
Format.BasicLink	<p>Creates a basic link to the given href using the given text. If the href is empty, it only outputs the text value.</p> <p>Parameters: text - Link text href - Link href</p>
Format.BasicLinkToReferencedItem	<p>Creates a basic link to the XSI referenced item. This uses the @guid attribute to build the link. If no @guid is available, only the text label is rendered.</p> <p>Parameters: xsiHref - XSI value of referenced node.</p>
Format.Uppercase	<p>Used to convert a string to uppercase text.</p> <p>Parameters: value - The value you want to convert to upper-case.</p>
Format.SmartSpace	<p>Used to convert a string to smart-spaced text.</p> <p>Parameters: value - The value you want to smart-space.</p>
Format.OutputDebugParameters	<p>Outputs the debug parameters for a section when the attribute DEBUG_PARAMS is enabled.</p>
Format.Debug	<p>Outputs the specified text in a debug format when attribute DEBUG is enabled.</p> <p>Parameters: text - Debug text.</p>

B.3 idmConfig.xsl

Table B-3 *idmConfig.xsl*

Template	Description
match "*" mode "xmlFigure"	<p>Build an XML Figure for any policy type.</p> <p>Parameters: title, description, alwaysShowPolicyXmlSource</p>
match "xsl:stylesheet xsl:transform"	<p>Filter.</p> <p>Parameters: title, description</p>
match "attr-name-map"	<p>Attribute mapping.</p>
match "policy"	<p>Policy matching.</p> <p>Parameters: title, description, alwaysShowPolicyXmlSource</p>
opConcat	

Template	Description
opDelim	
match "@**" mode "DirXMLScript"	
match "**" mode "DirXMLScript"	
match "arg-actions" mode "DirXMLScript"	
match "arg-dn" mode "DirXMLScript"	
match "arg-value" mode "DirXMLScript"	
match "token-text" mode "DirXMLScript"	
getLabel	Utility method used to get policy related text labels. Parameters: name - The name of the label.
match "actions"	

B.4 idmUtil.xsl

Table B-4 *idmUtil.xsl*

Template	Description
IdmUtil.ItemPropertyTable	Shows a table of values for the current Item. Depending on the item might filter attributes. Parameters: title - Title text. description - Description text. showEmpty - Show empty values. The default value is False.
IdmUtil.StartOptionPropertyRow	Shows the appropriate icon and text for the startup option on the current Item. (0 = Disabled, 1 = Manual, 2 = Auto) Parameters: propertyName, propertyValue, border
IdmUtil.ItemNumbering	Gets the current item numbering in context to the Designer source (such as "2.4.5.2."). This template helps centralize what should be counted in the numbering process because several places reuse this information.
IdmUtil.ItemText	Based on the XSI type, returns text for the type, followed by a colon and the name value (for example, Identity Vault: my vault 1)
IdmUtil.ItemType	Returns text representing the type of the current Item (such as Identity Vault, Domain, or Driver Set)

Template	Description
IdmUtil.PolicySetPropertyRow	<p>Builds a property row with a list of the policies based on the next policy value.</p> <p>Parameters: policy - Root policy of the policy set, passed by attribute name (such as mappingPolicy). label - Label for the displayed value. emptyLabel - Text to show if the value is empty. The default value is (none defined).</p>
IdmUtil.PolicySetLinks	<p>Returns a list of policy set links, called recursively.</p> <p>Parameters: xsiRootPolicyHref - Root policy of the policy set.</p>
IdmUtil.ConfigValuesTable	<p>Shows a Config Value table for the given XML.</p> <p>Parameters: title - Title text. description - Description text. xml - XML value to use to create the table. border - Border used for the table. The default value is 0.5pt solid black. emptyLabel - Text to show if the value is empty. The default value is (none defined).</p>
IdmUtil.FilterTable	<p>Shows a Filter table for the given XML.</p> <p>Parameters: title - Title text. description - Description text. xml - XML value to use to create the table. emptyLabel - Text to show if the value is empty. The default value is (none defined).</p>
IdmUtil.showSyncIcon	<p>Show an IDM sync icon based on input type and sub-type.</p> <p>Parameters: type - Type is pub, sub. sub-type - Sub-type is "", sync, ignore, notify, reset.</p>
IdmUtil.ValueOfReferencedItem	<p>Returns the value of the node given the XSI expression. When extracting the name of an item, you should use the Format.BasicLinkToReferencedItem method so that the text is created as a link inside the document.</p> <p>Parameters: xsiHref - XSI value of referenced node. suffix - Suffix to append before select (the default is the current node). The default value is "."</p>
IdmUtil.ItemCustomIconFileName	<p>Get the custom icon filename for the given GUID.</p> <p>Parameters: guid - The item's GUID</p>
IdmUtil.ShowManagerIcon	<p>This method is left here for backwards compatibility. Use IdmUtil.ShowIcon.</p>

Template	Description
IdmUtil.ShowIcon	Shows the icon for the current Item. This first checks the customIconURI for a referenced image, then builds to a generic path based on the type attribute (for Drivers and Applications). Parameters: image-width - The image width to use. The default value is 49px.

Designer's Architecture

C

- ♦ [Section C.1, “Built on Eclipse Technology,” on page 533](#)
- ♦ [Section C.2, “Core Model Uses the Eclipse Modeling Framework,” on page 533](#)

C.1 Built on Eclipse Technology

Designer is based on Eclipse technology. This open source technology is actively supported by over 60 leading companies (such as IBM*, HP*, Intel*, SAP, Fujitsu*, and Oracle) and by thousands of other developers throughout the commercial industry and the academic world.

With Designer based on Eclipse, you receive the following benefits:

- ♦ Published, well-documented, and supported APIs used heavily across the industry.
- ♦ An underlying model that integrates well with any plug-in.
- ♦ A rich, advanced UI.
- ♦ Capability to fully and easily extend our product in the future.
- ♦ Better integration and leveraging of other applications.
- ♦ Open source as the basic framework.

You can install Novell tools into any Eclipse-based framework.

- ♦ Out-of-box tools (Java* editor, debugger, JUnit, ANT, CVS integration) that are free if you want to access them (and other tools) within the Designer.
- ♦ Cross-platform benefits of Java, but native performance and native look of the platform.
- ♦ Quick access to meaningful tools.
- ♦ A framework that will quickly pick up entirely new capabilities contributed by a larger community.

Instead of focusing on tools, developers are able to concentrate on providing quality content and not spend so much time on the plumbing, debugging, and fixing of the framework.

C.2 Core Model Uses the Eclipse Modeling Framework

At the heart of Designer for Identity Manager is a strong object model that is based on the Eclipse Modeling Framework. This model uses the industry standard XMI format (XML Metadata Interchange Format) for storing objects and their relationships. By basing Designer on this technology, you get the following benefits:

- ♦ Published industry-standard APIs so that you can programmatically interact with the model.
- ♦ A solid event-notification mechanism if the model ever changes.
- ♦ All views, editors, and plug-ins will be notified and in sync.
- ♦ The ability to auto-generate Java code or UML from the model, or vice-a-versa.
- ♦ Undo/redo the stack across all views, editors, and plug-ins that are in sync with the model.

The main purpose of XMI is to enable easy interchange of metadata between modeling tools (based on the OMG UML) and between tools and metadata repositories (OMG MOF based) in distributed heterogeneous environments. XMI integrates three key industry standards:

- ◆ XML (extensible Markup Language), a W3C standard
- ◆ UML (Unified Modeling Language), an OMG modeling standard
- ◆ MOF (Meta Object Facility), OMG modeling and metadata repository standard

The integration of these three standards into XMI integrates the best of OMG and W3C metadata and modeling technologies, allowing developers of distributed systems to share object models and other metadata. XMI, together with MOF and UML, forms the core of the OMG repository architecture that integrates object-oriented modeling and design tools with each other and with a MOF-based extensible repository framework.

Quickview: Menu Options

D

The following links provide information about options on the main menu and toolbar. If other items appear in the user interface, they are contributed by other plug-ins. Documentation for those items are in that plug-in's section of the help system.

- ◆ [Section D.1, “Options on the File Menu,” on page 535](#)
- ◆ [Section D.2, “Options on the Edit Menu,” on page 537](#)
- ◆ [Section D.3, “Options on the View Menu,” on page 538](#)
- ◆ [Section D.4, “Options on the Project Menu,” on page 539](#)
- ◆ [Section D.5, “Options on the Tools Menu,” on page 539](#)
- ◆ [Section D.6, “Options on the Live Menu,” on page 540](#)
- ◆ [Section D.7, “Options on the Model Menu,” on page 541](#)
- ◆ [Section D.8, “Options on the Window Menu,” on page 546](#)
- ◆ [Section D.9, “Options on the Help Menu,” on page 552](#)

D.1 Options on the File Menu

Table D-1 Options: File Menu

Option	Description
<i>File > New > Identity Manager Project</i>	Creates an Identity Manager project and location. The project then appears in the Project view. See Section 3.3, “How to Use Designer,” on page 34 .
<i>File > New > Provisioning</i>	Helps you define directory abstraction layer objects and provisioning request definitions for the user application.
<i>File > New > More Projects</i>	Launches the Identity Manager Projects Wizard.
<i>File > New > More Projects > General > File</i>	Creates a new, empty file resource.
<i>File > New > More Projects > General > Folder</i>	Creates a new folder in the project resource folder.
<i>File > New > More Projects > General > Untitled Text File</i>	Creates an empty, untitled text file in the project folder.
<i>File > New > More Projects > Designer Document Generation > Document Generation Style (.docgen)</i>	Creates a Document Generator style (template). NOTE: At least one Designer project must exist before you can create a Document Generation Style.
<i>File > New > More Projects > Designer for Identity Manager > Identity Manager Project</i>	Creates an Identity Manager project name and location.

Option	Description
<i>File > New > More Projects > Provisioning > Directory Abstraction Layer Entity</i>	Specifies the project and application for the new entity, as well as the display name and key for the new entity. See Configuring the Directory Abstraction Layer (http://www.novell.com/documentation/idm/index.html?page=/documentation/idm/agpro/data/agproconfigdirabstractionlayer.html) .
<i>File > New > More Projects > Provisioning > Directory Abstraction Layer List</i>	Specifies the project and application for the new list as well as the key for the new list. See Configuring the Directory Abstraction Layer (http://www.novell.com/documentation/idm/index.html?page=/documentation/idm/agpro/data/agproconfigdirabstractionlayer.html) .
<i>File > New > More Projects > Provisioning > Directory Abstraction Layer Query</i>	Specifies the project and provisioning application for the new directory abstraction layer query object, as well as the display name and key. See Configuring the Directory Abstraction Layer (http://www.novell.com/documentation/idm/index.html?page=/documentation/idm/agpro/data/agproconfigdirabstractionlayer.html) .
<i>File > New > More Projects > Provisioning > Directory Abstraction Layer Relationship</i>	Specifies the project and application for the new relationship as well as the display name and key for the new relationship.
<i>File > New > More Projects > Provisioning > Provisioning Request Definition</i>	Specifies the basic information from a workflow template of the provisioning request definition.
<i>File > New > More Projects > Provisioning > Provisioning Team</i>	Enables you to define a set of users who can act as a team for the Requests & Approvals tab of the User Application. The team definition determines who can manage provisioning requests and approval tasks associated with this team.
<i>File > New > More Projects > XML > XML</i>	Creates a new project XML file with specified project files. See Section 23.7, "Preferences: Web and XML," on page 470 .
<i>File > Open File</i>	Enables you to browse to and open a text or XML file.
<i>File > Close</i>	Closes the active editor.
<i>File > Close All</i>	Closes all open editors.
<i>File > Save</i>	Saves changes made in the active editor.
<i>File > Save As</i>	Saves changes made in the active editor to a different file.
<i>File > Save All</i>	Saves changes in all editors.
<i>File > Move</i>	Moves a file, folder, or project to a different location in the file system. Only available in the Navigator view.
<i>File > Rename</i>	Renames a file or project. Only available in the Navigator View.

Option	Description
<i>File > Refresh</i>	Loads the project or file from disk and makes sure it is in sync. (If you add new files outside of the tool directly in the file system, this option displays them in the Project View.)
<i>File > Print</i>	Prints from an editor.
<i>File > Switch Workspace</i>	Opens another workspace with its projects. This option automatically restarts Designer.
<i>File > Import</i>	Imports resources, projects, or preferences. See Section 15.1, "Importing Projects," on page 257 .
<i>File > Import > Project (From File System)</i>	Imports existing project files from selected directories or archive files.
<i>File > Import > Project (From Identity Vault)</i>	Imports existing projects from specified secure entities after entering the required authentication information.
<i>File > Import > Project (From iManager Export File)</i>	Imports an existing Identity Manager project from an iManager export file (driver set or driver).
<i>File > Export</i>	Exports resources or preferences. See Section 19.9, "Exporting a Project," on page 387 .
<i>File > Export > General > Archive File</i>	Exports resources to an archive file on the local file system.
<i>File > Export > General > File System</i>	Exports resources to the local file system.
<i>File > Export > General > Preferences</i>	Exports preferences to the local file system.
<i>File > Export > Designer for Identity Manager > iManager Configuration File</i>	Exports a driver set, driver, channel, or policy to an iManager-compatible configuration file.
<i>File > Properties</i>	Enables you to view the file's full name, date, size, type, and read-only status.
<i>File > History</i>	Lists the history of files and projects opened. By selecting items in this history list, you can reopen a file or project. To configure the length of this history list, use the Preferences dialog box.
<i>File > Exit</i>	Closes the application.

D.2 Options on the Edit Menu



Table D-2 Options: Edit Menu

Option	Description
<i>Edit > Undo Change Location</i>	Undoes the last operation.
<i>Edit > Redo</i>	Redoes the last undone operation.
<i>Edit > Cut</i>	Copies the selected item to the Clipboard and removes it after it is pasted.

Option	Description
<i>Edit > Copy</i>	Copies the selected item to the Clipboard.
<i>Edit > Paste</i>	Pastes the current item in the Clipboard to the selected area.
<i>Edit > Delete</i>	Deletes the selected item
<i>Edit > Select All</i>	Selects all items in a text or XML editor. To select all items in the Modeler, select it, then press Ctrl+A.
<i>Edit > Find/Replace</i>	Replaces text strings in text or XML editors.
<i>Edit > Add Bookmark</i>	Used in text editors.
<i>Edit > Add Task</i>	Adds a task and leaves a mark in the file. Used in text editors.

D.3 Options on the View Menu

Table D-3 *Options: View Menu*

Option	Description
<i>Zoom In</i>	Increases an item's size. You can also slide the Zoom slider in the main toolbar. To reset Zoom to 100%, click the <i>Zoom Restore</i> icon. 
<i>Zoom Out</i>	Decreases an item's size. You can also slide the Zoom slider in the main toolbar. To reset Zoom to 100%, click the <i>Zoom Restore</i> icon. 
<i>Grid</i>	Provides a grid, so that you can align items. See Section 7.11.3, "Using a Grid," on page 92 .
<i>Alignment Hints</i>	Helps you see relationships in your model. You can align or attach items to the left, center, or right of alignment guides. See Section 7.11, "Aligning and Laying Out Components," on page 88 .
<i>Rulers</i>	Turns rulers off and on, to help with alignment and layout. See Section 7.11.2, "Using Rulers," on page 90 .

D.4 Options on the Project Menu

Table D-4 Options: Project Menu

Option	Description
<i>Project > Import Project > File System</i>	Imports an existing project from the file system into Designer from a selected root directory or archive file.
<i>Project > Import Project > Identity Vault</i>	Imports an existing project from an Identity Vault into Designer.
<i>Project > Import Project > iManager Export File</i>	Creates a Designer project by selecting an iManager .xml export file. The .xml file can be a driver set export, driver export, or Novell® preconfiguration file.
<i>Project > Document > New Style</i>	Generates a new style document for a specified project.
<i>Project > Document > Edit Style</i>	Edits a style document for a project located in the Documentation folder under a project in the Project view.
<i>Project > Document > Generate Documentation</i>	Generates project PDF documentation after saving file changes. You can select the directory location where documentation is stored and specify to overwrite existing files and to create XML source files.
<i>Project > Properties</i>	Enables you to discover where the project is stored in the directory structure and when the project was last modified. Provides information on text file encoding and a line delimiter for Windows, Macintosh, or UNIX text files.

D.5 Options on the Tools Menu

Table D-5 Options: Tools Menu

Option	Description
<i>Tools > DS Trace Viewer</i>	Enables you to browse to and open a specified DSTrace log file.
<i>Tools > eDirectory Object Manager</i>	Opens an eDirectory browser to add a new tree.
<i>Tools > iManager</i>	Specifies the Novell® iManager URL, which is stored in the application preferences folder.
<i>Tools > Access User Application</i>	Automatically directs and opens a user application session from within the embedded HTML browser, provided a user application is configured in the Designer project.

Option	Description
<i>Tools > Export Localization Data</i>	Sends the User Application's display labels and e-mail notification templates to an XML or properties file so that the data can be localized.
<i>Tools > Import Localization Data</i>	Imports localization data from a project.

D.6 Options on the Live Menu

Table D-6 *Options: Live Menu*

Option	Description
<i>Live > Browse Identity Vault</i>	Browses specified Identity Vault configurations. See Section 9.3, "Configuring Identity Vaults," on page 148.
<i>Live > Import</i>	Imports driver sets, drivers, or applications. Used in the Modeler and in the Outline view. See Section 15.2, "Importing a Driver Set or Driver from the Identity Vault," on page 266.
<i>Live > Import Attributes</i>	Imports a driver configuration from an already-saved XML file. Section 15.3, "Importing a Driver Configuration File," on page 275.
<i>Live > Import Schema</i>	Imports a schema from an Identity Vault.
<i>Live > Deploy</i>	Deploys driver sets, drivers, or applications. Used in the Modeler and in the Outline view.
<i>Live > Deploy Attributes</i>	Deploys attributes to selected objects.
<i>Live > Deploy Schema</i>	Deploys a schema stored in Designer into an Identity Vault.
<i>Live > Compare</i>	Saves a driver configuration to a <code>.xml</code> file. See Appendix D, "Quickview: Menu Options," on page 535.
<i>Live > Refresh Application Schema</i>	Specifies the server on an eDirectory tree where the schema is refreshed after an application's schema changes. See Section 11.7.2, "Refreshing the Application Schema," on page 219.
<i>Live > Driver Status</i>	Displays each driver's status. Enables you to start, stop, or restart drivers. Enables you to browse, import, and deploy various functionality, depending on the item selected. See Section 9.6, "Configuring Drivers," on page 158.
<i>Live > Start Driver</i>	Starts an inactive driver. See Section 9.6, "Configuring Drivers," on page 158.
<i>Live > Stop Driver</i>	Stops an active driver.
<i>Live > Restart Driver</i>	Restarts an active driver.

Option	Description
<i>Live > Create eDir-to-eDir Certificates</i>	Configures two eDirectory drivers to communicate directly with each other. See Section 21.6, "Configuring TLS for eDir-to-eDir Drivers," on page 427.
<i>Live > Setup Driver Security</i>	Configures security parameters for drivers.

D.7 Options on the Model Menu

Table D-7 Options: Model Menu

Option	Description
<i>Model</i>	Accesses the Modeler applications, tools, drivers, and other resources. See Chapter 7, "Building a Model," on page 53.
<i>Model > Application</i>	Accesses tools to create Designer applications. NOTE: Functionality becomes active when objects in the Designer window are clicked.
<i>Model > Application > New</i>	Creates a new .xml application driver after a driver set is selected in the Modeler window. You can select from a preconfigured list of driver types and run the associated scripts to verify the driver requirements.
<i>Model > Application > Connection Settings</i>	Configures the VNC connectivity settings for objects created in the Modeler window.
<i>Model > Application > Remote Control Desktop</i>	Configures the remote control desktop of the resource if a VNC server is running on it.
<i>Model > Application > Show Subsystems</i>	Shows or hides an application's subsystems. See "Show/Hide Subsystems" on page 563.
<i>Model > Application > Notes</i>	Opens the environment properties window in Modeler to document platform, hardware, and other environment notes about objects you create.
<i>Model > Application > Administrator</i>	Opens the administrator properties window in Modeler to document personal and contact information and other notes about objects you create.
<i>Model > Application > Icon</i>	Opens the general properties window in Modeler to create or edit icons you create for Designer objects. See Section 9.6.7, "Editing or Creating a Driver iManager Icon," on page 164.
<i>Model > Application > Properties</i>	Opens the general properties window for objects selected in Modeler. See Section 9.8, "Configuring Application Properties," on page 170.

Option	Description
<i>Model > Design Element > New</i>	Instructs you to click the Architect tab, click the Design Element drawer in the palette, and drag and drop items into the Modeler.
<i>Model > Design Element > Connection Settings</i>	In Architect mode, launches the Connectivity page for design elements that have VNC connectivity. Enables you to connect to an element or object.
<i>Model > Design Element > Remote Control Desktop</i>	Connects to the application by using the VNC protocol. Displays an editor that you can use to control the application.
<i>Model > Design Element > Administrator</i>	Launches the Design Elements properties page. The Administrator option enables you to capture information about whom to contact concerning the application.
<i>Model > Design Element < Properties</i>	Specifies a design element's properties.
<i>Model > Domain Group > New</i>	Creates a new domain into which selected objects can be grouped in Modeler.
<i>Model > Domain Group > Clear Domain Contents</i>	Clears objects that are contained in a domain.
<i>Model > Domain Group > Add to Group</i>	Adds selected objects to a new group.
<i>Model > Domain Group > Ungroup</i>	Ungroups selected objects contained within a domain without deleting them.
<i>Model > Domain Group > Properties</i>	Specifies properties of domains contained within a group.
<i>Model > Driver > New</i>	Creates a new Driver object, specified from a list of templates.
<i>Model > Driver > Copy > Server-Specific Settings</i>	Copies replica data from one server to another server that is associated with the selected driver.
<i>Model > Driver > Copy > Settings</i>	Copies existing driver settings.
<i>Model > Driver > Simulate</i>	Opens the Simulate Policy Transformation - Entitlements Service Driver wizard to test the driver's set-up parameters.
<i>Model > Driver > Log Level</i>	Defines the level of errors that should be tracked and reported in the error log. See Section 9.6.3, "Driver Log Level," on page 160 .
<i>Model > Driver > Global Configuration Values</i>	Specifies settings for Identity Manager features in drivers and driver sets, including password synchronization values, driver heartbeat, and specific driver configuration. See Section 9.6.2, "Driver Global Configuration Values," on page 160 .
<i>Model > Driver > Trace Settings</i>	Adds a trace level to your driver to display the Identity Manager (DirXML) events as the engine processes the events. For a description of each trace field and platform requirements, see Section 9.5.5, "Driver Set Trace," on page 156 .

Option	Description
<i>Model > Driver > Named Passwords</i>	Stores multiple passwords securely for each driver, which can be can be referenced in a driver policy. See Section 9.6.6, "Driver Named Passwords," on page 163.
<i>Model > Driver > Driver Configuration</i>	Launches the Driver Configuration Wizard, provided a driver is selected.
<i>Model > Driver > iManager Icon</i>	Displays an application's icon used in iManager, provided a driver is selected.
<i>Model > Driver > Engine Controls</i>	Displays the Engine Control Values property page. Provides a list of associated servers and authentication credentials.
<i>Model > Driver > Manifest</i>	Specifies what the driver supports and a few configuration settings. See Section 9.6.5, "Driver Manifest," on page 163.
<i>Model > Driver > Add Entitlements</i>	Launches the entitlement wizard, which allows you to set up criteria for a person or group that, once met, initiate an event to grant or revoke access to business resources within the connected system. See Chapter 17, "Managing Entitlements," on page 339.
<i>Model > Driver > Edit Entitlements</i>	Enables you to change an entitlement's name, values, multi-value, and role-based entitlements.
<i>Model > Driver > Mark as Firewall</i>	Indicates that the selected driver is communicating across a firewall.
<i>Model > Driver > Password Synchronization</i>	Synchronizes passwords between connected systems. Section 12.6, "Integrating Passwords," on page 245 and "Password Synchronization across Connected Systems" in the Novell Identity Manager Administration Guide (http://www.novell.com/documentation/idm35/index.html) .
<i>Model > Driver > Credential Provisioning > Credential Repository</i>	Creates a driver repository.
<i>Model > Driver > Credential Provisioning > Credential Application</i>	Creates a credential application.
<i>Model > Driver > Icon</i>	Displays driver icons, or turns icons on or off. See Section 9.6.7, "Editing or Creating a Driver iManager Icon," on page 164.
<i>Model > Driver > Properties</i>	Launches the driver's Properties page.
<i>Model > Driver Set > New</i>	Launches a wizard to create a new driver set.
<i>Model > Driver Set > Copy > Driver Set Settings</i>	Copies existing driver set settings.

Option	Description
<i>Model > Driver Set > Copy > Global Configuration Values</i>	Copies settings for Identity Manager features such as password synchronization and driver heartbeat, as well as settings that are specific to the function of an individual driver configuration. See Section 9.6.2, "Driver Global Configuration Values," on page 160.
<i>Model > Driver Set > Import from File</i>	Imports an Identity Manager driver configuration file into the selected driver set for a project. See Section 15.3, "Importing a Driver Configuration File," on page 275.
<i>Model > Driver Set > Log Level</i>	Defines the level of errors that should be tracked. See Section 9.5.3, "Driver Set Log Levels," on page 154.
<i>Model > Driver Set > Server List</i>	Displays servers that the driver set is associated with as well as available servers.
<i>Model > Driver Set > Global Configuration Values</i>	Specifies settings for Identity Manager features such as password synchronization and driver heartbeat, as well as settings that are specific to the function of an individual driver configuration. See Section 9.6.2, "Driver Global Configuration Values," on page 160.
<i>Model > Driver Set > Trace Settings</i>	Sets the trace level to your driver set, which displays the Identity Manager (DirXML) events as the engine processes the events. See Section 9.5.5, "Driver Set Trace," on page 156.
<i>Model > Driver Set > Properties</i>	Launches the driver set Properties page. See Section 9.1, "Viewing Object Properties," on page 145.
<i>Model > ECMAScript > New</i>	Creates an ECMAScript object.
<i>ECMAScript > New from Copy</i>	Creates an ECMAScript object by copying an existing ECMAScript object.
<i>Model > eDir-to-eDir > New</i>	Launches the eDirectory-to-eDirectory wizard to create a new connection between two eDirectory objects.
<i>Model > eDir-to-eDir > Secure Connection Settings</i>	Opens the TLS Configuration window to enable SSL/TLS connection settings and encrypted key settings through advanced TLS configuration.
<i>Model > eDir-to-eDir > Disconnect eDir-to-eDir</i>	Disconnects the eDirectory-to-eDirectory connection.
<i>Model > Identity Vault > New</i>	Adds a new server to an Identity Vault.
<i>Model > Identity Vault > Manage Schema</i>	Opens a context-sensitive window to manage schema classes and attributes of the Identity Vault.
<i>Model > Identity Vault > Duplicate Template</i>	Opens an e-mail template list that you can duplicate.

Option	Description
<i>Model > Identity Vault > Edit Template</i>	Opens a template list to select and edit an e-mail template.
<i>Model > Identity Vault > E-Mail Server Properties</i>	Launches the e-mail notification server so you can enter notification settings and server authentication credentials.
<i>Model > Identity Vault > Administrator</i>	Opens the Administrator properties window of the identity vault to enter personal and contact information and other optional information.
<i>Model > Identity Vault > Server List</i>	Launches a wizard that displays the servers associated with a selected identity vault, which enables you to edit the properties and contact information. See “Driver Set Server List” on page 155 .
<i>Model > Identity Vault > Properties</i>	Opens the Properties window for the identity vault to access configuration, administrator, and server list properties. See Section 9.3, “Configuring Identity Vaults,” on page 148 .
<i>Model > Job > New</i>	Launches the Job Scheduler Wizard, to create a job.
<i>Model > Job > New from Copy</i>	Creates a job by copying an existing job.
<i>Model > Library > New</i>	Launches the New Library Wizard to create a library after you have selected an Identity Vault or a driver set.
<i>Model > Policy > New DirXML Script</i>	Opens a new Policy Name window to enable you to create XML policy scripts.
<i>Model > Policy > New Schema Mapping</i>	Opens a schema map editor to associate identity objects with associated service drivers. See Chapter 11, “Managing the Schema,” on page 187 .
<i>Model > Policy > New XSLT</i>	Opens a new Policy Name window to enable you to create Extensible Stylesheet Language Transformation (XSLT) policy scripts.
<i>Model > Policy > New from Copy</i>	Opens the Modeler browser to copy selected existing Driver Set objects.
<i>Model > Policy > Manage Policy</i>	Launches a policy manager GUI that exposes individual elements of a project design tree that can be edited in context by right-clicking each element.
<i>Model > Resource > New</i>	Prompts you to add new resource objects from the Modeler palette. After resource objects are added, you set up connection and control settings.
<i>Model > Server > New</i>	Associates a new eDirectory- or manually-based server to an Identity Vault.
<i>Model > Select Connected Applications</i>	Selects driver connections between application objects in modeler.
<i>Model > Clear All Items</i>	Enables you to clear all items in a project.

Option	Description
<i>Model > Straighten Connections</i>	Straightens line connections between selected objects in the Modeler.
<i>Model > Distribute</i>	Distributes selected objects in the Modeler either vertically or horizontally.
<i>Model > Align</i>	Aligns selected objects in the Modeler: top, bottom, left, right, center, or middle.
<i>Model > Snapshot to Image File</i>	In Developer or Architect mode, captures the model and stores it in a .bmp, .jpg, or .jpeg file.
<i>Model > Snapshot to Image File</i>	Saves a bitmap or JPEG snapshot image to a specified file in the <code>Designer driver_configs</code> directory.

D.8 Options on the Window Menu

Table D-8 Options: Window Menu

Option	Description
<i>Window > New Window</i>	Opens a new instance of the application, making it easier to copy and paste between certain elements. You need to synchronize changes between the instances.
<i>Window > New Editor</i>	Launches a new Designer editor.
<i>Window > Open Perspective > Designer</i>	Displays when Designer is the default perspective.
<i>Window > Open Perspective > Other</i>	Toggles among perspectives: Debug, Designer (default), exteNd Perspective, and Resource.
<i>Window > Show View > Dataflow</i>	Controls the flow of data between objects. See Section 12.1, “The Dataflow View,” on page 221 and Section 7.3.3, “Dataflow Mode,” on page 61 .
<i>Window > Show View > Navigator</i>	Displays the Navigator view. See Section 8.6, “The Navigator View,” on page 135 .
<i>Window > Show View > Outline</i>	Provides an outline view of the active editor (Modeler, XML Editor, or a text editor).
<i>Window > Show View > Policy Set</i>	Displays the Policy Set view. When you select a driver in the Modeler, this view displays the driver's policies. See Section 8.7, “The Policy Set View,” on page 138 .
<i>Window > Show View > Project</i>	Displays the main view for manipulating your project. See Chapter 6, “Creating a Project,” on page 49 .

Option	Description
<i>Window > Show View > Project Checker</i>	Checks Designer objects for any problems that might cause the deployment of a project into the Identity Vault to fail. It does not check the current objects in the Identity Vault. See Section 21.4, "Checking Your Projects," on page 414.
<i>Window > Show View > Provisioning View</i>	Displays entities, lists, and organizational charts; used in workflow-based provisioning.
<i>Window > Show View > Search Results</i>	Displays the Search Results view. See also Section 8.9, "The Search Results View," on page 142.
<i>Window > Show View > Tasks</i>	Enables you to add your own tasks and notes as you do your work. You can assign them a priority; mark them as completed; or delete, sort, or filter them.
<i>Window > Show View > Trace</i>	Displays internal errors and messages, so that you can find out why something isn't working as expected. Use this feature to provide information for Novell® Support, engineers, or other consulting resources. See "Trace" on page 467.
<i>Window > Show View > Other > General > Bookmarks</i>	Enables you to see bookmarks that you have set in your text editors.
<i>Window > Show View > Other > General > Console</i>	Displays the Console view.
<i>Window > Show View > Other > General > Internal Web Browser</i>	Enables you to immediately access the Eclipse Web site, or otherwise access the Internet.
<i>Window > Show View > Other > General > Navigator</i>	Displays the Navigator view. See Section 8.6, "The Navigator View," on page 135.
<i>Window > Show View > Other > General > Outline</i>	Displays the Outline view.
<i>Window > Show View > Other > General > Problems</i>	Displays the Problems view, with errors and warnings used for troubleshooting.
<i>Window > Show View > Other > General > Project Explorer</i>	Displays the project components.
<i>Window > Show View > Other > General > Properties</i>	Enables you to view and edit settings on the selected object.
<i>Window > Show View > Other > General > Tasks</i>	Enables you to list, prioritize, and track tasks
<i>Window > Show View > Other > Debug > Breakpoints</i>	Displays breakpoints in the active Designer project code.
<i>Window > Show View > Other > Debug > Debug</i>	Debugs the active Designer project.
<i>Window > Show View > Other > Debug > Expressions</i>	Displays all of the expressions available to the debugging filter.
<i>Window > Show View > Other > Debug > Memory</i>	Monitors the memory allocation used by a project.
<i>Window > Show View > Other > Debug > Registers</i>	Displays the Registers editor.
<i>Window > Show View > Other > Debug > Variables</i>	Displays the driver configuration variables.

Option	Description
<i>Window > Show View > Other > Designer for Identity Manager > Dataflow</i>	Controls the flow of data between objects. See Section 12.1, “The Dataflow View,” on page 221 and Section 7.3.3, “Dataflow Mode,” on page 61.
<i>Window > Show View > Other > Designer for Identity Manager > eDirectory Browser</i>	Opens an eDirectory Browser window from which you can specify the directory you want to browse. See Section 21.5.1, “Tool-Based Method,” on page 424.
<i>Window > Show View > Other > Designer for Identity Manager > Policy Set</i>	Displays all policies contained in the selected policy set. See Section 8.7, “The Policy Set View,” on page 138.
<i>Window > Show View > Other > Designer for Identity Manager > Project</i>	Displays the main view for manipulating your project. See Chapter 6, “Creating a Project,” on page 49.
<i>Window > Show View > Other > Designer for Identity Manager > Project Checker</i>	Checks Designer objects for any problems that might cause the deployment of a project into the Identity Vault to fail. It does not check the current objects in the Identity Vault. See Section 21.4, “Checking Your Projects,” on page 414.
<i>Window > Show View > Other > Designer for Identity Manager > Search Results</i>	Displays the Search Results view. See also Section 8.9, “The Search Results View,” on page 142.
<i>Window > Show View > Other > Designer for Identity Manager > Trace</i>	Displays internal errors and messages, so that you can find out why something isn’t working as expected. Use this feature to provide information for Novell® Support, engineers, or other consulting resources. See “Trace” on page 467.
<i>Window > Show View > Other > Help > Help</i>	Launches context-sensitive help in the Help view. Help topics depend on which part of the Modeler or Designer is active.
<i>Window > Show View > Other > PDE Runtime > Error Log</i>	Used for troubleshooting. See Section 24.1, “Viewing the Error Log,” on page 491.
<i>Window > Show View > Other > PDE Runtime > Plug-in Registry</i>	Displays the plug-in registry. Enables you to see which plug-ins are registered or running. See Section 24.3, “Checking Loaded Plug-Ins,” on page 494.
<i>Window > Show View > Other > Provisioning</i>	Enables you to view and configure entities, lists, and organizational charts. See Configuring the Directory Abstraction Layer (http://www.novell.com/documentation/idm/index.html?page=/documentation/idm/agpro/data/agpropartconfigappenv.html) in the <i>Identity Manager User Application: Administration Guide</i> (http://www.novell.com/documentation/idm/index.html?page=/documentation/idm/agpro/data/agpropartconfigappenv.html).
<i>Window > Show View > Other > Provisioning > Data Item Mapping</i>	Specifies how object attributes are mapped. See Section 11.8, “Mapping eDirectory to an LDAP Schema,” on page 220.

Option	Description
<i>Window > Show View > Other > Provisioning > E-Mail Notification</i>	Displays an e-mail notification map.
<i>Window > Show View > Other > Provisioning > Provisioning View</i>	Displays entities, lists, and organizational charts; used in workflow-based provisioning.
<i>Window > Show View > Other > XML > XPath Navigator</i>	Supports context-sensitive editing of XPath expressions. Automatically attaches to the currently selected XML editor.
<i>Window > Customize Perspective</i>	Configures the layout of a perspective. Controls menus (New, Open Perspective, and Show View) and action sets that show up on the tool bar. See Section 8.2, "Managing Perspectives," on page 105 .
<i>Window > Customize Perspective > Shortcuts</i>	Lists the plug-in shortcuts that you can add as cascade submenus on the current perspective.
<i>Window > Customize Perspective > Shortcuts > Designer Document Generation</i>	Enables addition of a .docgen shortcut to create a new document generation style.
<i>Window > Customize Perspective > Shortcuts > General</i>	Enables selection of shortcuts to create a new file, folder, or empty untitled text file resources.
<i>Window > Customize Perspective > Shortcuts > Novell Identity Manager</i>	Enables the addition of the Identity Manager Project shortcut to create a new project.
<i>Window > Customize Perspective > Shortcuts > Provisioning</i>	Enables selection of shortcuts to create new directory abstraction layer entities, lists, relationships, organizational charts provisioning request definitions.
<i>Window > Customize Perspective > Shortcuts > XML</i>	Enables addition of an XML shortcut.
<i>Window > Customize Perspective > Commands</i>	Alphabetically lists the command group plug-ins that you can add as cascade submenus on the current perspective. The Details field identifies which menu item organizational charts toolbar items are added to the perspective by the selected command group. See "Using Menus to Open a Perspective" on page 106 .
<i>Window > Customize Perspective > Commands > Breakpoints</i>	Adds breakpoints functionality to the menu.
<i>Window > Customize Perspective > Commands > com.novell.designer.idm.actionSet</i>	Adds Help set functionality to the menu: Check for Designer Updates, What's New, Designer Readme, Report a Bug or Feedback, and Identity Manager Resources.
<i>Window > Customize Perspective > Commands > Debug</i>	Adds Debug plug-in functionality to the menu.
<i>Window > Customize Perspective > Commands > Deploy</i>	Enables the Deploy plug-in command. See "Deploying the Schema into the eDirectory Tree" on page 204 .

Option	Description
<i>Window > Customize Perspective > Commands > Designer</i>	Adds standard Designer commands to the toolbar: Save All, Delete selected object(s), Undo, and Redo.
<i>Window > Customize Perspective > Commands > Designer Projects ActionSet</i>	Adds Designer project import control options to the menu.
<i>Window > Customize Perspective > Commands > Editor</i>	Adds cut, copy, and paste to the menu and toolbar.
<i>Window > Customize Perspective > Commands > Editor Navigation</i>	Adds Last Edit Location navigation to the menu and toolbar.
<i>Window > Customize Perspective > Commands > Editor Presentation</i>	Adds toggle to Show Selected Element Only on the toolbar.
<i>Window > Customize Perspective > Commands > Generate Documentation</i>	Adds Generate Documentation for this project to the toolbar.
<i>Window > Customize Perspective > Commands > Identity Vault</i>	Adds Identity Vault resource information for a project to the menu.
<i>Window > Customize Perspective > Commands > Insert</i>	Enables the Insert plug-in command.
<i>Window > Customize Perspective > Commands > Keyboard Shortcuts</i>	Adds keyboard shortcuts to the menu.
<i>Window > Customize Perspective > Commands > Launch</i>	Adds Run program functionality to the menu and Launch plug-in to the toolbar.
<i>Window > Customize Perspective > Commands > Manage Objects in a Novell eDirectory Tree</i>	Adds the eDirectory Object Manager to the menu.
<i>Window > Customize Perspective > Commands > Manage Objects in a Novell eDirectory Tree</i>	Adds the Live > Browse Identity Vault functionality to the menu.
<i>Window > Customize Perspective > Commands > Open Files</i>	Adds the browse to and open a text or XML file plug-in to the menu.
<i>Window > Customize Perspective > Commands > Profile</i>	Adds the Run > Profile tools on the menu and toolbar.
<i>Window > Customize Perspective > Commands > Project Actions</i>	Launches the project snap-in.
<i>Window > Customize Perspective > Commands > Project Checker</i>	Places the Project Checker icon on Designer's toolbar.
<i>Window > Customize Perspective > Commands > Resource Navigation</i>	Adds Navigation tools as an option on the menu.
<i>Window > Customize Perspective > Commands > Run the User Application</i>	Adds the Access User Application function as an option on the Tools menu.
<i>Window > Customize Perspective > Commands > Tools</i>	Adds the Tools function as an option on the Project menu.
<i>Window > Customize Perspective > Commands > View DSTrace Logs</i>	Adds the DSTrace function as an option on the Tools menu.

Option	Description
<i>Window > Customize Perspective > Commands > WebBrowser</i>	Adds the Launch Web Browser wizard on the Web browser toolbar.
<i>Window > Customize Perspective > Commands > WebBrowser</i>	Adds the Launch Web Browser wizard on the Window menu.
<i>Window > Save Perspective As</i>	An easy, efficient way to lay out the current perspective in a certain way and then save it. You can freely switch between perspectives and still operate in the same editors and views.
<i>Window > Reset Perspective</i>	Returns the perspective to its original layout. If the perspective is a copied perspective, it still returns to the original, preferred layout of the base perspective.
<i>Window > Close Perspective</i>	Closes the current perspective.
<i>Window > Close All Perspectives</i>	Closes all open perspectives.
<i>Window > Navigation > Show System Menu</i>	Displays the active system menu.
<i>Window > Navigation > Show View Menu</i>	When active, shows the View menu.
<i>Window > Navigation > Maximize Active View or Editor</i>	Maximizes the active view or editor. Ctrl+M toggles between views.
<i>Window > Navigation > Minimize Active View or Editor</i>	Minimizes the active view or editor. Ctrl+M toggles between views.
<i>Window > Navigation > Activate Editor</i>	Enables you to select which editor to activate. F12 is the hotkey.
<i>Window > Navigation > Next Editor</i>	Allows you to select and open an editor from a list. Ctrl+F6 navigates down this list.
<i>Window > Navigation > Previous Editor</i>	Allows you to select and open an editor from a list. Ctrl+Shift+F6 navigates up this list.
<i>Window > Navigation > Switch to Editor</i>	Allows you to select and manage editors.
<i>Window > Navigation > Quick Switch Editor</i>	Enables you to enter or select which editor to activate. Ctrl+Shift+E is the hotkey.
<i>Window > Navigation > Next View</i>	Allows you to select and open a view from a list. Ctrl+F7 navigates down this list.
<i>Window > Navigation > Previous View</i>	Allows you to select and open a view from a list. Ctrl+Shift+F7 navigates up this list.
<i>Window > Navigation > Next Perspective</i>	Allows you to toggle perspectives within projects from a list. Ctrl+F8 is the hotkey.
<i>Window > Navigation > Previous Perspective</i>	Allows you to view previous perspectives from a list. Ctrl+Shift+F8 is the hotkey.
<i>Window > Launch Web Browser</i>	Specifies your Home Page URL, which is stored in stored in Preferences.

Option	Description
<i>Window > Preferences</i>	<p>Enables you to customize Designer's behavior in the following categories:</p> <ul style="list-style-type: none"> ◆ General ◆ Designer for IDM ◆ Help ◆ Noell ◆ Provisioning ◆ Web and XML <p>See Chapter 23, "Setting Preferences," on page 441.</p>

D.9 Options on the Help Menu

Table D-9 *Options: Help Menu*

Option	Description
<i>Help > Welcome</i>	Displays the Welcome page. See Section 5.1, "Help from the Welcome Page," on page 41.
<i>Help > Help Contents</i>	Enables you to navigate to and view information on all features and functions in Designer.
<i>Help > Search</i>	Locates instances in the help system and provides links.
<i>Help > Dynamic Help</i>	Provides context-sensitive information.
<i>Help > Key Assist</i>	Lists shortcuts.
<i>Help > Check for Designer Updates</i>	Enables you to quickly download and install the latest Designer updates.
<i>Help > What's New</i>	Directs you to a Novell® Web page that describes new features in Designer.
<i>Help > Designer Readme</i>	Provides updated information on known issues.
<i>Help > Report a Bug or Give Feedback</i>	Allows you to enter an Identity Manager Designer bug or provide feedback into Bugzilla.
	<p>NOTE: You must first establish a Novell User Account (https://secure-www.novell.com/selfreg/jsp/createAccount.jsp) to provide feedback.</p>
<i>Help > Identity Manager Forums</i>	Provides a link to Identity Manager product support forums.
<i>Help > Identity Manager Resources</i>	Provides a link to the Designer Resource Page.
<i>Help > About Designer</i>	Provides information about the configuration, product, license, and plug-in.

Modeler Operations

E

Modeler operations are available when you right-click inside the Modeler. The list of operations depends upon whether you right-click the Modeler space or one of the objects.

- ◆ Section E.1, “Modeler Space Operations,” on page 553
- ◆ Section E.2, “Identity Vault Operations,” on page 554
- ◆ Section E.3, “Driver Set Operations,” on page 557
- ◆ Section E.4, “Driver Operations,” on page 559
- ◆ Section E.5, “Application Operations,” on page 562
- ◆ Section E.6, “Submenus,” on page 565
- ◆ Section E.7, “Keyboard Support,” on page 566

E.1 Modeler Space Operations

The following figure illustrates Modeler options that are available when you right-click empty Modeler space.

Figure E-1 Modeler-Space Operations



Table E-1 Modeler-Space Operations

Right-Click Operation	Description
<i>Undo Change Location</i>	Returns an item to its previous status.
<i>New > Application</i>	Selects an application from a list and places the application to the Modeler.
<i>New > Domain Group</i>	Places a Domain Group in the Modeler.

Right-Click Operation	Description
<i>New > Identity Vault</i>	Launches a dialog box that specifies a server and creates an Identity Vault.
<i>Straighten Connections</i>	Straightens lines for selected items. For example, you can straighten a line to a driver, all lines in a driver set, everything in a Domain Group, or an entire project. If a line is not within a few degrees of being horizontal or vertical, this option is dimmed.
<i>Distribute</i>	Evenly distributes applications vertically or horizontally. Press Ctrl, select the items that you want to distribute, then select a pattern.
<i>Align</i>	Aligns applications according to a pattern that you select. Press Ctrl, select the items, then select a pattern (for example, Align Bottom).
<i>Document Selection</i>	Launches the Document Generation Wizard, which documents your project.
<i>Live > Import</i>	Imports and Identity Vault.

E.2 Identity Vault Operations

The following figure illustrates Modeler operations that are available when you right-click an Identity Vault.

Figure E-2 Identity-Vault Operations

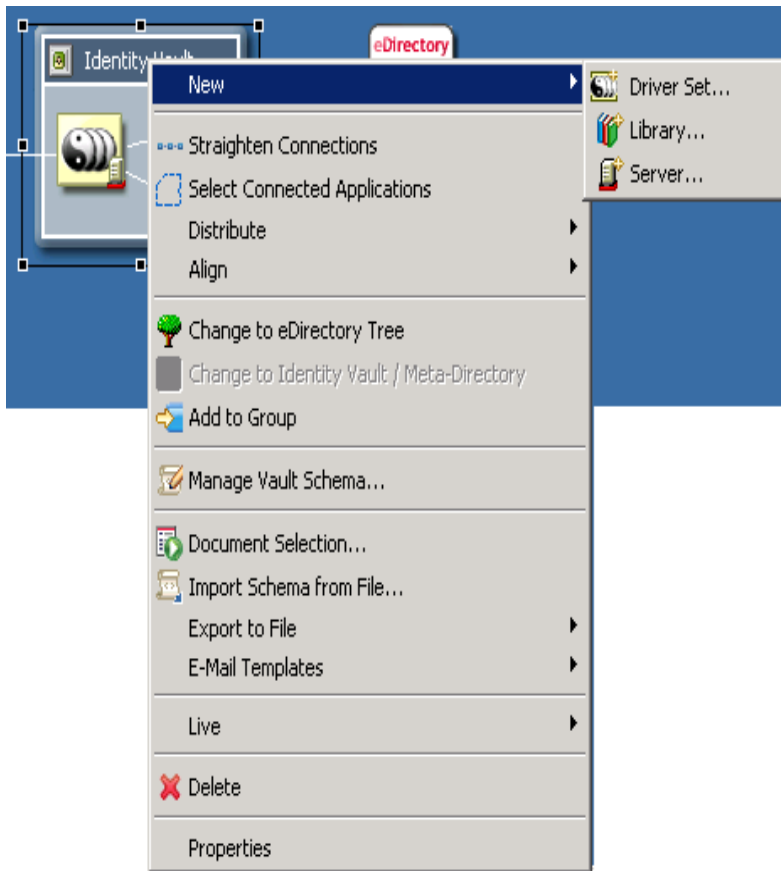


Table E-2 Identity Vault Operations

Operation	Description
<i>New > Driver Set</i>	Adds a Driver Set object to an Identity Vault.
<i>New > Library</i>	Launches the New Library Wizard.
<i>New > Server</i>	Launches the Add Server Wizard.
<i>Straighten Connections</i>	Straightens lines for selected items. For example, you can straighten a line to a driver, all lines in a driver set, everything in a Domain Group, or an entire project. If a line is not within a few degrees of being horizontal or vertical, this option is dimmed.
<i>Select Connected Applications</i>	Selects all applications that are connected to the driver set or Identity Vault. This is convenient if you have several applications connected to a driver set. You can quickly move them all or delete them all without browsing to and selecting each one.
<i>Distribute</i>	Evenly distributes applications vertically or horizontally. Press Ctrl, select the items that you want to distribute, then select a pattern.

Operation	Description
<i>Align</i>	Aligns applications according to a pattern that you select. Press Ctrl, select the items, then select a pattern (for example, Align Bottom). See Table E-6 on page 565 .
<i>Change to eDirectory Tree</i>	Changes an Identity Vault to an eDirectory™ tree. In Architect mode, this option displays a tree instead of a vault. This is just for diagramming purposes; there is no functional difference.
<i>Change to Identity Vault/Metadirectory</i>	Changes an eDirectory tree into an Identity Vault. In Developer mode, this option displays a vault instead of a tree. This is just for diagramming purposes; there is no functional difference.
<i>Add to Group</i>	Creates a Domain Group, and adds the selected items to it. The selected items are removed from any group to which they were previously associated.
<i>Manage Vault Schema</i>	Launches the Schema Manage tool, from which you can manipulate schema settings for the selected Identity Vault or directory.
<i>Document Selection</i>	Launches the Document Generation Wizard, which documents the selected Identity Vault.
<i>Import Schema from File</i>	Enables you to browse to a file and import a schema into a .sch or .ldif file.
<i>Export to File > Configuration</i>	Exports the Identity Vault to a .xml file. iManager can consume this format, and Designer can re-import it. For more information, see Section 19.10, “Exporting To a File,” on page 390 .
<i>Export to File > Schema</i>	Exports the schema to a .sch or .ldif file.
<i>E-mail Templates > E-Mail Server Properties</i>	Configures an e-mail server to send e-mail notifications. Edits templates used to notify users concerning password events. For more information, see Section 14.5, “Configuring the E-Mail Server,” on page 256 .
<i>E-Mail Templates > Edit Templates</i>	Launches the E-mail Templates dialog box, from which you can edit the e-mail templates associated with the selected Identity Vault. For more information, see Chapter 14, “Setting Up E-Mail Notification Templates,” on page 249 .
<i>E-Mail Templates > Update Templates</i>	Adds localized templates to the Default Notification Collection.
<i>Live > Import</i>	Enables you to connect to a server, browse to and select objects, and import the objects into the Identity Vault.
<i>Live > Deploy</i>	Prepares a deployment summary and then deploys selected objects and attributes.
<i>Live > Compare</i>	.Compares selected Identity Vaults. Enables you to reconcile or update Identity Vaults. See Section 19.7, “Using the Compare Feature When Deploying,” on page 383 .
<i>Live > Schema > Import</i>	Imports the schema from an existing Identity Vault.
<i>Live > Schema > Deploy</i>	Deploys the modified or imported schema.
<i>Live > Open iManager</i>	.Enables you to connect to a server and launch iManager.

Operation	Description
<i>Live > Browse Identity Vault</i>	Launches the eDirectory Object Manager, which allows you to view, and edit attributes for, the selected Identity Vault. For more information, see Section 21.5, “Managing Directory Objects,” on page 423 .
<i>Live > Driver Status</i>	Lists drivers that are stopped or running.
<i>Live > Start Drivers</i>	Starts all drivers associated with the selected object.
<i>Live > Stop Drivers</i>	Stops all drivers associated with the selected object.
<i>Live > Restart Drivers</i>	Restarts all drivers associated with the selected object.
<i>Delete</i>	.Deletes the Identity Vault.
<i>Properties</i>	.Displays the Identity Vault’s properties pages.

E.3 Driver Set Operations

The following figure illustrates Modeler operations that are available when you right-click a driver set.

Figure E-3 *Driver Set Operations*

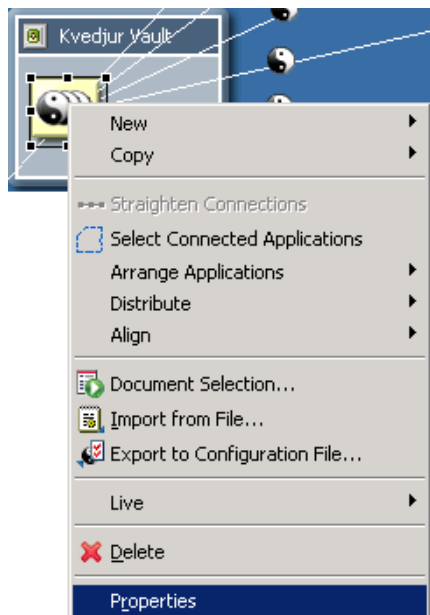


Table E-3 *Driver Set Operations*

Operation	Description
<i>New > Driver</i>	Launches the Driver Configuration Wizard, to add a driver to the driver set.
<i>New > Job</i>	Launches the Job Scheduler Wizard, to create a job.
<i>New > Library</i>	Launches the New Library Wizard.

Operation	Description
<i>Copy > Driver Set Settings</i>	Enables you to browse to a driver set and copy its settings. A pasted copy overwrites data in the target driver set.
<i>Copy > Global Configuration Values</i>	Enables you to copy Global Configuration Values (GCV) from one driver set to one or more other driver sets. This option enables you to configure GCVs in one place and then apply GCV settings to selected targets.
<i>Straighten Connections</i>	Straightens all lines in the driver set. If a line is not within a few degrees of being horizontal or vertical, this option is dimmed.
<i>Select Connected Applications</i>	Selects all applications that are connected to the driver set. You can quickly move or delete the applications without browsing to and selecting each one.
<i>Arrange Applications</i>	Arranges application icons around their associated driver set icon. A check mark indicates the current layout for the driver set. After the layout is set, any applications that you connect are automatically snapped into that layout. For more information, see Table E-7 on page 565 .
<i>Distribute</i>	Evenly distributes applications vertically or horizontally. Press Ctrl, select the items that you want to distribute, then select a pattern.
<i>Align</i>	Aligns applications according to a pattern that you select. Press Ctrl, select the items, then select a pattern (for example, Align Bottom). See Table E-6 on page 565 .
<i>Document Selection</i>	Launches the Document Generation Wizard, which documents the selected driver set.
<i>Import from File</i>	Reads in exports made from iManager or Designer. For more information, see Section 15.3, "Importing a Driver Configuration File," on page 275 .
<i>Export to Configuration File</i>	Exports the driver set to a .xml file. iManager can consume this format, and Designer can re-import it. For more information, see Section 19.10, "Exporting To a File," on page 390 .
<i>Live > Import</i>	Enables you to connect to a server, browse to and select objects, and import the objects into the driver set.
<i>Live > Deploy</i>	Prepares a deployment summary and then deploys selected objects and attributes.
<i>Live > Compare</i>	.Compares selected driver sets. Enables you to reconcile or update driver sets. See Section 19.7, "Using the Compare Feature When Deploying," on page 383 .
<i>Live > Driver Set Configuration > Import Attributes</i>	Imports attributes from an existing driver set.
<i>Live > Driver Set Configuration > Deploy Attributes</i>	Deploys the modified or imported attributes
<i>Live > Driver Status</i>	Lists drivers that are stopped or running.
<i>Live > Start Drivers</i>	Starts all drivers associated with the selected object.

Operation	Description
<i>Live > Stop Drivers</i>	Stops all drivers associated with the selected object.
<i>Live > Restart Drivers</i>	Restarts all drivers associated with the selected object.
<i>Delete</i>	Deletes the driver set.
<i>Properties</i>	Enables you to configure Identity Vaults, driver sets, drivers, and applications.

E.4 Driver Operations

The following figure illustrates Modeler operations that are available when you right-click a driver.

Figure E-4 *Driver Operations*

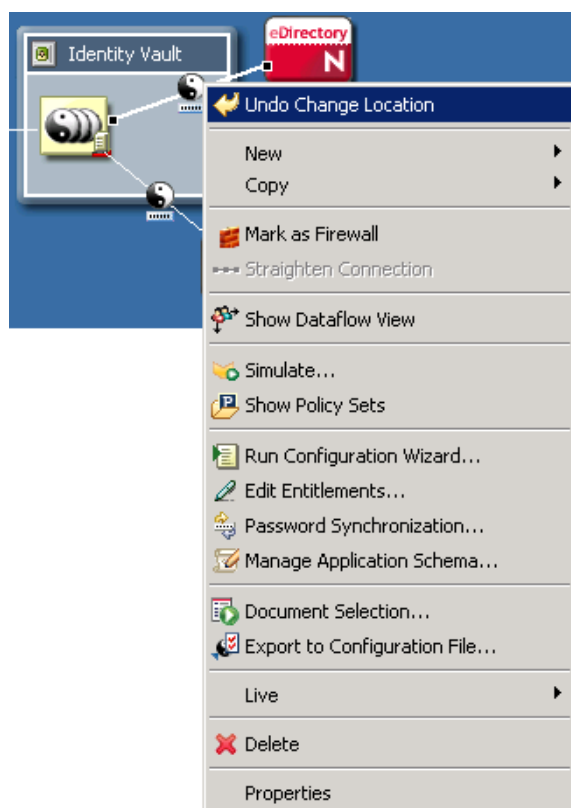


Table E-4 *Driver Operations*

Operation	Description
<i>Undo Change Location</i>	Returns an item to its previous status.
<i>New > Credential Application</i>	Creates a new Application object, which stores static single sign-on parameters for a specific application. For more information, see <i>Novell Credential Provisioning Policies for Identity Manager 3.5.1</i> .

Operation	Description
<i>New > Credential Repository</i>	Creates a new Repository Object, which stores static configuration information for an authentication credential repository; either Novell SecretStore or Novell SecureLogin. For more information, see “ Overview ” in the <i>Novell Credential Provisioning Policies for Identity Manager 3.5.1</i>
<i>New > DirXMLScript</i>	Launches the Policy Builder, creates a policy, and creates a new DirXML script. DirXML Script is the primary method of implementing policies in the Novell Identity Manager Metadirectory engine.
<i>New > ECMAScript</i>	Creates an ECMAScript object and opens the ECMAScript editor.
<i>New > Entitlement</i>	Launches the Entitlement Wizard and adds an entitlement to the selected driver. For more information, see Chapter 17, “Managing Entitlements,” on page 339 .
<i>New > Job</i>	Launches the Job Scheduler Wizard, to create a job.
<i>New > Mapping Table</i>	Creates a Mapping Table object. A policy uses a Mapping Table object to map one set of values to another set of corresponding values.
<i>New > Resource</i>	Creates a Resource object. Resource objects (for example, generic, ECMAScript, mapping table, application, or repository resources) store information that drivers use. The information can be arbitrary data in any format (for example, XML or text)..
<i>New > Schema Map</i>	Creates a schema map policy and launches the Schema Map editor. A schema map policy maps class names and attribute names between the Identity Vault namespace and the application namespace. The schema map policy is applied in both directions.
<i>New > XSLT</i>	Creates an XSLT policy. XSLT is a standard language for transforming XML documents. You can use the XSLT option to implement policies as XSLT style sheets.
<i>New > From Copy</i>	Creates a policy by copying from an existing object.
<i>Copy > Settings</i>	Copies data from the selected driver to a target driver. A pasted copy overwrites data in the target driver.
<i>Copy > Server-Specific Settings</i>	Copies data from the selected server to a target server. A pasted copy overwrites data in the target server.
<i>Mark/Unmark as Firewall</i>	Enables you to mark where a driver is communicating through a firewall. Used in Developer mode. If driver icons are turned off, the firewall icon doesn't appear.
<i>Straighten Connection</i>	Straightens a driver connection line. If a line is not within a few degrees of being horizontal or vertical, this option is dimmed.
<i>Show Dataflow View</i>	Displays the flow of information between the application and the driver in the Developer view. Launches the Dataflow view. For more information, see Chapter 12, “Managing the Flow of Data,” on page 221 .

Operation	Description
<i>Dataflow</i>	(Viewable when the Dataflow view is active.) Displays the flow of information between the application and the driver in the Developer view. Launches the Dataflow view. For more information, see Chapter 12, “Managing the Flow of Data,” on page 221 and Table E-8 on page 565.
<i>Simulate</i>	Runs the Simulate Policy Transformation program against the selected driver.
<i>Show Policy Sets</i>	Launches the Policy Set and Policy Flow views. For more information, see Section 8.7, “The Policy Set View,” on page 138.
<i>Run Configuration Wizard</i>	Guides you through creating a driver. After you fill out the wizard’s forms, Designer automatically generates policies that configure the driver to function as described in the forms.
<i>Edit Entitlements</i>	Enables you to select an entitlement that is associated with the driver and edit the entitlement’s settings. For more information, see Chapter 17, “Managing Entitlements,” on page 339.
<i>Password Synchronization</i>	Configures and displays the flow of password synchronization. For more information, see Section 12.6, “Integrating Passwords,” on page 245.
<i>Manage Application Schema</i>	Enables you to manage a copy of the connected system’s schema. You can make changes to a copy of the application schema so that you can test the Identity Manager drivers in Designer. See Section 11.7, “Managing a Copy of an Application Schema,” on page 216.
<i>Document Selection</i>	Launches the Document Generation Wizard, which documents the selected driver.
<i>Export to Configuration File</i>	Exports the driver to a .xml file. iManager can consume this format, and Designer can re-import it. For more information, see Section 19.10, “Exporting To a File,” on page 390.
<i>Live > Import</i>	Enables you to connect to a server, browse to and select a driver, and import the objects into the driver.
<i>Live > Deploy</i>	Prepares a deployment summary and then deploys selected objects and attributes.
<i>Live > Compare</i>	Compares selected drivers. Enables you to reconcile or update drivers. See Section 19.7, “Using the Compare Feature When Deploying,” on page 383.
<i>Live > Driver Configuration > Import Attributes</i>	Imports attributes from an existing driver.
<i>Live > Driver Configuration > Deploy Attributes</i>	Deploys the modified or imported attributes.
<i>Live > Driver Configuration > Refresh Application Schema</i>	.Specifies the server on an eDirectory tree where the schema is refreshed after an application’s schema changes. See Section 11.7.2, “Refreshing the Application Schema,” on page 219.
<i>Live > Driver Status</i>	Reports whether the driver is stopped or running.

Operation	Description
<i>Live > Start Drivers</i>	Starts the driver.
<i>Live > Stop Drivers</i>	Stops the driver.
<i>Live > Restart Drivers</i>	Restarts the drivers.
<i>Set Up Driver Security</i>	Launches the Driver Security Equals/Exclusions dialog box. Enables you to configure the selected driver's security equivalences and to exclude selected users from administrative roles.
Delete	Deletes the selected driver and its policies.
<i>Properties</i>	Launches the driver's property pages. Enables you to configure the driver.

E.5 Application Operations

The following figure illustrates Modeler operations that are available when you right-click an application.

Figure E-5 *Application Operations*

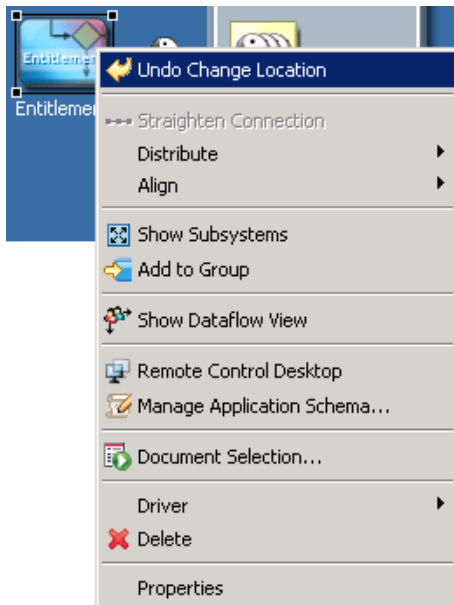


Table E-5 *Application Operations*

Operation	Description
<i>Undo Change Location</i>	Returns an item to its previous status.
<i>Disconnect eDir-to-eDir</i>	(Viewable when you select an eDir-to-eDir application) Separates the eDir-to-eDir application into two eDirectory drivers.

Operation	Description
<i>Straighten Connection</i>	Straightens a driver connection line. If a line is not within a few degrees of being horizontal or vertical, this option is dimmed.
<i>Distribute</i>	Evenly distributes applications vertically or horizontally. Press Ctrl, select the items that you want to distribute, then select a pattern.
<i>Align</i>	Aligns the selected objects horizontally and/or vertically. For more information, see Table E-6 on page 565 .
<i>Change to eDirectory Tree</i>	(Viewable when you select an eDirectory application.) Runs the Driver Configuration Wizard to install an eDir-to-eDir driver. Places a tree icon in the Identity Vault.
<i>Change to Identity Vault/Meta-Directory</i>	.(Viewable when you select an eDirectory application.) Runs the Driver Configuration Wizard to install an eDir-to-eDir driver. Places a vault con in the Identity Vault.
<i>Show/Hide Subsystems</i>	Lets you model an application's or operating system's subsystems. For example, if you have a Linux system, you can open it and drop MySQL inside as a subapplication that runs on Linux. This is for diagramming purposes only, but can be convenient for accurately capturing the structure of the enterprise systems around which you are building the identity solution.
<i>Add to Group</i>	Creates a Domain Group, and adds the selected items to it. The selected items are removed from any group that they were previously associated with.
<i>Show Dataflow View</i>	Displays the flow of information between the application and the driver in the Developer view. Launches the Dataflow view and lists Dataflow on the menu. For more information, see Chapter 12, "Managing the Flow of Data," on page 221 .
<i>Dataflow</i>	(Viewable when the Dataflow view is active.) Displays the flow of information between the application and the driver in the Developer view. Launches the Dataflow view. For more information, see Chapter 12, "Managing the Flow of Data," on page 221 and Table E-8 on page 565 .
<i>Secure Connection Settings</i>	.Runs the TLS Configuration Wizard so that you can enable SSL/TLS and create certificates.
<i>Remote Control Desktop</i>	Launches a remote control session for the selected application. The host server must have an existing VNC server running.
<i>Manage Application Schema</i>	Enables you to manage a copy of the connected system's schema. You can make changes to a copy of the application schema so that you can test the Identity Manager drivers in Designer. See Section 11.7, "Managing a Copy of an Application Schema," on page 216 .
<i>Document Selection</i>	Launches the Document Generation Wizard, which documents the application.
<i>Driver > Simulate</i>	.Runs the Simulate Policy Transformation program against the selected driver.

Operation	Description
<i>Driver > Show Policy Sets</i>	Launches the Policy Set and Policy Flow views. For more information, see Section 8.7, “The Policy Set View,” on page 138.
<i>Driver > Run Configuration Wizard</i>	Guides you through creating a driver. After you fill out the wizard’s forms, Designer automatically generates policies that configure the driver to function as described in the forms.
<i>Driver > Password Synchronization</i>	Configures and displays the flow of password synchronization. For more information, see Section 12.6, “Integrating Passwords,” on page 245.
<i>Driver > Document Selection</i>	Launches the Document Generation Wizard, which documents the driver.
<i>Driver > Export to Configuration File</i>	Exports the driver to a .xml file. iManager can consume this format, and Designer can re-import it. For more information, see Section 19.10, “Exporting To a File,” on page 390.
<i>Driver > Import</i>	Enables you to connect to a server, browse to and select a driver, and import the objects into the driver.
<i>Driver > Deploy</i>	Prepares a deployment summary and then deploys selected objects and attributes.
<i>Driver > Driver Configuration > Import Attributes</i>	Imports attributes from an existing driver.
<i>Driver > Driver Configuration > Deploy Attributes</i>	.Deploys the modified or imported attributes.
<i>Driver > Driver Status</i>	Reports whether the driver is stopped or running.
<i>Driver > Start Driver</i>	Starts the driver.
<i>Driver > Stop Driver</i>	Stops the driver.
<i>Driver > Restart Driver</i>	Restarts the drivers.
<i>Driver > Set Up Driver Security</i>	Launches the Driver Security Equals/Exclusions dialog box. Enables you to configure the selected driver’s security equivalences and to exclude selected users from administrative roles.
<i>Driver > Properties</i>	Launches the driver’s property pages. Enables you to configure the driver.
<i>Delete</i>	Deletes the application and driver.
<i>Properties</i>	Enables you to configure Identity Vaults, driver sets, drivers, and applications.

E.6 Submenus

Table E-6 *Align Submenu*

Operation	Description
<i>Align Top</i>	Aligns the top edge of the selected objects.
<i>Align Bottom</i>	Aligns the bottom edge of the selected objects.
<i>Align Left</i>	Aligns the left edge of the selected objects.
<i>Align Right</i>	Aligns the right edge of the selected objects.
<i>Align Center</i>	Horizontally aligns the centers of the selected objects.
<i>Align Middle</i>	Vertically aligns the middles of the selected objects.

Table E-7 *Arrange Applications Submenu*

Operation	Description
<i>Arrangement Off</i>	Disables a previously selected auto-arrangement method.
<i>Box</i>	Arranges application icons in a square around the driver set.
<i>Circle</i>	Arranges application icons in a circle around the driver set.
<i>Half Circle</i>	Arranges application icons in a semicircle around the driver set.
<i>Star</i>	Arranges application icons in a star around the driver set.
<i>Fan Out - Bottom</i>	Arranges application icons in a fan shape below the driver set.
<i>Fan Out - Left</i>	Arranges application icons in a fan shape to the left of the driver set.
<i>Fan Out - Right</i>	Arranges application icons in a fan shape to the right of the driver set.
<i>Fan Out - Top</i>	Arranges application icons in a fan shape above the driver set.
<i>Expand/Contract</i>	Expands or contracts the layout of the application icons. Selecting this option opens a dialog box from which you drag the slide in the Factor field to change the layout.

Table E-8 *Dataflow Submenu*

Operation	Description
<i>Publish</i>	Specifies that the Publisher channel is synchronized for the selected objects (uni-directional sync from selected objects.) For more information, see Chapter 12, "Managing the Flow of Data," on page 221.

Operation	Description
<i>Subscribe</i>	Specifies that the Subscriber channel is synchronized on the selected objects (uni-directional sync to selected objects.) For more information, see Chapter 12, "Managing the Flow of Data," on page 221.
<i>Sync</i>	Specifies that both the Publisher and Subscriber channel are synchronized for the selected objects (bidirectional sync.) For more information, see Chapter 12, "Managing the Flow of Data," on page 221.
<i>Ignore from</i>	Specifies that the selected objects ignore Subscriber channel synchronization. For more information, see Chapter 12, "Managing the Flow of Data," on page 221.
<i>Ignore to</i>	Specifies that the selected objects ignore both Publisher and Subscriber channel synchronization. For more information, see Chapter 12, "Managing the Flow of Data," on page 221.
<i>Ignore both directions</i>	Specifies that the selected objects ignore Publisher channel synchronization. For more information, see Chapter 12, "Managing the Flow of Data," on page 221.

Table E-9 *Distribute Operations Submenu*

Operation	Description
<i>Horizontal</i>	Evenly spaces the selected objects horizontally.
<i>Vertical</i>	Evenly spaces the selected objects vertically.

E.7 Keyboard Support

The following table describes common keyboard shortcuts available in the Modeler.

Table E-10 *Shortcut Keys*











Keystroke	Description
/	Navigates to the item's next connection.
\	Navigates to the item's previous connection.
Delete	Deletes the selected item or line.
Left-arrow	Navigates left.
Right-arrow	Navigates right.
Up-arrow	Navigates up.
Down-arrow	Navigates down.
<Alt>+Down-arrow	Navigates into a subgroup.
<Alt>+Up-arrow	Navigates out of a subgroup.






Keystroke	Description
<Ctrl> + =	Zooms in.
<Ctrl> + -	Zooms out.
<Ctrl> + A	Selects all objects in the current project.
<Ctrl> + C	Copies the selected objects to the Clipboard.
<Ctrl> + F	Opens the Find dialog box for searching the project.
<Ctrl> + V	Pastes the Clipboard contents to the selected location.

Key Terms

F

Table F-1 *Key Terms Used in Designer*

Icon	Term	Description
	Application Object	Represents a connected system, whether that system is a database or CSV files.
	Driver Object	A collection of channels, policies, rules, and filters that connect an application to an eDirectory™ tree running Identity Manager. Each driver performs different tasks. Policies, rules, and filters tell the driver how to manipulate the data to perform those tasks.
	Driver Set	A collection of Identity Manager drivers that are stored in a single location.
	Driver's Properties Shortcut	Double-clicking the shortcut brings up the Properties window associated with the object's name.
	Filter	A gatekeeper that determines which objects (groups, users, and the attributes associated with them) can be sent to and from eDirectory. Allows the objects to be sent.
	Identity Vault	Represents the eDirectory tree that has Identity Manager installed on it.
	Notification Collection	A placeholder for all e-mail notification templates.
	Notification Object	A notification template. Used to send an e-mail notification to a user concerning a password event (for example, a request for a password hint).
	Policy	A collection of rules and arguments that enable you to configure an application to send or receive events to or from an Identity Vault. You use policies to manipulate the data that you receive from the Identity Vault or from the application. Each driver performs different tasks. Policies tell the driver how to manipulate the data to perform those tasks.
	Project Name	The name of the project when you create it. The name of the project stored in the <code>\workspace\project name</code> directory.

Icon	Term	Description
	Publisher Channel	A combination of rules and policies. Designer enables you to import a channel instead of the entire driver. The Subscriber and Publisher channels describe the direction in which the information flows. The Publisher channel takes the event from the application, database, or CSV file and sends that event to eDirectory. The Subscriber and Publisher channels act independently; actions in one are not affected by what happens in the other.
	Schema Mapping	Enables Identity Manager to synchronize information to the connected system without changing the connected system. The schema translates all classes and attributes for the connected system.
	Server Object	The server in the tree where you have Identity Manager installed.
	Subscriber Channel	A combination of rules and policies. Designer enables you to import a channel instead of the entire driver. The Subscriber channel takes the event from eDirectory and sends that event to whatever the receiving system is (application, database, or CSV file). The Subscriber and Publisher channels act independently; actions in one are not affected by what happens in the other.
	XSLT	Represents XSLT stylesheets. XSLT is used instead of the DirXML Script.