

# Novell ZENworks® Web Self-Service

[www.novell.com](http://www.novell.com)

---

INSTALLATION AND ADMINISTRATION

April 14, 2003



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.

You may not export or re-export this product in violation of any applicable laws or regulations including, without limitation, U.S. export regulations or the laws of the country in which you reside.

Copyright © 2000 -2003 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.

U.S. Patent Pending.

Novell, Inc.  
1800 South Novell Place  
Provo, UT 84606  
U.S.A.

[www.novell.com](http://www.novell.com)

Installation and Administration

April 14, 2003

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

## **Novell Trademarks**

ApprovalFlow is a trademark of Novell, Inc.

ConsoleOne is a registered trademark of Novell, Inc. in the United States and other countries.

DeFrame is a trademark of Novell, Inc.

DirXML is a registered trademark of Novell, Inc. in the United States and other countries.

eDirectory is a trademark of Novell, Inc.

exteNd is a trademark of Novell, Inc.

exteNd Director is a trademark of Novell, Inc.

iChain is a registered trademark of Novell, Inc. in the United States and other countries.

NDS is a registered trademark of Novell, Inc. in the United States and other countries.

NetWare is a registered trademark of Novell, Inc. in the United States and other countries.

NLM is a trademark of Novell, Inc.

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

Novell Application Launcher is a trademark of Novell, Inc.

Novell Client is a trademark of Novell, Inc.

Novell Consulting is a service mark of Novell, Inc.

Novell iFolder s is a registered trademark of Novell, Inc. in the United States and other countries.

ZENworks is a registered trademark of Novell, Inc. in the United States and other countries.

ZENworks OnDemand Services is a trademark of Novell, Inc.

## **Third-Party Trademarks**

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



# Contents

<b>About This Guide</b>	<b>9</b>
<b>Part I    Overview</b>	
<b>1    What Is ZENworks Web Self-Service?</b>	<b>13</b>
<b>2    Web Self-Service Software Components</b>	<b>15</b>
OnDemand Services Portal . . . . .	15
UsageServer and UsageClient . . . . .	16
Maintenance Process. . . . .	16
<b>Part II   Installation</b>	
<b>3    Software Requirements and Sample Configurations</b>	<b>19</b>
Software Requirements. . . . .	19
Web Server . . . . .	19
Network Server . . . . .	20
Novell eDirectory and ConsoleOne . . . . .	20
Desktop Operating Systems and Web Browsers . . . . .	21
Sample Configurations . . . . .	22
Sample Configuration 1 - Single-Server Environment . . . . .	23
Sample Configuration 2 - Two-Server Environment . . . . .	24
Sample Configuration 3 - Three-Server Environment . . . . .	25
Sample Configuration 4 - Two-Server Environment with an exteNd Director Portal . . . . .	26
<b>4    Preparing for Installation</b>	<b>27</b>
Setting Up the Web Server Environment . . . . .	27
Increasing the Tomcat Memory Allocation . . . . .	27
NetWare 6 Server. . . . .	27
Windows 2000 Server. . . . .	28
Synchronizing Server Dates and Times . . . . .	28
<b>5    Installing Web Self-Service</b>	<b>29</b>
Installing the Portal Software . . . . .	29
Installation Worksheet . . . . .	29
Installing the OnDemand Services Portal . . . . .	31
Installing the OnDemand Services Gadgets to an Existing Novell exteNd Director 4.1 Portal . . . . .	32
Assigning Rights to the OnDemand Admin User . . . . .	33
Setting Up ZENworks Application Association . . . . .	34
Setting Up Cost Centers . . . . .	35
Creating a Cost Center . . . . .	35
Adding Budget Holders . . . . .	36
Assigning Users. . . . .	37
Configuring the Catalog Root and ApprovalFlow E-Mail Settings . . . . .	39
Assigning Portal Access . . . . .	40
Assigning Pages to Users. . . . .	41
Assigning Pages to Budget Holders . . . . .	43

Enabling Usage Tracking of ZfD DeFrame Thin-Client Applications . . . . .	46
Starting the Maintenance and UsageServer Processes. . . . .	46
Starting the Maintenance Process. . . . .	46
Starting the UsageServer Process. . . . .	47
What's Next . . . . .	48
<b>Part III Upgrade</b>	
<b>6 What's New in ZENworks 6 Web Self-Service</b>	<b>51</b>
<b>7 Software Requirements</b>	<b>53</b>
Web Server Requirements . . . . .	53
Network Server Requirements . . . . .	53
Novell eDirectory and ConsoleOne Requirements . . . . .	54
eDirectory and LDAP Naming Restrictions . . . . .	54
ConsoleOne . . . . .	55
Desktop Operating System and Web Browser Requirements. . . . .	55
<b>8 Upgrading to Web Self-Service</b>	<b>57</b>
Upgrading the Portal Software . . . . .	57
Upgrading an NPS 1.5 Runtime Portal . . . . .	57
Upgrading an NPS 1.5 Full Version Portal. . . . .	59
Recustomizing Gadget Style Sheets. . . . .	60
Assigning Portal Access . . . . .	60
Upgrading the Application Association Software . . . . .	65
Restarting the Portal Server . . . . .	66
Upgrading the Usage Software (Optional). . . . .	66
Upgrading the UsageServer Software . . . . .	66
Upgrading the UsageClient Software . . . . .	68
Upgrading the Maintenance Software (Optional) . . . . .	69
<b>Part IV Administration</b>	
<b>9 Administration Basics</b>	<b>73</b>
OnDemand Services Gadgets. . . . .	73
Portal Administration . . . . .	73
Web Self-Service Snap-Ins for ConsoleOne . . . . .	73
Novell eDirectory Management . . . . .	73
<b>10 Creating Application Packages</b>	<b>75</b>
Preparing the Package's Applications. . . . .	75
Desktop Applications . . . . .	75
Thin-Client Applications . . . . .	76
Web Applications . . . . .	76
Creating the Package Object . . . . .	78
Adding Applications to the Package. . . . .	79
Establishing the Package's Cost . . . . .	80
<b>11 Tracking Application Usage</b>	<b>83</b>
Usage Tracking Support for Each Application Type. . . . .	83
Enabling Usage Tracking of an Application . . . . .	84
Ensuring Accurate Usage Tracking for Thin-Client Applications . . . . .	84
Starting the UsageServer and Maintenance Processes. . . . .	85
Starting the UsageServer Process. . . . .	85
Starting the Maintenance Process. . . . .	86

<b>12</b>	<b>Limiting Package Availability</b>	<b>87</b>
	Configuring the Package Request Gadget to Limit Package Availability . . . . .	87
	Giving Users Trustee Rights to Packages . . . . .	88
	Making Users Trustees of a Package Object . . . . .	88
	Making Users Trustees of a Container Object . . . . .	88
<b>13</b>	<b>Organizing Packages and Applications</b>	<b>91</b>
	Organizing Packages in the Package Request Gadget . . . . .	91
	Organizing Applications in the Launch Item Gadget . . . . .	92
<b>14</b>	<b>Securing Web Applications With Novell iChain</b>	<b>93</b>
	Setting Up Protected Resources . . . . .	93
	Creating ACL Rules . . . . .	94
	Configuring iChain Server ACL List Refresh . . . . .	95
	Making the OnDemand Admin User Object a Trustee of ACL Rules. . . . .	96
	Associating ACL Rules with Web Applications . . . . .	97
<b>15</b>	<b>Customizing Notification Messages</b>	<b>99</b>
	Modifying the Message Subject . . . . .	99
	Modifying the Message Body . . . . .	99
<b>16</b>	<b>Creating Cost Centers</b>	<b>101</b>
	Creating a Cost Center . . . . .	101
	Adding Budget Holders . . . . .	102
	Assigning Users to the Cost Center . . . . .	103
	Using ConsoleOne to Individually Assign Users . . . . .	103
	Using the OnDemand User Configuration Utility to Assign Multiple Users . . . . .	104
<b>17</b>	<b>Billing for Purchases</b>	<b>107</b>
	Developing Your Own Interface . . . . .	107
	Getting Development Help . . . . .	107
<b>18</b>	<b>Changing the Status of a Purchase</b>	<b>109</b>
<b>19</b>	<b>Managing User Workstation Software</b>	<b>111</b>
	Installing the Workstation Software. . . . .	111
	Using the OnDemand Services Launch Item Gadget to Install the Workstation Software . . . . .	111
	Using Novell Application Launcher to Install the Workstation Software. . . . .	112
	Manually Installing the Workstation Software. . . . .	113
	Updating the ICA Web Client . . . . .	113
<b>20</b>	<b>Managing the Maintenance and UsageServer Processes</b>	<b>115</b>
	Understanding the Maintenance Process . . . . .	115
	Understanding the UsageServer Process . . . . .	115
	How the UsageServer and UsageClient Work . . . . .	116
	What to Do If the UsageServer and the UsageClient Are in Different Subnets . . . . .	116
	Multiple UsageServers and Load Balancing . . . . .	118
	Explicitly Defining the UsageServer's Address . . . . .	118
	Adding the UsageServer's Address to the Launch Item Gadget or Novell Application Launcher. . . . .	119
	Adding the UsageServer's Address to a DeFrame Server's Registry. . . . .	120
	Installing Additional Maintenance and UsageServer Processes . . . . .	120
	Modifying the Maintenance and UsageServer Configuration File (Commerceserver.cfg) . . . . .	121
	Modifying Logging Settings . . . . .	123

## **Part V   Appendixes**

<b>A</b>	<b>Gadget Configuration Settings</b>	<b>127</b>
	Accessing Gadget Configuration Settings . . . . .	127
	Configuring Gadget Assignment Settings . . . . .	127
	Configuring Gadget Object Settings . . . . .	128
	Approval Gadget (OD_ApprovalGadget) . . . . .	128
	Configuration Wizard Gadget (OD_ConfigWizardGadget) . . . . .	128
	Current Budget Holder Gadget (OD_CurrentBHGadget) . . . . .	129
	Edit Account Gadget (OD_EditAccountGadget). . . . .	129
	Launch Item Gadget (OD_LaunchItemGadget) . . . . .	129
	Package Request Gadget (OD_PackageRequestGadget) . . . . .	130
	Report Gadget (OD_ReportGadget). . . . .	130
	Tabbed Container Gadget (OD_TabbedContainerGadget) . . . . .	131
	User Administration Gadget (OD_UserAdminGadget) . . . . .	131
	Workflow Tracking Gadget (OD_WorkflowTrackingGadget) . . . . .	132
<b>B</b>	<b>OnDemand Admin User Rights</b>	<b>133</b>
<b>C</b>	<b>Maintaining User Access to the Portal During Upgrade</b>	<b>135</b>
	Maintaining Access to a Windows Portal Server . . . . .	135
	Maintaining Access to a NetWare Portal Server . . . . .	135
	Single-Server Upgrade Procedures . . . . .	136



# About This Guide

This guide is designed to help you set up Novell® ZENworks® Web Self-Service as quickly and easily as possible. The guide is intended for network administrators who will install and administer the Web Self-Service components and is organized as follows:

- ♦ “Overview” on page 11
- ♦ “Installation” on page 17
- ♦ “Upgrade” on page 49
- ♦ “Administration” on page 71

## ZENworks Web Self-Service Software

Depending on how you obtained your software, the location from which you start installation procedures will vary. This guide assumes that you have a CD from which to install the software.

If you have downloaded the software from the [Novell Product Downloads page \(http://download.novell.com\)](http://download.novell.com), the Web Self-Service software is at the root of the \ZENworks6\_Web\_Self-Service directory where you extracted the .zip file. When completing the installation procedures, you will need to substitute the \ZENworks6\_Web\_SelfService directory for any references to the *ZENworks 6 Web Self-Service Program* CD.

## Documentation Updates

For the most recent version of the *ZENworks Web Self-Service Installation and Administration* guide, visit the [Novell ZENworks 6 Web site \(http://www.novell.com/documentation/lg/zenworks6\)](http://www.novell.com/documentation/lg/zenworks6).

## 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.

## User Comments

We want to hear your comments and suggestions about this manual and the other documentation included with ZENworks Web Self-Service. To contact us, send e-mail to [proddoc@novell.com](mailto:proddoc@novell.com).





## Overview

The following sections introduce Novell® ZENworks® Web Self-Service and provide an overview of the software components. For installation instructions, see [“Installation” on page 17](#). For instructions about upgrading from ZENworks OnDemand Services™ 2, see [“Upgrade” on page 49](#).

[Chapter 1, “What Is ZENworks Web Self-Service?,” on page 13](#)

[Chapter 2, “Web Self-Service Software Components,” on page 15](#)



# 1

## What Is ZENworks Web Self-Service?

Novell® ZENworks® Web Self-Service, working in conjunction with ZENworks 6 Desktop Management or ZENworks for Desktops (ZfD) 4.0.1, provides distribution of applications to users through a Web browser interface.

### Administrator Provisioning

The Application Management component of ZfD enables administrators to provision, or push, applications to users. By defining applications in Novell eDirectory™ and assigning them to users, you make the applications available on the users' workstations through Novell Application Launcher™.

ZENworks Web Self-Service extends administrator provisioning to individual organizations. You still define applications in eDirectory. However, by setting up organizations as cost centers with budget holders, you enable each organization, or cost center, to provision the eDirectory applications to users within the organization. The provisioning is done through a Web browser interface rather than ConsoleOne®, the standard eDirectory management utility.

Users can launch administrator-provisioned applications from Novell Application Launcher or the Web Self-Service browser interface.

### User Self-Service

In addition to extending the administrator provisioning capabilities of ZfD, Web Self-Service provides users with the ability to request applications. A Web browser interface presents users with a catalog of available applications. When a user requests one of the applications, the request goes through an automated process during which the request can be approved or denied by the budget holder for the user's cost center. If the request is approved, the user gains access to the application and can launch it through the Web browser interface or ZfD Novell Application Launcher.

### Web Browser Interface

Web Self-Service uses a Web browser interface through which users can request applications, track their requests, and launch applications. Budget holders, in addition to having the same capabilities as users, can approve or deny requests for applications.

### Packages

Web Self-Service requires applications to be grouped together into packages. A package can contain a single application or a suite of applications. When a user receives access to either an administrator-provisioned or a user-requested package, he or she receives access to all applications in the package.

A package can include any of the application types (desktop, thin-client, or Web) supported by ZfD.

## **Usage Tracking**

Web Self-Service enables you to track how much or how often a user runs an application. The application must belong to a package for usage tracking to work. Usage tracking will not work on eDirectory applications (Application objects) associated to users outside of a package.

## **Package Fees and Cost Centers**

You can provide applications free of charge, or you can charge purchase and usage fees. Fees are associated with the package, not individual applications, and apply when the package is purchased (purchase fee) or any of the package's applications are run (usage fee).

If you assign fees to a package, a user's charges are tracked against his or her cost center, regardless of whether the package was administrator provisioned or user requested.

## **Novell iChain Integration**

Novell iChain<sup>®</sup> provides secure authentication and access to portals, Web-based content, and Web applications. Web Self-Service integrates with Novell iChain 2.x to enable you to secure Web applications included in packages. Through the use of iChain ACL (Access Control List) rules, Web Self-Service restricts access to only those users who have access to the Web application's package.

# 2

## Web Self-Service Software Components

The following sections provide an overview of the Web Self-Service software components:

- ♦ “OnDemand Services Portal” on page 15
- ♦ “UsageServer and UsageClient” on page 16
- ♦ “Maintenance Process” on page 16

### OnDemand Services Portal

The OnDemand Services portal is the Web interface for self-service application management. The portal is powered by Novell® exteNd Director™ 4.1, a Java\* servlet that runs on a Web server. A group of basic Novell exteNd Director administration gadgets and ZENworks® OnDemand Services™ gadgets provides the functionality required to administer the portal and use it for self-service application management.

The OnDemand Services portal enables you to customize the functionality that is available to a user. For example, you could:

- ♦ Assign all users the functionality required to request applications, track application requests, and launch applications.
- ♦ Assign budget holders (those users responsible for approving requests from users associated with their cost center) the functionality required to approve or deny requests; add, change, or remove budget holders; and generate reports for their cost centers.
- ♦ Assign administrators the functionality required to manage the OnDemand Services portal.

A user can receive multiple assignments. For example, a budget holder can receive the budget holder assignment as well as a user assignment. His or her portal view would then include both the user and budget holder functionality.

The OnDemand Services portal is provided on the *ZENworks 6 Web Self-Service Program CD*.

#### Novell exteNd Director Portal

The OnDemand Services portal is limited to the functionality required to administer the basic portal and use it for self-service application management.

If you require a more extensive portal, you can use Novell exteNd Director 4.1 Standard Edition. This is a fully configurable portal where you can use the OnDemand Services gadgets and other exteNd Director gadgets that provide additional functionality and information. For information about purchasing Novell exteNd Director 4.1, contact your nearest Novell office. To locate the nearest Novell office, see the [Novell Worldwide Office Web site \(http://www.novell.com/offices\)](http://www.novell.com/offices).

# UsageServer and UsageClient

Web Self-Service lets you track how much (number of minutes) and how often (number of times) a user runs an application. You can use this for information purposes only, or you can base an application's cost on its usage (for example, two dollars per use or five cents per minute of use).

Usage tracking is enabled through a Java process, referred to as the UsageServer process, and a UsageClient. The UsageServer process runs on a network server while the UsageClient runs on user workstations. Working together, the UsageServer process and the UsageClient record usage information for applications in Novell eDirectory™.

The UsageServer process is installed with the ZfD Server software. You can run the UsageServer process on the server, or you can copy the files to another server and run it from there.

The UsageClient must be run on the same machine as the application being tracked:

- ♦ For desktop applications, the UsageClient must be on the user's workstation.
- ♦ For ZfD DeFrame™ thin-client applications, the UsageClient must be on the DeFrame terminal server.

Web Self-Service includes an installation program that you can use to install the UsageClient on ZfD DeFrame terminal servers. The installation program installs the UsageClient along with the ZfD DeFrame client, RDP Web client, ICA Web client, and ZfD Novell Application Launcher plug-in. The UsageClient is installed automatically on user workstations the first time they launch an application that requires either the UsageClient, ZfD DeFrame client, RDP client, ICA client, or ZfD Novell Application Launcher plug-in. If necessary, you can use the installation program on user workstations.

For more information about the UsageServer process and the UsageClient, see [“Understanding the UsageServer Process” on page 115](#).

## Maintenance Process

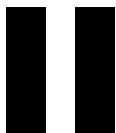
When a user receives access to an application package, a Purchase object is created in Novell eDirectory. If the user's access to the application package is usage-based, a Usage object is created under the Purchase object each time the user launches one of the package's applications. Each Usage object records a single instance, or use, of one of the package's applications. This results in multiple Usage objects for each Purchase object.

Web Self-Service uses a Java process, referred to as the Maintenance process, to clean up Purchase and Usage objects in eDirectory and close out usage tracking of applications if the UsageServer process becomes unavailable. The Maintenance process must be running at all times.

The Maintenance process is installed with the ZfD Server software. You can run the Maintenance process on the server, or you can copy the files to another server and run it from there.

For more information about the Maintenance process, see [“Understanding the Maintenance Process” on page 115](#).





# Installation

The following sections explain how to create a new Novell® ZENworks® Web Self-Service installation. If you are upgrading from ZENworks OnDemand Services™ 2, see [“Upgrade” on page 49](#).

[Chapter 3, “Software Requirements and Sample Configurations,” on page 19](#)

[Chapter 4, “Preparing for Installation,” on page 27](#)

[Chapter 5, “Installing Web Self-Service,” on page 29](#)



# 3

## Software Requirements and Sample Configurations

Before you begin installing Novell® ZENworks® Web Self-Service, you should review the software requirements and sample configurations in the following sections:

- ♦ [“Software Requirements” on page 19](#)
- ♦ [“Sample Configurations” on page 22](#)

### Software Requirements

The following sections list the requirements for Web Self-Service:

- ♦ [“Web Server” on page 19](#)
- ♦ [“Network Server” on page 20](#)
- ♦ [“Novell eDirectory and ConsoleOne” on page 20](#)
- ♦ [“Desktop Operating Systems and Web Browsers” on page 21](#)

### Web Server

The OnDemand Services portal requires one of the Web server environments listed in the table below.

Operating System	Web Server	Web Application Server	JVM
NetWare® 6 with Support Pack 3 or later	Apache 1.3.27	Tomcat 4.1.18	Novell JVM* for NetWare 1.4
Windows* 2000 with Support Pack 2 or later	IIS 5.0 or Apache 2.0.44	Tomcat 4.1.18	Sun* 1.4.1_01 or IBM* 1.3

**IMPORTANT:** On a Windows 2000 server, you must run Tomcat as a standard Windows application in the user space rather than as a Windows service in the system space.

If you have an existing Novell exteNd Director™ 4.1 Standard Edition portal that you will use instead of the OnDemand Services portal, you can install the OnDemand Services gadgets to that portal. The exteNd Director 4.1 portal must reside in one of the Web server environments listed above.

For an overview of the OnDemand Services portal, see [“OnDemand Services Portal” on page 15](#).

## Network Server

Web Self-Service includes two Java processes, Maintenance and UsageServer. The Maintenance process cleans up purchase and usage records in Novell eDirectory™ and closes out usage tracking of applications when the UsageServer becomes unavailable. The Maintenance process must be running at all times. The UsageServer process enables application usage to be tracked. It needs to be running if you want to be able to track the number of times or minutes a user uses an application.

By default, the Maintenance and UsageServer processes are installed to the ZfD server by the ZfD Server installation program. If you don't want to run the processes on the server, you can copy them to any network server that meets the following requirements:

Operating System	JVM
NetWare 5.1 or 6	Novell JVM for NetWare 1.3 or later
Windows NT*/2000	Sun 1.3 or later IBM 1.3 or later

For more information about the UsageServer process, see [“UsageServer and UsageClient” on page 16](#).

For more information about the Maintenance process, see [“Maintenance Process” on page 16](#).

## Novell eDirectory and ConsoleOne

Web Self-Service requires Novell eDirectory 8.6.2 or later. As you prepare your environment for Web Self-Service, please be aware of the following:

- ♦ If the portal server is a NetWare server, the server must contain Read/Write replicas of the partitions where the OnDemand Services objects (OnDemand Admin User object, OnDemandService object, and package objects), Application objects, and User objects reside.
- ♦ If the portal server is a Windows server, eDirectory does not need to be installed on the server. However, the OnDemand Services gadgets must have LDAP access to an eDirectory server that contains Read/Write replicas of the partitions where the OnDemand Services objects (OnDemand Admin User object, OnDemandService object, and package objects), Application objects, and User objects reside.
- ♦ Because the OnDemand Services objects use auxiliary classes that are only supported in NDS® 8.x or higher, we recommend that you install Web Self-Service in a tree that is exclusively NDS 8.x or later. If you want to use the OnDemand User Configuration utility to configure ApprovalFlow™ settings (cost center, account ID, and so forth) for multiple users at one time, the User objects must be located on eDirectory 8.5 (or later) replicas.

If you require NDS 6/7 servers in the same tree as Web Self-Service, the NDS 6/7 servers should not hold replicas that contain OnDemand Services objects (OnDemand Admin User object, OnDemandService object, and package objects), Application objects, or User objects for Web Self-Service users.

**NOTE:** NDS 6/7 servers are capable of synchronizing the schema associated with auxiliary classes and can therefore coexist in the same tree. However, objects that have auxiliary classes on them are converted to "Unknown" objects on NDS 6/7 servers. These objects will be displayed as "Unknown" objects and this will prevent you from administering these objects when your primary connection is to a NDS 6/7 server. In addition, Web Self-Service users whose objects appear as "Unknown" will be unable to authenticate to the NDS 6/7 servers.

**eDirectory and LDAP Naming Restrictions**

The OnDemand Services gadgets, UsageServer process, and Maintenance process use LDAP to access eDirectory. In all eDirectory contexts that will be accessed by these components, you should make sure that object names (including container and leaf objects) do not include any of the following characters. These characters, when used in LDAP names (for example, cn=jsmith,ou=users,o=novell), will result in invalid syntax.

" (double quote)	+ (plus)
= (equals)	\ (backslash)
< (open angle bracket)	; (semicolon)
> (close angle bracket)	/ (forwardslash)
, (comma)	# (pound sign)

In general, you should ensure that your entire eDirectory tree is LDAP-compliant. To support Web Self-Service, make sure the following eDirectory contexts comply to LDAP naming conventions:

- ◆ User object contexts (for example, cn=jsmith,ou=users,o=novell)
- ◆ Application object contexts (for example, cn=app1,ou=apps,o=novell)
- ◆ OnDemand Services object contexts, including the OnDemand Admin User object, the OnDemandService object, package objects, and item objects (for example, cn=OnDemandService,ou=services,o=novell)
- ◆ Portal object contexts (for example, cn=PortalUser,ou=portal,o=novell)

**ConsoleOne**

Web Self-Service requires ConsoleOne® 1.3.5 or later. ConsoleOne 1.3.5 is included on the *ZENworks 6 Companion 1* CD.

**Desktop Operating Systems and Web Browsers**

The following table lists the desktop operating systems and Web browsers that Web Self-Service supports.

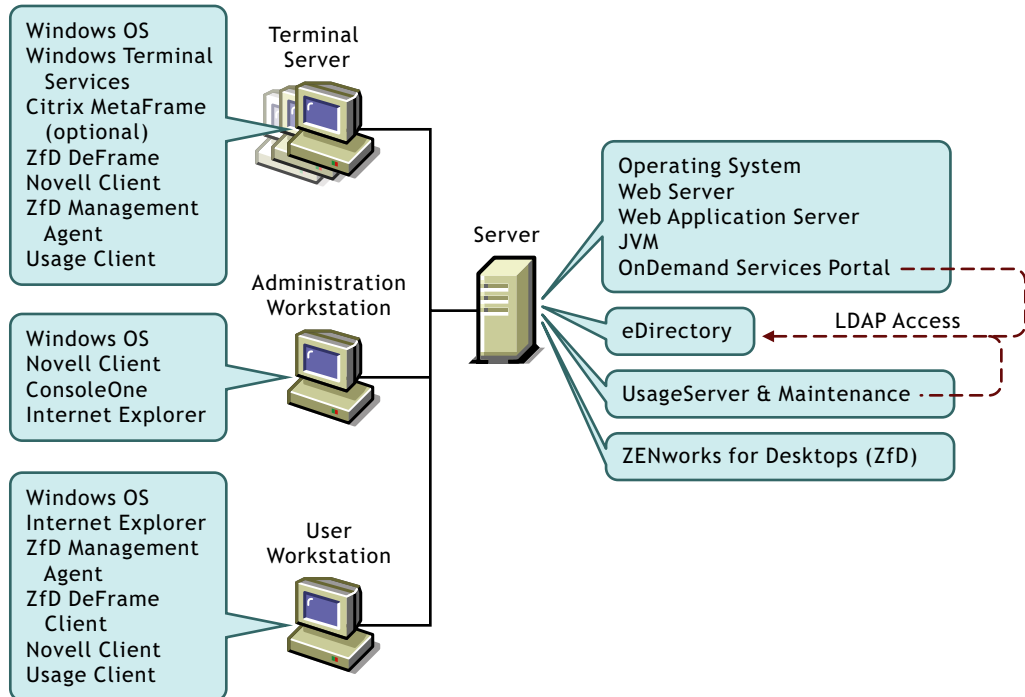
Operating System	Web Browsers
Windows 98 SE	Microsoft* Internet Explorer 5.5 with SP2
Windows NT 4.0 Workstation	Microsoft Internet Explorer 6.0
Windows 2000 Professional	
Windows XP Professional	

Netscape\* and other browsers are not supported.

You should make sure that the latest support pack is applied to each of these software components.

# Sample Configurations

The following diagram shows the software components required in a Web Self-Service environment.

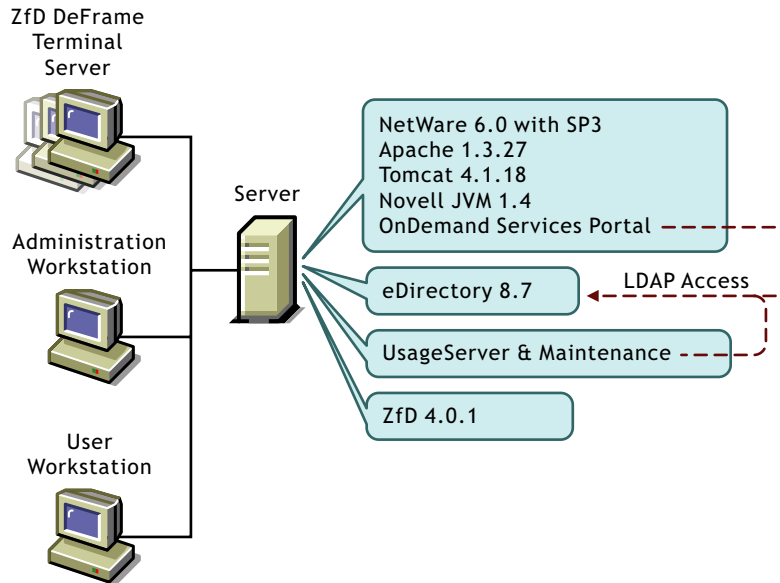


All software components included in a shaded box must be installed on the same machine, but software components in different boxes (on the server) do not necessarily need to be installed on the same server. The following sections provide examples of possible configurations:

- ♦ “Sample Configuration 1 - Single-Server Environment” on page 23
- ♦ “Sample Configuration 2 - Two-Server Environment” on page 24
- ♦ “Sample Configuration 3 - Three-Server Environment” on page 25
- ♦ “Sample Configuration 4 - Two-Server Environment with an exteNd Director Portal” on page 26

## Sample Configuration 1 - Single-Server Environment

In this configuration, all server software components are installed on a NetWare 6 server.



The NetWare 6 server includes the Web server environment (Apache, Tomcat, JVM, and the OnDemand Services portal), eDirectory, ZENworks for Desktops, UsageServer process, and Maintenance process.

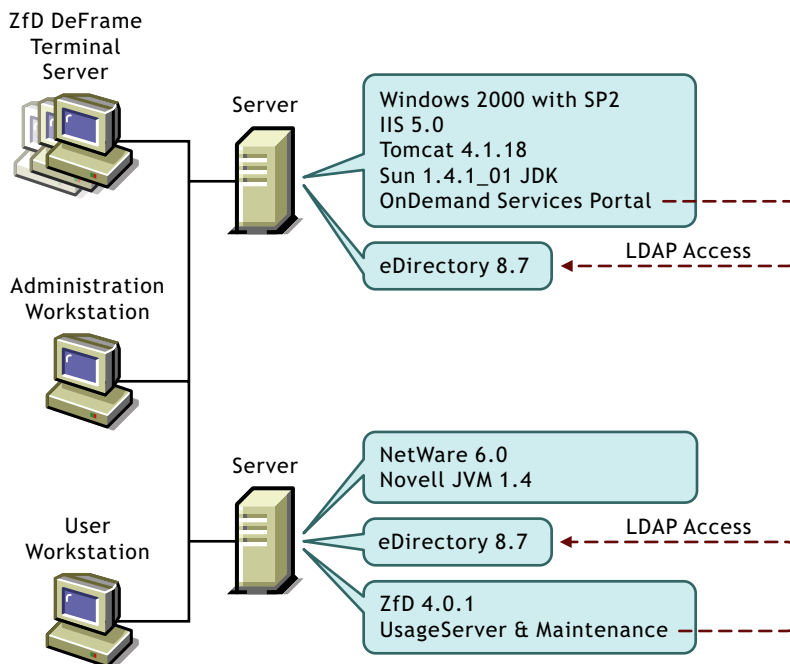
The terminal server is configured as a ZfD DeFrame™ terminal server to enable distribution of thin-client applications.

The user workstation includes all components required to use either the Web Self-Service portal for requesting and launching applications or the ZfD Novell Application Launcher™ for launching applications.

This configuration saves on hardware but might sacrifice performance. Users will be accessing the OnDemand Services portal through the Web server, while at the same time ZfD, the UsageServer process, and the Maintenance processes will be running. This configuration is recommended only if the server has a lot of processing power or is servicing a limited number of users.

## Sample Configuration 2 - Two-Server Environment

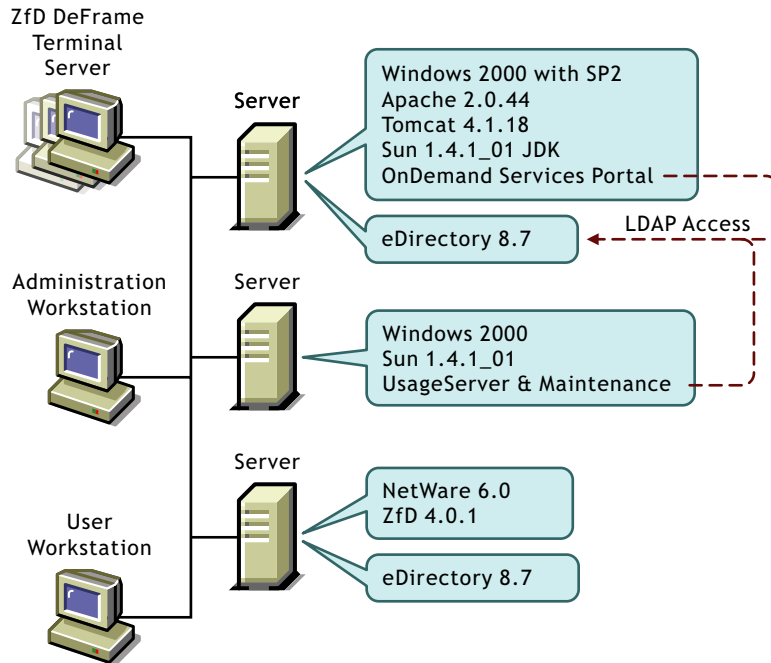
This sample configuration has Microsoft Internet Information Services being used as the Web server with Tomcat and Sun JDK\*. The ZfD server software and the UsageServer and Maintenance processes are installed on a NetWare 6 server. eDirectory 8.7 is installed on both servers, although it would not be required on the Web server as long as the portal could have LDAP access to eDirectory on the NetWare 6 server.





## Sample Configuration 3 - Three-Server Environment

In this configuration, three different servers are being used: a portal server, a UsageServer and Maintenance process server, and a ZfD server.



The portal server is a Windows 2000 server with Apache, Tomcat, Sun JDK, and the OnDemand Services portal. eDirectory is installed on the portal server.

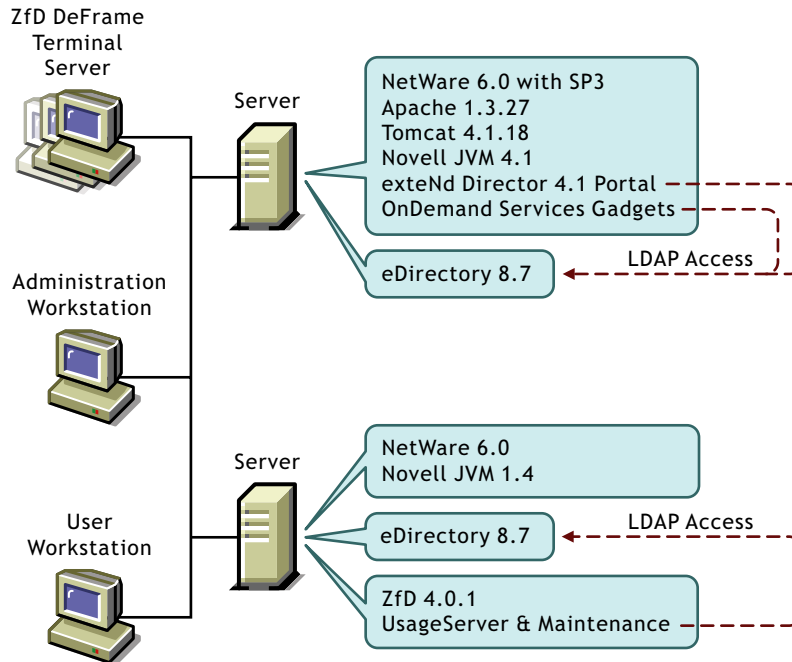
The UsageServer and Maintenance processes are installed on a second server. The processes have LDAP access to eDirectory on the portal server.

The ZfD server includes the ZfD server software and eDirectory (a requirement for the ZfD server software).

## Sample Configuration 4 - Two-Server Environment with an exteNd Director Portal

This sample configuration uses one server for the portal components and one server for the ZfD server software, UsageServer process, and Maintenance process.

The portal server is using Novell exteNd Director 4.1 Standard Edition for the portal rather than the OnDemand Services portal. The OnDemand Services gadgets have been installed to the portal.



# 4

## Preparing for Installation

Before beginning your Novell® ZENworks® Web Self-Service installation, you need to complete the preparation tasks in the following sections:

- ♦ “Setting Up the Web Server Environment” on page 27
- ♦ “Increasing the Tomcat Memory Allocation” on page 27
- ♦ “Synchronizing Server Dates and Times” on page 28

### Setting Up the Web Server Environment

The OnDemand Services portal requires one of the Web server environments listed in “**Web Server**” on page 19. Before continuing, you should make sure that a supported environment exists and is functioning properly.

**IMPORTANT:** If you plan to install the portal on a NetWare® 6 server and need help installing and configuring Apache 1.3.27 and Tomcat 4.1.18 on the server, please refer to [Technical Information Document \(TID\) 10081429](http://support.novell.com/cgi-bin/search/searchtid.cgi?/10081429.htm) (<http://support.novell.com/cgi-bin/search/searchtid.cgi?/10081429.htm>) .

In addition, the portal Web server must meet the Novell eDirectory™ requirements listed in “**Novell eDirectory and ConsoleOne**” on page 20. If it does not, take the necessary steps fulfill the requirements.

### Increasing the Tomcat Memory Allocation

To ensure that Tomcat has sufficient memory to accommodate the OnDemand Services portal, you should increase the memory allocated to the Tomcat Java process.

The amount you allocate is dependent on the server’s available memory, the number of Web Self-Service users who will be supported, and so forth. We recommend that the initial memory allocation be no less than 128 MB for the first 200 users and that you increase the allocation by 64 MB for every 200 additional users. The maximum memory allocation should be limited only by the amount of server memory you want allocated to the Java process.

#### NetWare 6 Server

- 1 Open the tomcat4.ncf file, located in the sys:\tomcat\4\bin directory.
- 2 On the Java line, change the initial memory allocation parameter (Xms) and the maximum memory allocation parameter (Xmx).

The following example shows the recommended parameter settings for supporting up to 200 users.

```
-Xms128m -Xmx256m
```

The maximum heap size parameter (Xmx) should reflect the maximum amount of the server's memory you want used by the Java servlet engine. On NetWare, this amount cannot exceed 400 MB; if you enter a value greater than 400, the Java servlet engine will revert to using only 32 MB.

**3** Save the file.

**4** Skip to [Synchronizing Server Dates and Times](#) below.

## Windows 2000 Server

**1** On the Windows server, click the Start menu > Settings > Control Panel > System > Advanced > Environment Variables.

**2** Under System Variables, click New, then fill in the following fields:

**Variable Name:** TOMCAT\_OPTS

**Variable Value:** -Xms128m -Xmx256m

**3** Click OK to save the variable, then close the Environment Variables dialog box.

**4** Continue with the next section, [Synchronizing Server Dates and Times](#).

## Synchronizing Server Dates and Times

The OnDemand Services gadgets, the Maintenance process, and the UsageServer process all place time stamps on Purchase and Usage objects in eDirectory and perform various tasks (expiring purchases, deleting objects, and so forth) based on the time stamps.

By default, the OnDemand Services gadgets are installed to a portal server and the Maintenance and UsageServer processes are installed to a ZENworks for Desktops (ZfD) server. If these are different servers, you need to synchronize the dates and times of the two servers to ensure the accurate execution of any tasks based on time stamps.

# 5

## Installing Web Self-Service

To install and configure Novell® ZENworks® Web Self-Service, complete the following tasks:

- ♦ “Installing the Portal Software” on page 29
- ♦ “Assigning Rights to the OnDemand Admin User” on page 33
- ♦ “Setting Up ZENworks Application Association” on page 34
- ♦ “Setting Up Cost Centers” on page 35
- ♦ “Configuring the Catalog Root and ApprovalFlow E-Mail Settings” on page 39
- ♦ “Assigning Portal Access” on page 40
- ♦ “Enabling Usage Tracking of ZfD DeFrame Thin-Client Applications” on page 46
- ♦ “Starting the Maintenance and UsageServer Processes” on page 46

After you’ve completed the above tasks, see “What’s Next” on page 48 for directions on what you need to do to start using Web Self-Service.

If you are upgrading from ZENworks OnDemand Services™ 2, see “Upgrade” on page 49.

### Installing the Portal Software

Complete the tasks in the following sections to install the Web Self-Service software:

- ♦ “Installation Worksheet” on page 29
- ♦ “Installing the OnDemand Services Portal” on page 31
- ♦ “Installing the OnDemand Services Gadgets to an Existing Novell exteNd Director 4.1 Portal” on page 32

### Installation Worksheet

The following worksheet lists the information you will be asked to provide during the installation process. You might want to print the worksheet and record your information for use during installation.

Item	Description
1) LDAP Server Login <ul style="list-style-type: none"> <li>Server address and port:</li> <li>Admin user context:</li> <li>Admin user password:</li> </ul>	<p>The installation program uses LDAP to access Novell eDirectory™. You need to supply the address (hostname or IP address) and port number an LDAP server that can provide access to the eDirectory tree where you want to create the portal objects. In addition, you need to provide the LDAP context and password of the Admin user (or Admin-equivalent user).</p> <p>Example:</p> <ul style="list-style-type: none"> <li>LDAP Server address and port: 123.456.78.90:389</li> <li>Admin user context: cn=admin,o=novell</li> <li>Admin user password: bigredn</li> </ul>
2) Archive to Deploy <ul style="list-style-type: none"> <li>ZENworks.xar</li> </ul>	<p>The ZENworks.xar archive file contains the files that will be extracted to the Tomcat webapps directory. The installation program will list the ZENworks.xar file automatically. If it does not, you will be able to browse for it on the <i>ZENworks 6 Web Self-Service Program</i> CD.</p>
3) Installation Type <ul style="list-style-type: none"> <li>Custom</li> <li>Typical</li> </ul>	<p>The installation program updates the eDirectory schema, creates portal objects, installs the portal modules, and sets the base portal user container. Users who reside under the base portal user container only need to enter a common name (jsmith) rather than a distinguished name (cn=jsmith,ou=users,o=novell) when logging in to the portal.</p> <p>A custom install lets you determine the names and contexts (that is, the distinguished names) for the portal objects, and select the base portal user container.</p> <p>A typical installation uses default distinguished names for the portal objects and sets the logged-in administrator's Organization container as the base portal container.</p>
4) Portal Object <ul style="list-style-type: none"> <li>Distinguished name:</li> <li>Password seed:</li> </ul>	<p>The portal is represented in eDirectory by a Portal object.</p> <p>If you perform a typical installation, the Portal object is created with the following distinguished name:</p> <ul style="list-style-type: none"> <li>cn=Portal,ou=Portal,o=org</li> </ul> <p>If you perform a custom installation, you can specify the distinguished name and password seed for the Portal object. The password seed is combined with the portal GUID to generate the portal's password.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>Distinguished name: cn=Portal,ou=services,o=novell</li> <li>Password seed: novell</li> </ul>

Item	Description
5) Public User Object:	The portal requires a public user in eDirectory.
♦ Distinguished name:	If you perform a typical installation, the Public User object is created with the following distinguished name:
♦ Password:	<ul style="list-style-type: none"> <li>♦ <code>cn=PublicUser,ou=Portal,o=org</code></li> </ul> <p>If you perform a custom installation, you can specify the distinguished name and password for the Portal User object.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>♦ Distinguished name: <code>cn=PublicUser,ou=services,o=novell</code></li> <li>♦ Password: <code>novell</code></li> </ul>
6) User Contexts	<p>To configure the portal for contextless login (meaning users only need to enter a common name rather than a distinguished name when logging in), you can specify the contexts where User objects reside.</p> <p>If you perform a typical installation, the user context is set to the Organization container (O) where the Public User object is being created.</p> <p>If you perform a custom installation, you can select one or more user containers. Any users in the selected containers and their subcontainers will be able to use contextless login.</p> <p>Example:</p> <pre>ou=users,o=novell ou=users,ou=services,o=novell</pre>
7) OnDemand Admin User Object	The OnDemand Services gadgets authenticate and receive rights to eDirectory through the OnDemand Admin User object.
♦ Distinguished name:	At installation, the OnDemand Admin User object receives Supervisor rights to its container and subcontainers. This gives the OnDemand Services gadgets the rights they need to User objects, Application objects, and various other objects. We recommend you create the OnDemand Admin user in a container that encompasses all of the objects listed in <a href="#">“Assigning Rights to the OnDemand Admin User” on page 33</a> .
♦ Password:	
	<p>Example:</p> <ul style="list-style-type: none"> <li>♦ Distinguished name: <code>cn=OnDemandAdminUser,o=novell</code></li> <li>♦ Password: <code>novell</code></li> </ul>

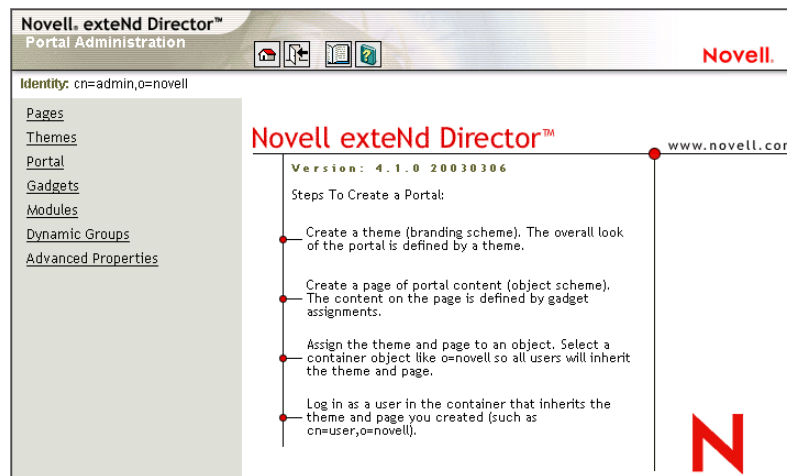
## Installing the OnDemand Services Portal

The following sections explain how to set up an OnDemand Services portal on a NetWare® 6 or Windows 2000 server. To install the OnDemand Services gadgets to an existing Novell exteNd Director™ 4.1 Standard Edition portal, skip to [“Installing the OnDemand Services Gadgets to an Existing Novell exteNd Director 4.1 Portal” on page 32](#).


- 1 Make sure the server meets the requirements listed in [“Software Requirements” on page 19](#).

- 2** Copy the NPS.war and ZENworks.xar files from the root of the *ZENworks 6 Web Self-Service Program* CD to the Tomcat webapps directory (for example, sys:\tomcat\4\webapps or c:\jakarta-tomcat-4.1.18\webapps).
- 3** Restart Tomcat.  
Tomcat extracts the NPS.war to the appropriate locations in the webapps directory.
- 4** In a Web browser, enter the following URL to launch the installation/configuration servlet:  
`http://server_address/nps/servlet/configure`  
where *server\_address* is the server's IP address or hostname.
- 5** Click Start to display the License Agreement page.
- 6** After reviewing the license agreement, click I Accept the Terms of the License Agreement, then click Next to display the LDAP Directory Server page.
- 7** Follow the prompts to complete the installation. If you filled out the **installation worksheet**, use the information you recorded.

After installation is complete, you are logged into the portal using the credentials you provided during the installation. The portal appears with the Portal Administration page displayed.



At this point, there are additional installation and configuration tasks you need to complete outside of the portal before using the Portal Administration page to configure the portal.

- 8** Click the Exit button  to exit the portal.
- 9** If the portal is on a NetWare 6 server, restart the server.
- 10** Skip to **“Setting Up ZENworks Application Association” on page 34.**

## Installing the OnDemand Services Gadgets to an Existing Novell exteNd Director 4.1 Portal

The following steps explain how to install the OnDemand Services gadgets to an existing Novell exteNd Director 4.1 Standard Edition portal on a NetWare 6 or Windows 2000 server.

If you don't have an existing exteNd Director 4.1 portal, you need to install the OnDemand Services portal. See **“Installing the OnDemand Services Portal” on page 31.**



- 1** At a Windows workstation from which you can administer your exteNd Director 4.1 portal, insert the *ZENworks 6 Web Self-Service Program* CD into the CD drive.

The *ZENworks 6 Web Self-Service Program* CD contains the portal module file, *ZENworks.npm*, that includes the OnDemand Services gadgets that you will install to the portal.
- 2** Log in to your portal as an administrator.
- 3** Click Portal Administration > Administer the Portal to display the Portal Administration page.
- 4** Click Modules to display the Modules page.
- 5** Click Install to display the Install a New Module page.
- 6** Fill in the following fields:

**Unique Name for the Module:** Enter a name for the module that is different than any other installed modules (for example, ZENworks).

**Description:** This field is optional. If desired, enter text to describe the module (for example, ZENworks 6 OnDemand Services Gadgets).

**Path to Module File Package:** Click Browse, then browse to and select the *ZENworks.npm* file on the root of the *ZENworks 6 Web Self-Service Program* CD.
- 7** Click Install.

After the OnDemand Services gadgets are installed, an OnDemand Admin User and Password page is displayed. The OnDemand Services gadgets authenticate to eDirectory through this user.
- 8** Fill in the following fields:

**OnDemand Admin User:** Using LDAP syntax, enter a name and context for the OnDemand Admin user (for example, cn=OnDemandAdminUser,ou=services,o=novell,c=us). The containers you specify (for example, ou=services,o=novell,c=us) must already exist. If the User object does not exist (for example, cn=OnDemandAdminUser), it will be created.

**Password:** Enter a password for the user.
- 9** Click OK to create the OnDemand Admin user object and complete the installation.
- 10** When installation is complete, exit the portal.
- 11** If the portal is on a NetWare 6 server, restart the server.

or

If the portal is on a Windows 2000 server, either restart the server or restart Tomcat.
- 12** Continue with the next section, [Assigning Rights to the OnDemand Admin User](#).

## Assigning Rights to the OnDemand Admin User

The OnDemand Services gadgets authenticate to eDirectory through the OnDemand Admin User object.

During installation, the OnDemand Admin User object is assigned Supervisor rights to the container where it resides. As long as the objects listed in the following table reside in or beneath the OnDemand Admin User object's container, the OnDemand Services gadgets will have the eDirectory rights they need.

If all objects do not reside beneath the OnDemand Admin User object's container, you can select a base container that encompasses all of the objects and give the OnDemand Admin User object Supervisor rights to the container. Or, you can use the following table to assign appropriate rights to individual objects or containers.

Object	Trustee Rights	Explanation
Company root container	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read	The <b>company root container</b> (as defined on the OnDemandService object) and all subcontainers.
User containers	[Entry Rights] - Browse, Create [All Attributes Rights] - Compare, Read, Write	All containers where User objects reside.
Catalog container	[Entry Rights] - Browse, Create [All Attributes Rights] - Compare, Read, Write	The <b>catalog root container</b> (as defined on the OnDemandService object) and all subcontainers.
Application objects containers	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read, Write	All containers where Application objects reside.
Commerce Item objects	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read	All containers where Commerce Item (Web application) objects reside.
iChain ACL Rule objects	[Entry Rights] - Browse, Create [All Attributes Rights] - Compare, Read, Write	All iChain® ACL rules that control access to Web applications.
OnDemandService object	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read	The OnDemandService object located in the same directory as the OnDemand Admin User object.
Cost Center objects	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read, Write	All <b>cost center</b> (Organizational Role) objects

## Setting Up ZENworks Application Association

In order for a user to gain access to an application, the OnDemand Services gadgets must create an association in eDirectory between the user's object and the application's object.

If you installed the OnDemand Services portal on a NetWare server (or the OnDemand Services gadgets to an exteNd Director 4.1 portal on a NetWare server), the zenapp32.nlm program used by the OnDemand Services gadgets to create application associations is automatically installed to the sys:\system directory. It will be loaded the first time it is needed. You can skip the rest of this section and continue with the next section.

If you installed the OnDemand Services portal on a Windows server (or the OnDemand Services gadgets to an exteNd Director 4.1 portal on a Windows server), you need to complete the following steps to enable the OnDemand Services gadgets to create application associations:

- 1 Install the Novell Client™ and the ZENworks for Desktops 4.0.1 Management Agent to the Windows server.

On a Windows server, the OnDemand Services gadgets use the Application Management components of the ZfD Management Agent to access eDirectory and create application associations. The Novell Client is required to access eDirectory. For information about installing the ZfD Management Agent and the Novell Client, see the *ZfD Installation* guide at the [ZENworks for Desktops 4.0.1 documentation Web site \(http://www.novell.com/documentation/lg/zdpr\)](http://www.novell.com/documentation/lg/zdpr).

- 2 At the Windows server, log in to eDirectory through a user account that has rights to all containers where Application objects and User objects reside.

By default, the OnDemand Admin User object must have rights to all containers where Application objects and User objects reside. You can log in at the Windows server as the OnDemand Admin user or any other user with sufficient rights (Admin, for example).

## Setting Up Cost Centers

Cost centers are required for both user self-service and administrator provisioning of application packages. You must assign each user to a cost center, and each cost center must have one or more budget holders.

When a user requests an application, the budget holder assigned to the user's cost center approves or denies the request. Likewise, for a budget holder to provision an application to a user, the user must be assigned to the budget holder's cost center.

If a purchase or usage fee is associated with an application package, the package charges are accrued to the user's cost center.

Depending on your organization's needs, you can create one cost center for all users, or you can create multiple cost centers. The following sections explain how to create cost centers, add budget holders, and assign users:

- ♦ "Creating a Cost Center" on page 35
- ♦ "Adding Budget Holders" on page 36
- ♦ "Assigning Users" on page 37

## Creating a Cost Center

You create cost centers in eDirectory by using Organizational Role objects. Each user you add as an occupant to the organizational role becomes a budget holder for the cost center.

- 1 In ConsoleOne®, right-click the container where you want to create the cost center, click New, then click Object.
- 2 Select Organizational Role in the Class list, then click OK to display the New Organizational Role dialog box.
- 3 Enter a name for the cost center (for example, DEPT12345), then click OK to create the object.  
**IMPORTANT:** Do not use an & character in the name.
- 4 Right-click the newly created object, then click Extensions of this Object.
- 5 In the Extensions of this Object dialog box, click Add Extension.
- 6 In the list of auxiliary class extensions, select commerceBudgetHolderRole, then click OK to display the BudgetHolder page.

**7** Fill in the following fields:

**Current Budget Holder:** The cost center can have more than one budget holder. This allows for reassignment of the budget holder responsibilities from one person to another (for example, if a budget holder goes on vacation). The current budget holder determines which budget holder is active. Select the user you want to initially assign as the current budget holder.

**User Containers:** This field remains for backwards compatibility with previous versions of Novell ZENworks OnDemand Services. It is not used with Web Self-Service. Leave it blank.

**8** Click OK to close the page, then click Close to close the Extensions of this Object dialog box.

**9** To create additional cost centers, repeat **Step 1** through **Step 8**.

or

To add additional users as budget holders for the cost center, continue with the next section, **Adding Budget Holders**.

or

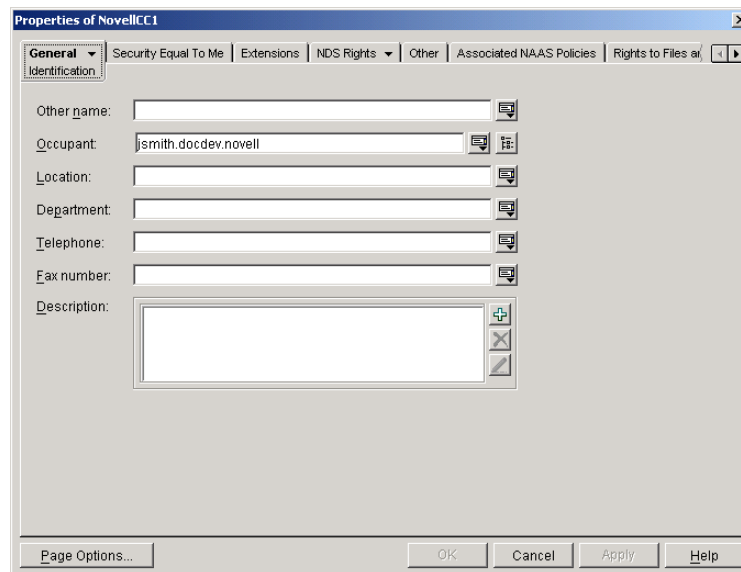
To assign users to the cost center, skip to **“Assigning Users” on page 37**.

## Adding Budget Holders

When you create a cost center (see the previous section, **Creating a Cost Center**), you assign a user to be the current budget holder. Only the current budget holder can approve requests for the cost center. You can, however, add additional budget holders who can be assigned as the current budget holder at any time. The current budget holder assignment can be changed in ConsoleOne, or any of the budget holders can use the Current Budget Holder gadget to change the assignment.

To add budget holders to a cost center:

**1** In ConsoleOne, right-click the cost center’s object, then click Properties.



**2** On the Identification page, use the Occupant field to add the users who will be the cost center’s budget holders.

- 3 Click OK to save your budget holder changes.
- 4 Continue with the next section, **Assigning Users**.

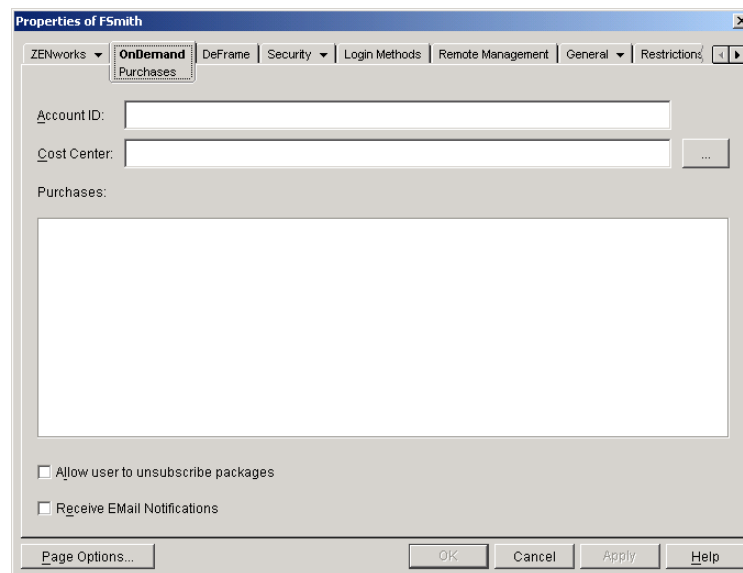
## Assigning Users

You can use ConsoleOne to individually assign users to a cost center, or you can use the OnDemand User Configuration Utility to assign multiple users to a cost center at one time. The following sections provide instructions for both methods:

- ♦ “Using ConsoleOne to Individually Assign Users” on page 37
- ♦ “Using the OnDemand User Configuration Utility to Assign Multiple Users” on page 38

### Using ConsoleOne to Individually Assign Users

- 1 In ConsoleOne, right-click the User object, then click Properties.
- 2 Click the OnDemand tab to display the Purchases page.



- 3 Fill in the following fields:

**Account ID:** The user’s account ID is recorded on each Purchase and Usage object that he or she generates. If you have a third-party billing system or reporting system, you can base your billing or reporting on the account ID.

The account ID can be a unique identifier, such as the user’s employee workforce number, or it can be a common identifier, such as the user’s cost center ID as it is defined in the billing or reporting system.

**Cost Center:** Browse to and select the cost center that you want the user assigned to. Cost centers are represented by Organizational Role objects.

**Allow User to Unsubscribe Packages:** If you want the user to be able to cancel his or her subscription to a package, select this option. If you do so, the user can use the Workflow Tracking gadget to view an approved subscription purchase and cancel the subscription.

**Receive E-Mail Notifications:** Select this option to enable the user to receive e-mail messages notifying him or her of approved or denied purchases.

- 4 Click OK to save the information.

## Using the OnDemand User Configuration Utility to Assign Multiple Users

The OnDemand User Configuration utility lets you do the following:

- ♦ Enable users as ZfD DeFrame™ users. If you've already enabled users as DeFrame users, the utility will not affect the DeFrame settings.
- ♦ Configure the cost center and account ID settings required for ApprovalFlow™.

This Java-based utility is included on the *ZENworks 6 Companion 1* CD. Before you can use it, you will need to install it to a server or workstation. You will then need to run it from a Windows workstation/server that has a Java Runtime Engine (JRE) installed.

- 1 Copy the files from the `zenworksfordesktops\odusrcfg` directory on the *ZENworks 6 Companion 1* CD to a server or workstation directory.

The files can be copied to any server or workstation location. One recommended location is the `ConsoleOne` directory.

- 2 From a Windows workstation/server that has a JRE installed, run `odusrcfg.bat` from the directory to display the OnDemand User Configuration Utility Introduction page.

- 3 Click Next to display the Novell eDirectory Authentication page.

- 4 Fill in the following fields:

**LDAP Server Hostname or IP Address:** Enter the hostname or IP address of an LDAP server that provides access to eDirectory. If the LDAP server's port is not 389, include the port number.

**Admin Name:** Using LDAP syntax, specify the context of a user that has admin-equivalent eDirectory rights.

**Password:** Specify the user's password.

- 5 Click Next to authenticate to eDirectory and display the User Selection page.

- 6 Select the Enable ApprovalFlow option.

- 7 Select the users you want to assign to a cost center.

All users you select will be assigned to the same cost center. You can select individual users, or you can select a container to add all the container's users.

- 8 Click Next to display the ApprovalFlow Settings page.

- 9 Fill in the following fields:

**Account ID:** Enter an account ID to identify the user. The user's account ID is recorded on each Purchase and Usage object that he or she generates. If you have a third-party billing system or reporting system, you can base your billing or reporting on the account ID.

**Cost Center:** Browse to and select the cost center that you want the user assigned to. Cost centers are represented by Organizational Role objects.

**Enable User to Receive E-Mail Notifications:** Select this option to enable the user to receive e-mail messages notifying him or her of approved or denied purchases.

**Enable User to Cancel Package Subscriptions:** If you want the user to be able to cancel his or her subscription to an application package, select this option. If you do so, the user can use the Workflow Tracking gadget to view an approved subscription purchase and cancel the subscription.

**Overwrite Existing ApprovalFlow Settings:** Select this option to have these settings overwrite any ApprovalFlow settings currently assigned to the selected users.

**10** Click Finish, then click Yes to confirm that you want to configure the selected users.

**11** When the user configuration is complete, click Finish to exit the utility.

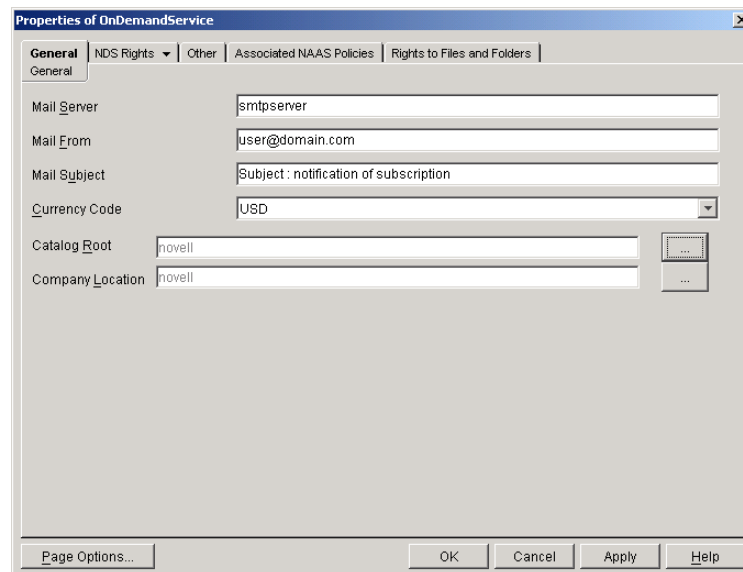
## Configuring the Catalog Root and ApprovalFlow E-Mail Settings

The installation program creates an OnDemandService object in the same eDirectory container as the OnDemand Admin User object. The OnDemandService object includes settings that control the Package Request gadget's catalog root and the e-mail server, address, and subject used with ApprovalFlow messages.

To configure the settings:

- 1** In ConsoleOne, right-click the OnDemandService object, then click Properties to display the object's property pages.

The OnDemandService object is located in the same directory as the OnDemand Admin User object.

The screenshot shows a Windows-style dialog box titled "Properties of OnDemandService". It has several tabs: "General", "NDS Rights", "Other", "Associated NAAS Policies", and "Rights to Files and Folders". The "General" tab is selected. Inside the dialog, there are several labeled text fields: "Mail Server" with the value "smtpserver", "Mail From" with the value "user@domain.com", "Mail Subject" with the value "Subject: notification of subscription", "Currency Code" with a dropdown menu showing "USD", "Catalog Root" with the value "novell", and "Company Location" with the value "novell". To the right of the "Catalog Root" and "Company Location" fields are buttons with ellipses (...). At the bottom of the dialog are buttons for "Page Options...", "OK", "Cancel", "Apply", and "Help".

- 2** Fill in the following fields:

**Mail Server:** Enter the fully distinguished domain name of the mail server (for example, mail.novell.com) through which the OnDemand Services gadgets can send e-mail notifications. Notifications are sent when a user requests a package and when a budget holder approves or denies a package request.

**Mail From:** Enter an e-mail address to display in the e-mail message's From field. This must be a valid e-mail address (for example, jsmith@mail.novell.com).

**Mail Subject:** Enter the text you want displayed in the e-mail message's Subject field. This field is optional.

**Currency Code:** Select the currency code you want associated with any monetary fields (for example, USD for United States dollar, GBP for United Kingdom pound sterling, FRF for French franc, and so forth). For a description of the currency codes, click the Help button and see the Currency Code section.

**Catalog Root:** The OnDemand Services Package Request gadget displays a catalog of application packages that are available for users to request.

By default, the OnDemand Admin User object's container is assigned as the catalog root. This means users will see all eDirectory containers and application packages that reside under the catalog root.

We recommend that you specify a catalog root that limits exposure of your eDirectory tree. You might want to create a container specifically for your application packages (for example, packages.novell) and specify the packages container as the catalog root. If necessary, you can create additional containers under the packages container for organizational purposes.

To specify a new catalog root container, click the browse button and select a container. The container must already exist.

**Company Location:** This field is designed primarily for customers whose eDirectory tree might contain more than one company's users and packages. This field specifies the root container for the company associated with this OnDemandService object.

- 3 Click OK to save your changes.

## Assigning Portal Access

Each OnDemand Services gadget represents a window to specific content or functionality. For a gadget to be displayed, it must be added to a portal page. During installation of the OnDemand Services portal (or the OnDemand Services gadgets to an exteNd Director 4.1 portal), the four pages listed in the following table are created for the gadgets.

Page Name	Gadgets	User Assignments
Applications	OD_LaunchItemGadget: Used to launch applications.	All users (including budget holders)
OnDemand Services	OD_LaunchItemGadget: Used to launch applications.	All users (including budget holders)
	OD_PackageRequestGadget: Used to request an application package.	
	OD_WorkflowTrackingGadget: Used to check the status of requests and to cancel requests.	
	OD_ReportGadget: Used to generate reports for subscription and usage-based applications.	



Page Name	Gadgets	User Assignments
Cost Center Management	OD_UserAdminGadget: Used to assign application packages to users.  OD_CurrentBHGadget: Used to add or remove a cost center's budget holders, and to change a cost center's current (active) budget holder.	budget holders
Purchase Approval	OD_ApprovalGadget: Used to approve or deny users' requests for application packages.	budget holders

You must assign the appropriate pages to users and budget holders before they will have access to the pages. The following sections provide instructions:

- ♦ “Assigning Pages to Users” on page 41
- ♦ “Assigning Pages to Budget Holders” on page 43

**NOTE:** You are not required to use the default pages created during installation. They are included to enable you to get started quickly using Web Self-Service. If you want to create your own pages, or add the OnDemand Services gadgets to pages in an existing Novell exteNd Director 4.1 Standard Edition portal, see the documentation at the [Novell exteNd Director 4.1 Standard Edition documentation Web site \(http://www.novell.com/documentation/lg/nedse41\)](http://www.novell.com/documentation/lg/nedse41)

## Assigning Pages to Users

You can give users access to two pages, the Applications page and the OnDemand Services page. Generally, you will not want to assign both pages to users. The following table provides the reasons for using each page.

Page Name	Gadgets	Reasons to Use this Page
Applications	OD_LaunchItemGadget	This page includes only the Launch Item gadget, which means that users are limited to launching only administrator-provisioned applications. They cannot request applications.  Use this page if you don't want to support user requests for applications, or if you want to provide two pages (Applications and OnDemand Services) from which users can launch applications.
OnDemand Services	OD_LaunchItemGadget OD_PackageRequestGadget OD_ReportGadget OD_WorkflowTrackingGadget	This page includes all the gadgets required for users to both request applications and launch applications.  Use this page to support user requests for applications.

During the OnDemand Services portal installation (or installation of the OnDemand Services gadgets in an exteNd Director 4.1 portal), all users located within the portal containers (Portal Administration page > Portal > Configuration Properties) are assigned access to the Applications page. If you plan to use the Applications page and all users are located within your portal user

containers, you can skip the following steps. If you want to use the OnDemand Services page or need to change (or verify) user assignments for the Applications page, complete the following steps:

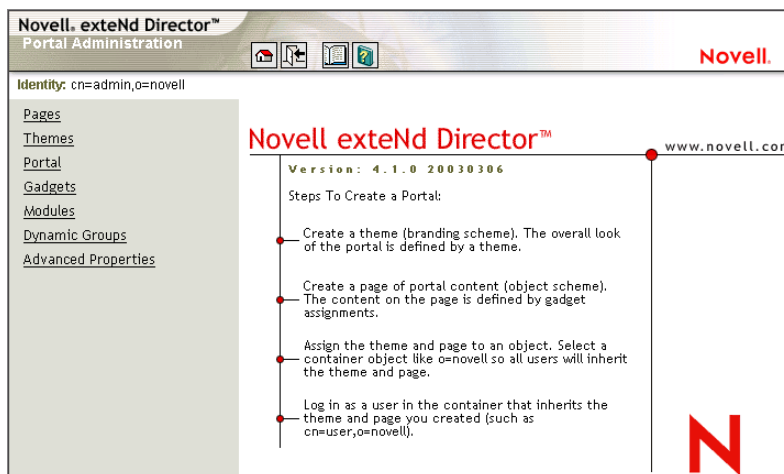
**1** Log in to the portal as an administrator:

**1a** Enter `http://server_address/nps/servlet/portal`, where *server\_address* is the IP address or hostname of your portal server, to display the login page.

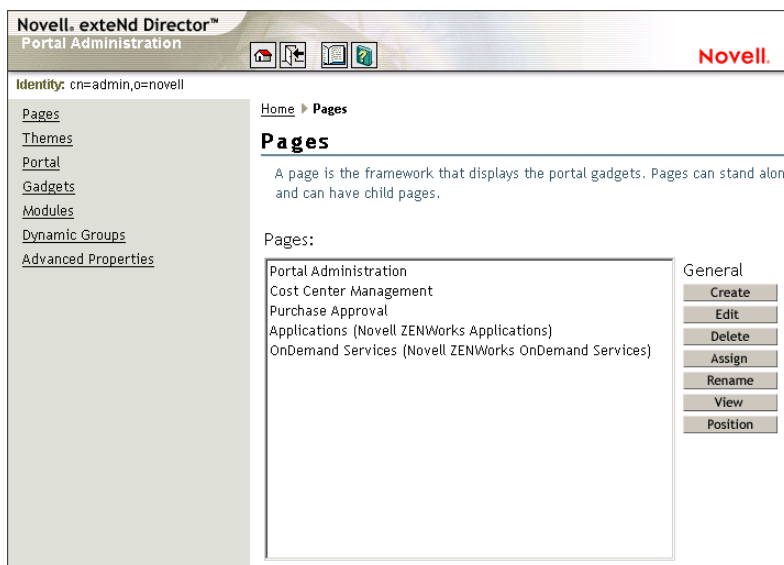
**1b** Enter your administrator username and password.

If you are using the OnDemand Services portal, you need to use the same username you supplied when installing the portal.

**2** Click Administer the Portal to display the Portal Administration page.



**3** Click Pages to display the Pages page.



**4** To change or verify the Applications page assignments:

**4a** Select the Applications page, then click Assign.

By default, the page assignments include all containers defined as your portal containers (Portal Administration page > Portal > Configuration Properties). You can assign the page to additional users by adding individual User objects, Group objects in which the users are members, or container objects where the users reside.

**4b** To add a user, click Add.

You can use the Search For field to search for a User, Group, or container object, or, if you know the context of the object you want to add, you can enter the context in the Object Name field. Choose the method you want to use, then follow the prompts to add the object.

**4c** To remove a user, user group, or user container, select the object, then click Remove.

**5** To assign the OnDemand Services page to users:

**5a** Select the OnDemand Services page, then click Assign.

**5b** Click Add.

You can assign the page to users by adding individual User objects, Group objects in which the users are members, or container objects where the users reside. By default, no objects are listed.

**5c** If you don't know the context of the object you want to add, select the object type in the Search For field, click OK, then follow the prompts.

or

If you know the context of the object you want to add, type the context in the Object Name field, then click OK.

**6** When you are finished making page assignments, click Close (on the Assign Page to Objects page).

**7** Continue with the next section, [Assigning Pages to Budget Holders](#).

By default, the OnDemand Services gadgets used on the Applications page and OnDemand Services page do not require any modifications to their configuration. If you want to view the configuration settings or change them, see [Appendix A, "Gadget Configuration Settings," on page 127](#).

## Assigning Pages to Budget Holders

You can give budget holders access to two separate pages: the Cost Center Management page and the Purchase Approval page. The following table provides the reasons for using each page.

Page Name	Gadgets	Reasons to Use this Page
Cost Center Management	OD_UserAdminGadget OD_CurrentBHGadget	This page enables budget holders to assign applications to users, add or remove budget holders for the cost center, and change the cost center's current (active) budget holder.

Page Name	Gadgets	Reasons to Use this Page
Purchase Approval	OD_ApprovalGadget	<p>This page enables budget holders to approve or deny user requests for applications.</p> <p>If you've assigned the OnDemand Services page to users and given them the ability to request applications, you need to assign this page to budget holders.</p>

The following sections provide instructions:

- ♦ “Creating a Dynamic Group for Budget Holders” on page 44
- ♦ “Assigning Pages to the Budget Holder Dynamic Group” on page 45

## Creating a Dynamic Group for Budget Holders

A budget holder dynamic group lets the portal automatically assign or remove page access when a budget holder is added to or removed from a cost center. A dynamic group is the best way to assign your budget holder pages because it is updated on-the-fly as users are added or removed as budget holders.

- 1** On the Portal Administration page, click Dynamic Groups.
- 2** Click Create to display the Create Dynamic Group page, then fill in the following fields:  
**Name:** Enter a name for the budget holder dynamic group (for example, BudgetHolders).

**Query Filter String:** Enter the following:

```
( |(securityequals=cn=cost_center1_context)
(securityequals=cn=cost_center2_context)
(securityequals=cn=cost_center3_context)... )
```

where *cost\_center\_context* is the context (using LDAP syntax) of the Organizational Role objects that are serving as cost centers. For example:

```
( |(securityequals=cn=SalesCC,ou=services,o=novell)
(securityequals=cn=MarketingCC,ou=services,o=novell) )
```

In this example, all users who have been added as budget holders on the SalesCC and MarketingCC cost centers will be included in the dynamic group.

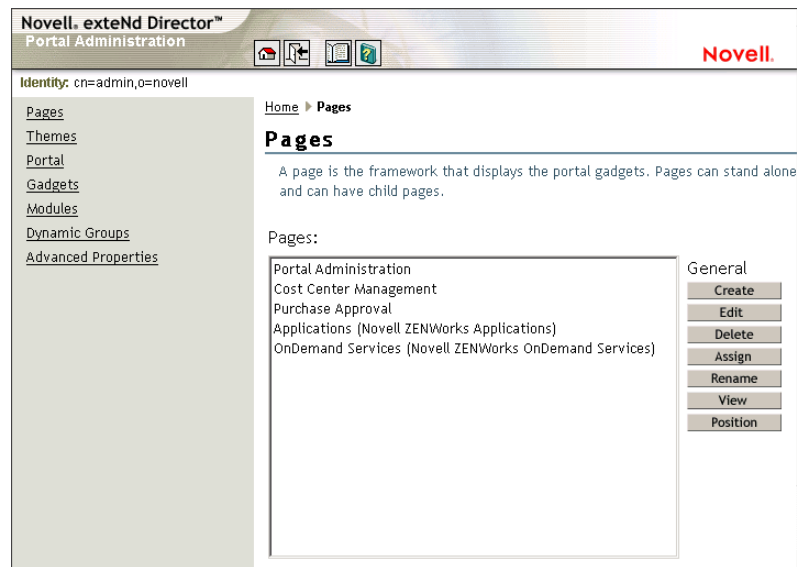
To test the query filter string, click Test, then follow the prompts.

**Container:** By default, the portal will search all portal containers. If you want to limit the search, enter a search container.

- 3** Click Create to create the dynamic group, then click OK to close the Create Dynamic Group page.
- 4** Continue with the next section, [Assigning Pages to the Budget Holder Dynamic Group](#).

## Assigning Pages to the Budget Holder Dynamic Group

- 1 Click Pages to display the Pages page.



- 2 To assign the Cost Center Management page to the budget holder dynamic group:
  - 2a Select the Cost Center Management page, then click Assign.
  - 2b Click Add to display the Select Object Type page.
  - 2c In the Search For field, select Dynamic Groups, then click OK to display the Select Object page.
  - 2d In the Available Dynamic Groups list, select the budget holder dynamic group, then click Assign.
  - 2e Click Close to close the Assign Page to Objects page.
- 3 To assign the Purchase Approval page, select the Purchase Approval page, click Assign, then repeat **Step 2b** through **Step 2e**.
- 4 Continue with the next section, **Enabling Usage Tracking of ZfD DeFrame Thin-Client Applications**.

By default, the OnDemand Services gadgets used on the Cost Center Management page and Purchase Approval page do not require any modifications to their configuration. If you want to view the configuration settings or change them, see **Appendix A, “Gadget Configuration Settings,” on page 127**.

# Enabling Usage Tracking of ZfD DeFrame Thin-Client Applications

If you plan to track usage of ZfD DeFrame thin-client applications, you need to install the UsageClient to each ZfD DeFrame terminal server.

- 1 Launch setup.exe from the \WorkstationSetup directory on the *ZENworks 6 Web Self-Service Program* CD.
- 2 Follow the prompts.

**NOTE:** For usage tracking of ZfD desktop applications, the UsageClient must be installed on each user's workstation. This happens automatically the first time a usage-based desktop application is launched from the OnDemand Services Launch Item gadget. If you have locked down workstations so that users don't have the file system rights needed to install applications, you need to log in with administrator rights at each workstation and launch a usage-based desktop application from the Launch Item gadget or run the setup.exe program.

## Starting the Maintenance and UsageServer Processes

When a user receives access to an application package, a Purchase object is created in eDirectory below the Package object. If there are usage-based applications in the package, each time the user launches one of the usage-based applications a Usage object is created below the user's Purchase object.

The Maintenance process cleans up expired Purchase and Usage objects in eDirectory and closes out tracking of usage-based applications when the UsageServer becomes unavailable. The Maintenance process must to be running at all times.

The UsageServer process is required for tracking of usage-based applications. If you don't plan to use usage-based applications, it does not need to be running.

The Maintenance and UsageServer processes are installed with the ZfD server software. You can run the processes on a ZfD server, or you can move them to another server. The following sections explain how to run the processes on a ZfD server. For information about moving the processes to another server, see [“Installing Additional Maintenance and UsageServer Processes” on page 120](#).

- ♦ [“Starting the Maintenance Process” on page 46](#)
- ♦ [“Starting the UsageServer Process” on page 47](#)

## Starting the Maintenance Process

- 1 At the NetWare server console, enter  
**sys:\zenworks\ondemand\bin\commercemaintenance.ncf.**

or

At the Windows server command prompt, change to the directory where the Maintenance process was installed (by default, zenworks\ondemand\bin), then enter  
**commercemaintenance.bat.**

To avoid entering the entire executable path or changing to the executable directory, add the zenworks\ondemand\bin path to the server's search path. In addition, if you want the Maintenance process to start automatically during a NetWare server startup, you can uncomment the following lines in the server's autoexec.ncf file:

```
#SEARCH ADD SYS:\zenworks\OnDemand\bin
#CommerceMaintenance.ncf
```

- 2 When prompted, enter the password for the Admin or Admin-equivalent user that the Maintenance process will use for LDAP authentication to eDirectory.

By default, the Maintenance process will authenticate with the username used when installing the ZfD server software. If you don't know the username under which the ZfD server software was installed, or if this is not the username through which you want the Maintenance process to authenticate, you can change it in the `commerceserver.cfg` file, located in the `sys:\zenworks\ondemand` directory. You should check the following settings:

**ldap.host:** Specifies the IP address or DNS host name of the LDAP server used to authenticate to eDirectory. The default IP address, 127.0.0.1, provides authentication through the local LDAP server.

**ldap.loginName:** Specifies the user through which authentication occurs. Use LDAP syntax to specify the user's distinguished name (for example, `cn=admin,o=novell`).

If the user's distinguished name includes extended (multibyte) characters and the Maintenance process is running on a NetWare server, you need to run `commerceserver.cfg` through the Java utility `native2ascii`. At the server console, enter the following:

```
native2ascii \zenworks\ondemand\commerceserver.cfg
```

This needs to be done any time you change the user name.

**baseContext:** Specifies the container that the Maintenance process will use as its root container. The Maintenance process only processes Purchase and Usage objects located in the root container and its subcontainers. Use LDAP syntax to specify the root container distinguished name (for example, `ou=apps,ou=services,o=novell`).

## Starting the UsageServer Process

- 1 At the NetWare server console, enter **usageserver.ncf**.

or

At the Windows server command prompt, change to the directory where the UsageServer process was installed (by default, `zenworks\ondemand\bin`), then enter **usageserver.bat**.

To avoid entering the entire executable path or changing to the executable directory, add the `zenworks\ondemand\bin` path to the server's search path. In addition, if you want the UsageServer process to start automatically during a NetWare server startup, you can uncomment the following lines in the server's `autoexec.ncf` file:

```
#SEARCH ADD SYS:\zenworks\OnDemand\bin
#UsageServer.ncf
```

- 2 Enter the password for the Admin or Admin-equivalent user that the UsageServer process will use for LDAP authentication to eDirectory.

The UsageServer process uses the same LDAP authentication credentials as the Maintenance process.

## What's Next

Your Web Self-Service installation should now be functional. You, your users, and your budget holders can log in to your portal to access the OnDemand Services page using the following URL:

```
http://server_address/nps/servlet/portal
```

where *server\_address* is the IP address or hostname of your portal server.

At this point, the only applications available to users will be the ones that have already been associated with them in eDirectory. These are the same ones that are currently available to them through the ZfD Novell Application Launcher. The applications will be displayed in the Launch Item gadget in either the Applications page or the OnDemand Services page.

To package applications that can be requested by users or provisioned to them by their cost center budget holders, see [Chapter 10, “Creating Application Packages,” on page 75](#).





## Upgrade

The following sections explain how to upgrade a Novell® ZENworks® OnDemand Services™ 2 installation to ZENworks 6 Web Self-Service. If you do not have OnDemand Services 2, see [“Installation” on page 17](#) for information about creating a new Web Self-Service installation.

[Chapter 6, “What’s New in ZENworks 6 Web Self-Service,” on page 51](#)

[Chapter 7, “Software Requirements,” on page 53](#)

[Chapter 8, “Upgrading to Web Self-Service,” on page 57](#)



# 6

## What's New in ZENworks 6 Web Self-Service

Before beginning your upgrade you should be aware of the following differences between Novell® ZENworks® 6 Web Self-Service and ZENworks OnDemand Services™ 2:

- ♦ DeFrame™ is now included in ZENworks 6 Desktop Management (and ZENworks for Desktops 4.0.1) instead of Web Self-Service. For information about upgrading DeFrame, see the [ZENworks for Desktops Upgrade guide \(http://www.novell.com/documentation/lg/zdpr/zdprupgr/data/ahjoi9u.html\)](http://www.novell.com/documentation/lg/zdpr/zdprupgr/data/ahjoi9u.html).
- ♦ Novell Portal Services 1.5 has been replaced by Novell exteNd Director™ 4.1. As with OnDemand Services 2, the version included with Web Self-Service is a runtime version designed to support Web Self-Service only.
- ♦ Novell exteNd Director 4.1 Standard Edition, sold separately, is the full version of the runtime portal included with Web Self-Service. Web Self-Service requires either the runtime version or the full version. If you are using the full version of Novell Portal Services 1.5 and want to retain full portal functionality, you will need to upgrade your NPS 1.5 portal to Novell exteNd Director 4.1 Standard Edition.
- ♦ The functionality of several gadgets has changed:
  - ♦ The Configuration Wizard gadget (OD\_ConfigWizardGadget) is no longer required, unless you need to change the location of the OnDemandService object in eDirectory.
  - ♦ The Edit Account gadget is no longer supported. Users can no longer use this gadget to edit their user account information (name, password, e-mail address, and so forth). This information must be modified through standard eDirectory administration tools such as ConsoleOne®.
  - ♦ The User Administration gadget no longer includes the ability to create, modify, or disable user accounts. These modifications must be done through standard eDirectory administration tools such as ConsoleOne.
- ♦ The Maintenance and UsageServer processes are installed as part of the Desktop Management (ZfD) server software. As a result, the new processes might not be installed to the server where you are running your current processes. However, you can easily copy the processes from one server to another.



# 7

## Software Requirements

The following sections list the requirements for Novell® ZENworks® 6 Web Self-Service:

- ♦ “Web Server Requirements” on page 53
- ♦ “Network Server Requirements” on page 53
- ♦ “Novell eDirectory and ConsoleOne Requirements” on page 54
- ♦ “Desktop Operating System and Web Browser Requirements” on page 55

For examples of supported hardware and software configurations, see “Sample Configurations” on page 22.

### Web Server Requirements

The OnDemand Services portal is created by installing the runtime version of Novell exteNd Director™ 4.1, included with Web Self-Service, along with the OnDemand Services gadgets. The OnDemand Services portal requires one of the Web server environments listed in the table below.

Operating System	Web Server	Web Application Server	JVM
NetWare® 6 with Support Pack 3 or later	Apache 1.3.27	Tomcat 4.1.18	Novell JVM for NetWare 1.4
Windows 2000 with Support Pack 2 or later	IIS 5.0 or Apache 2.0.44	Tomcat 4.1.18	Sun 1.4.1_01 or IBM 1.3

**IMPORTANT:** On a Windows 2000 server, you must run Tomcat as a standard Windows application in the user space rather than as a Windows service in the system space.

If you are currently using the full version of Novell Portal Services (NPS) 1.5 with OnDemand Services 2 and want to retain full portal functionality, you will need to upgrade NPS 1.5 to Novell exteNd Director 4.1 Standard Edition. The Web server to which you install the Standard Edition must meet the requirements listed above. For upgrade instructions, see the [Novell exteNd Director 4.1 Standard Edition documentation \(http://www.novell.com/documentation/lg/nedse41\)](http://www.novell.com/documentation/lg/nedse41).

For an overview of the OnDemand Services portal, see “OnDemand Services Portal” on page 15.

### Network Server Requirements

The Maintenance and UsageServer processes are installed to the Desktop Management (ZfD) server by the Desktop Management Server installation program. If you don’t want to run the processes on the server, you can copy them to any network server that meets the following requirements:

Operating System	JVM
NetWare 5.1 or 6	Novell JVM for NetWare 1.3 or later
Windows NT/2000	Sun 1.3 or later IBM 1.3 or later

For more information about the UsageServer process, see [“UsageServer and UsageClient” on page 16](#).

For more information about the Maintenance process, see [“Maintenance Process” on page 16](#).

## Novell eDirectory and ConsoleOne Requirements

Web Self-Service requires Novell eDirectory 8.6.2 or later. As you prepare your environment for Web Self-Service, please be aware of the following:

- ♦ If the portal server is a NetWare server, the server must contain Read/Write replicas of the partitions where the OnDemand Services objects (OnDemand Admin User object, OnDemandService object, and package objects), Application objects, and User objects reside.
- ♦ If the portal server is a Windows server, eDirectory does not need to be installed on the server. However, the OnDemand Services gadgets must have LDAP access to an eDirectory server that contains Read/Write replicas of the partitions where the OnDemand Services objects (OnDemand Admin User object, OnDemandService object, and package objects), Application objects, and User objects reside.
- ♦ Because the OnDemand Services objects use auxiliary classes that are only supported in NDS<sup>®</sup> 8.x or higher, we recommend that you install Web Self-Service in a tree that is exclusively NDS 8.x or later. If you want to use the OnDemand User Configuration utility to configure ApprovalFlow™ settings (cost center, account ID, and so forth) for multiple users at one time, the User objects must be located on eDirectory 8.5 (or later) replicas.

If you require NDS 6/7 servers in the same tree as Web Self-Service, the NDS 6/7 servers should not hold replicas that contain OnDemand Services objects (OnDemand Admin User object, OnDemandService object, and package objects), Application objects, or User objects for Web Self-Service users.

**NOTE:** NDS 6/7 servers are capable of synchronizing the schema associated with auxiliary classes and can therefore coexist in the same tree. However, objects that have auxiliary classes on them are converted to "Unknown" objects on NDS 6/7 servers. These objects will be displayed as "Unknown" objects and this will prevent you from administering these objects when your primary connection is to a NDS 6/7 server. In addition, Web Self-Service users whose objects appear as "Unknown" will be unable to authenticate to the NDS 6/7 servers.

## eDirectory and LDAP Naming Restrictions

The OnDemand Services gadgets, UsageServer process, and Maintenance process use LDAP to access eDirectory. In all eDirectory contexts that will be accessed by these components, you should make sure that object names (including container and leaf objects) do not include any of the following characters. These characters, when used in LDAP names (for example, cn=jsmith,ou=users,o=novell), will result in invalid syntax.

" (double quote)

+ (plus)

= (equals)	\ (backslash)
< (open angle bracket)	; (semicolon)
> (close angle bracket)	/ (forwardslash)
, (comma)	# (pound sign)

In general, you should ensure that your entire eDirectory tree is LDAP-compliant. To support Web Self-Service, make sure the following eDirectory contexts comply to LDAP naming conventions:

- ♦ User object contexts (for example, cn=jsmith,ou=users,o=novell)
- ♦ Application object contexts (for example, cn=app1,ou=apps,o=novell)
- ♦ OnDemand Services object contexts, including the OnDemand Admin User object, the OnDemandService object, package objects, and item objects (for example, cn=OnDemandService,ou=services,o=novell)
- ♦ Portal object contexts (for example, cn=PortalUser,ou=portal,o=novell)

## ConsoleOne

Web Self-Service requires ConsoleOne® 1.3.5 or later. ConsoleOne 1.3.5 is included on the *ZENworks 6 Companion 1* CD.

## Desktop Operating System and Web Browser Requirements

The following table lists the desktop operating systems and Web browsers that Web Self-Service supports.

Operating System	Web Browsers
Windows 98 SE	Microsoft Internet Explorer 5.5 with SP2
Windows NT 4.0 Workstation	Microsoft Internet Explorer 6.0
Windows 2000 Professional	
Windows XP Professional	

Netscape and other browsers are not supported.

You should make sure that the latest support pack is applied to each of these software components.





# 8

## Upgrading to Web Self-Service

The following sections explain how to upgrade Novell® ZENworks® OnDemand Services™ 2 to ZENworks 6 Web Self-Service.

- ♦ “Upgrading the Portal Software” on page 57
- ♦ “Upgrading the Usage Software (Optional)” on page 66
- ♦ “Upgrading the Maintenance Software (Optional)” on page 69

### Upgrading the Portal Software

Complete the tasks in the following sections to upgrade your portal software:

- ♦ “Upgrading an NPS 1.5 Runtime Portal” on page 57
- ♦ “Upgrading an NPS 1.5 Full Version Portal” on page 59
- ♦ “Recustomizing Gadget Style Sheets” on page 60
- ♦ “Assigning Portal Access” on page 60
- ♦ “Upgrading the Application Association Software” on page 65
- ♦ “Restarting the Portal Server” on page 66

**IMPORTANT:** If necessary, you can maintain user access to your portal during the upgrade process. For information, see [Appendix C, “Maintaining User Access to the Portal During Upgrade,” on page 135](#) before you begin upgrading.

### Upgrading an NPS 1.5 Runtime Portal

The following steps explain how to upgrade an OnDemand Services 2 installation that is using the runtime version of Novell Portal Services 1.5. The runtime version is the version included with OnDemand Services 2. If you installed the OnDemand Services gadgets to a full version of NPS 1.5 rather than the runtime version, skip to [“Upgrading an NPS 1.5 Full Version Portal” on page 59](#).

**1** Prepare the portal server for upgrade:

**1a** Make sure the portal server meets the requirements listed in [“Web Server Requirements” on page 53](#).

**IMPORTANT:** If you plan to install the portal on a NetWare® 6 server and need help installing and configuring Apache 1.3.27 and Tomcat 4.1.18 on the server, refer to [Technical Information Document \(TID\) 10081429 \(<http://support.novell.com/cgi-bin/search/searchtid.cgi?/10081429.htm>\)](#).

**1b** Make sure you know the information for the items listed in the following list. You will be prompted for this information during the upgrade.

- ♦ **LDAP Server Authentication Information:** The portal installation program uses LDAP to access Novell eDirectory™. You need to supply the address (hostname or IP address) and port number of an LDAP server that can provide access to the

eDirectory tree where your current portal resides. In addition, you need to provide the LDAP context and password of the Admin user (or Admin-equivalent user).

- ♦ **Portal Object Distinguished Name and Password Seed:** The portal is represented in eDirectory by a Portal object. The installation program will display a list of all Portal objects in the eDirectory tree. You will need to select the Portal object for your current NPS portal and supply the password seed.

If you do not know the password seed, open the PortalServlet.properties file in the *tomcat33\webapps\nps\WEB-INF* directory. The password seed is specified by the System.Password parameter. The password seed is case sensitive.

- ♦ **OnDemand Admin User Distinguished Name and Password:** The OnDemand Services gadgets authenticate and receive rights to eDirectory through the OnDemand Admin User object. You need to specify the distinguished name and password of your existing OnDemand Admin User object. The object is named ODSAdmin.

**1c** If you have not already done so, stop the Tomcat 3.3 Web Application Server and save a backup of the webapps\nps directory, located in the Tomcat 3.3 root directory, to a location outside of the webapps directory.

**2** To begin the upgrade, copy the nps.war and ZENworks.xar files from the root of the *ZENworks 6 Web Self-Service Program* CD to the Tomcat 4 webapps directory (for example, *sys:\tomcat\4\webapps* or *c:\jakarta-tomcat-4.1.18\webapps*).

**3** Start (or restart) Tomcat 4.

Tomcat extracts the nps.war file to the appropriate locations in the webapps directory.

**4** In a Web browser, enter the following URL to launch the installation/configuration servlet:

`http://server_address/nps/servlet/configure`

where *server\_address* is the server's IP address or hostname.

**5** Click Start to display the License Agreement page.

**6** After reviewing the license agreement, click I Accept the Terms of the License Agreement, then click Next to display the LDAP Directory Server page.

**7** Enter the appropriate LDAP directory authentication information, then click Next to display the Archive to Deploy page.

**8** Select the ZENworks.xar option, then click Next to display the Type of Configuration page.

**9** Select Typical, then click Next to display the Existing Portal Object page.

**10** Select the Portal object for your current NPS portal, then click Next to display the Password Seed page.

If you do not know the password seed, open the PortalServlet.properties file in the *tomcat33\webapps\nps\WEB-INF* directory. The password seed is specified by the System.Password parameter. The password seed is case sensitive.

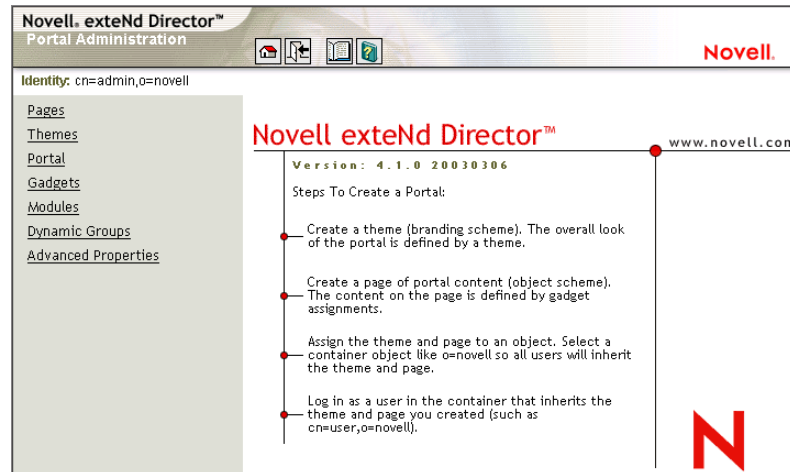
**11** Enter the password seed, then click Next to display the OnDemand Admin User page.

**12** Enter the distinguished name and password of the OnDemand Admin User object (ODSAdmin), then click Next to display the summary page.

**13** Click Configure to begin configuration of the portal.

**14** When configuration is complete, click Continue.

You are logged into the portal using the credentials you provided during the installation. The portal appears with the Portal Administration page displayed.



- 15** Skip to “Recustomizing Gadget Style Sheets” on page 60.

## Upgrading an NPS 1.5 Full Version Portal

The following steps explain how to upgrade the OnDemand Services gadgets in an existing Novell Portal Services 1.5 (full version) portal on a NetWare 6 or Windows 2000 server.

- 1** Prepare the portal server for upgrade:
  - 1a** Make sure the portal server meets the requirements listed in “Web Server Requirements” on page 53.
  - 1b** Upgrade NPS 1.5 to Novell exteNd Director™ 4.1 Standard Edition. For instructions, see the [Novell exteNd Director 4.1 Standard Edition documentation \(http://www.novell.com/documentation/lg/nedse41\)](http://www.novell.com/documentation/lg/nedse41).
  - 1c** Make sure you know the distinguished name and password of your existing OnDemand Admin User object. The object is named ODSAdmin. You will be prompted for this information during the upgrade.
- 2** At a Windows workstation from which you can administer your exteNd Director 4.1 portal, insert the *ZENworks 6 Web Self-Service Program* CD into the CD drive.

The *ZENworks 6 Web Self-Service Program* CD contains the portal module file, *ZENworks.npm*, that includes the OnDemand Services gadgets that you will install to the portal.
- 3** Log in to your portal as an administrator.
- 4** Click Portal Administration > Administer the Portal to display the Portal Administration page.
- 5** Click Modules to display the Modules page.
- 6** Click Install to display the Install a New Module page.
- 7** Fill in the following fields:

**Unique Name for the Module:** Enter a name for the module that is different from any other installed modules (for example, ZENworks).

**Description:** This field is optional. If desired, enter text to describe the module (for example, ZENworks 6 OnDemand Services Gadgets).

**Path to Module File Package:** Click Browse, then browse to and select the ZENworks.npm file on the root of the *ZENworks 6 Web Self-Service Program CD*.

**8** Click Install.

After the OnDemand Services gadgets are installed, an OnDemand Admin User and Password page is displayed. The OnDemand Services gadgets authenticate to eDirectory through this user.

**9** Fill in the following fields:

**OnDemand Admin User:** Using LDAP syntax, enter the distinguished name of your existing OnDemand Admin user (for example, cn=ODSAdmin,ou=services,o=novell,c=us).

**Password:** Enter the ODSAdmin user's password.

**10** Click OK to complete the installation.

**11** Continue with the next section, **Recustomizing Gadget Style Sheets**.

## Recustomizing Gadget Style Sheets

If you modified any of the style sheets for OnDemand Services 2 gadgets and you want to retain those modifications, you will need to carefully compare the old and new style sheets and manually transfer the modifications to the new style sheets.

## Assigning Portal Access

During the portal upgrade process, the four pages listed in the following table are created for the OnDemand Services gadgets.

Page Name	Gadgets	User Assignments
Applications	OD_LaunchItemGadget: Used to launch applications.	All users (including budget holders)
OnDemand Services	OD_LaunchItemGadget: Used to launch applications.  OD_PackageRequestGadget: Used to request an application package.  OD_WorkflowTrackingGadget: Used to check the status of requests and to cancel requests.  OD_ReportGadget: Used to generate reports for subscription and usage-based applications.	All users (including budget holders)
Cost Center Management	OD_UserAdminGadget: Used to assign application packages to users.  OD_CurrentBHGadget: Used to add or remove a cost center's budget holders, and to change a cost center's current (active) budget holder.	budget holders

Page Name	Gadgets	User Assignments
Purchase Approval	OD_ApprovalGadget: Used to approve or deny users' requests for application packages.	budget holders

You must assign the appropriate pages to users and budget holders before they will have access to the pages. The following sections provide instructions:

- ♦ “Assigning Pages to Users” on page 41
- ♦ “Assigning Pages to Budget Holders” on page 43

**NOTE:** You are not required to use the default pages created during installation. They are included to enable you to get started quickly using Web Self-Service. If you want to create your own pages, or add the OnDemand Services gadgets to existing pages in an existing Novell exteNd Director 4.1 Standard Edition portal, see the [Novell exteNd Director 4.1 Standard Edition documentation Web site \(http://www.novell.com/documentation/lg/nedse41\)](http://www.novell.com/documentation/lg/nedse41).

## Assigning Pages to Users

You can give users access to two pages, the Applications page and the OnDemand Services page. Generally, you will not want to assign both pages to users. The following table provides the reasons for using each page.

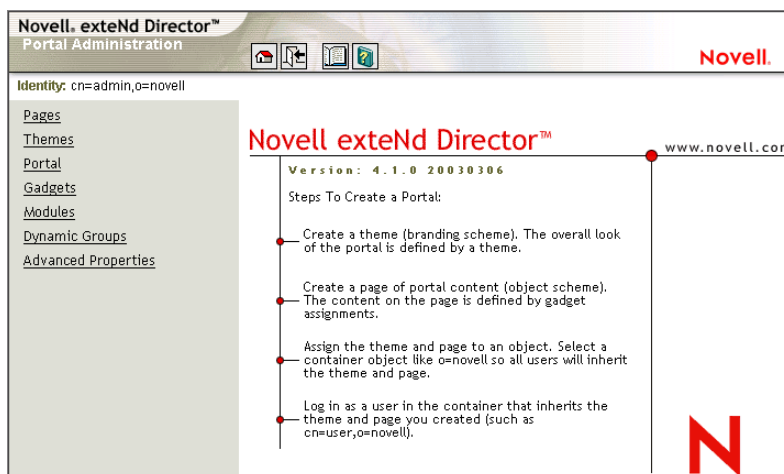
Page Name	Gadgets	Reasons to Use this Page
Applications	OD_LaunchItemGadget	<p>This page includes only the Launch Item gadget, which means that users are limited to launching only administrator-provisioned applications. They cannot request applications.</p> <p>Use this page if you don't want to support user requests for applications, or if you want to provide two pages (Applications and OnDemand Services) from which users can launch applications.</p>
OnDemand Services	OD_LaunchItemGadget OD_PackageRequestGadget OD_ReportGadget OD_WorkflowTrackingGadget	<p>This page includes all the gadgets required for users to both request applications and launch applications.</p> <p>Use this page to support user requests for applications.</p>

All users located within the portal containers (Portal Administration page > Portal > Configuration Properties) are automatically assigned access to the Applications page. If you plan to use the Applications page and all users are located within your portal user containers, you can skip the following steps. If you want to use the OnDemand Services page or need to change (or verify) user assignments for the Applications page, complete the following steps:

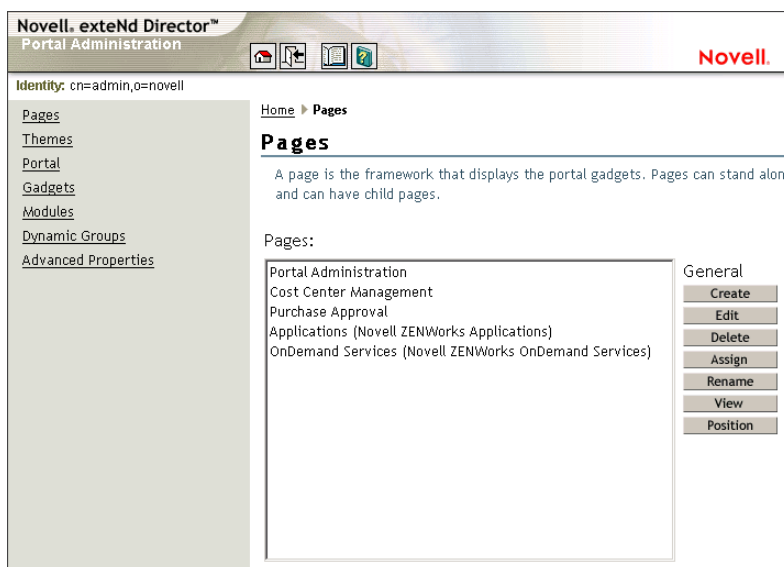
- 1** If you are not already logged in, log in to the portal as an administrator:
  - 1a** Enter `http://server_address/nps`, where *server\_address* is the IP address or hostname of your portal server, to display the login page.
  - 1b** Enter your administrator username and password.

If you are using the OnDemand Services portal, you need to use the same username you supplied when installing the portal.

- 2** Click Administer the Portal to display the Portal Administration page.



- 3** Click Pages to display the Pages page.



- 4** To change or verify the Applications page assignments:

- 4a** Select the Applications page, then click Assign.

By default, the page assignments include all containers defined as your portal containers (Portal Administration page > Portal > Configuration Properties). You can assign the page to additional users by adding individual User objects, Group objects in which the users are members, or container objects where the users reside.

- 4b** To add a user, click Add.

You can use the Search For field to search for a User, Group, or container object, or, if you know the context of the object you want to add, you can enter the context in the

Object Name field. Select the method you want to use, then follow the prompts to add the object.

**4c** To remove a user, user group, or user container, select the object, then click Remove.

**5** To assign the OnDemand Services page to users:

**5a** Select the OnDemand Services page, then click Assign.

**5b** Click Add.

You can assign the page to users by adding individual User objects, Group objects in which the users are members, or container objects where the users reside. By default, no objects are listed.

**5c** If you don't know the context of the object you want to add, select the object type in the Search For field, click OK, then follow the prompts.

or

If you know the context of the object you want to add, type the context in the Object Name field, then click OK.

**6** When you are finished making page assignments, click Close (on the Assign Page to Objects page).

**7** Continue with the next section, [Assigning Pages to Budget Holders](#).

By default, the OnDemand Services gadgets used on the Applications page and OnDemand Services page do not require any configuration changes. If you want to view the configuration settings or change them, see [Appendix A, "Gadget Configuration Settings," on page 127](#).

## Assigning Pages to Budget Holders

You can give budget holders access to two separate pages: the Cost Center Management page and the Purchase Approval page. The following table provides the reasons for using each page.

Page Name	Gadgets	Reasons to Use this Page
Cost Center Management	OD_UserAdminGadget OD_CurrentBHGadget	This page enables budget holders to assign applications to users, add or remove budget holders for the cost center, and change the cost center's current (active) budget holder.
Purchase Approval	OD_ApprovalGadget	This page enables budget holders to approve or deny user requests for applications.  If you've assigned the OnDemand Services page to users and given them the ability to request applications, you need to assign this page to budget holders.

The following sections provide instructions:

- ♦ ["Creating a Dynamic Group for Budget Holders" on page 44](#)
- ♦ ["Assigning Pages to the Budget Holder Dynamic Group" on page 45](#)

## Creating a Dynamic Group for Budget Holders

A budget holder dynamic group lets the portal automatically assign or remove page access when a budget holder is added to or removed from a cost center. A dynamic group is the best way to assign your budget holder pages because it is updated on-the-fly as users are added or removed as budget holders.

- 1 On the Portal Administration page, click Dynamic Groups.
- 2 Click Create to display the Create Dynamic Group page, then fill in the following fields:  
**Name:** Enter a name for the budget holder dynamic group (for example, BudgetHolders).  
**Query Filter String:** Enter the following:

```
( |(securityequals=cn=cost_center1_context)
(securityequals=cn=cost_center2_context)
(securityequals=cn=cost_center3_context)...) )
```

where *cost\_center\_context* is the context (using LDAP syntax) of the Organizational Role objects that are serving as cost centers. For example:

```
( |(securityequals=cn=SalesCC,ou=services,o=novell)
(securityequals=cn=MarketingCC,ou=services,o=novell) )
```

In this example, all users who have been added as budget holders on the SalesCC and MarketingCC cost centers will be included in the dynamic group.

If you still have access to your NPS 1.5 portal, you can copy and paste the query filter string you used in that portal for the budget holder dynamic group.

To test the query filter string, click Test, then follow the prompts.

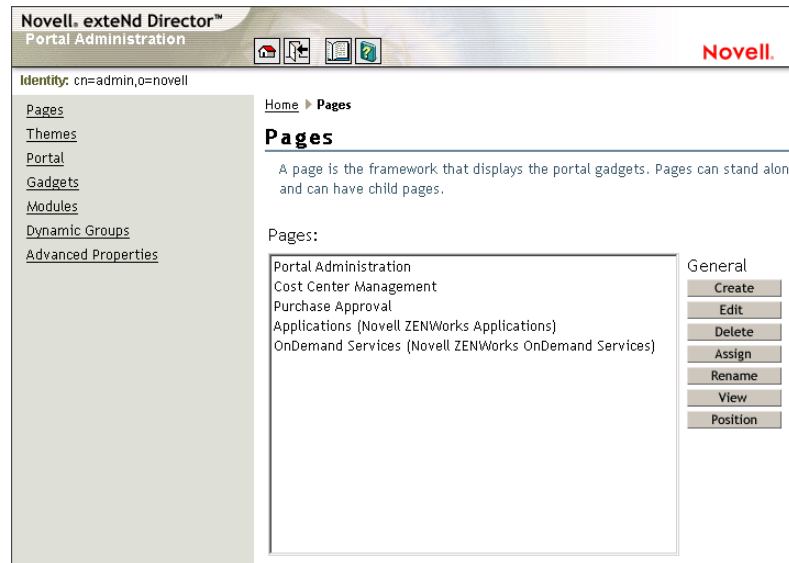
**Container:** By default, the portal will search all portal containers. If you want to limit the search, enter a search container.

- 3 Click Create to create the dynamic group, then click OK to close the Create Dynamic Group page.
- 4 Continue with the next section, [Assigning Pages to the Budget Holder Dynamic Group](#).

## Assigning Pages to the Budget Holder Dynamic Group

- 1 Click Pages to display the Pages page.





- 2** To assign the Cost Center Management page to the budget holder dynamic group:
  - 2a** Select the Cost Center Management page, then click Assign.
  - 2b** Click Add to display the Select Object Type page.
  - 2c** In the Search For field, select Dynamic Groups, then click OK to display the Select Object page.
  - 2d** In the Available Dynamic Groups list, select the budget holder dynamic group, then click Assign.
  - 2e** Click Close to close the Assign Page to Objects page.
- 3** To assign the Purchase Approval page, select the Purchase Approval page, click Assign, then repeat **Step 2b** through **Step 2e**.
- 4** Continue with the next section, **Upgrading the Application Association Software**.

By default, the OnDemand Services gadgets used on the Cost Center Management page and Purchase Approval page do not require any modifications to their configuration. If you want to view the configuration settings or change them, see the Web Self-Service *Administration* guide at the [ZENworks 6 documentation Web site](http://www.novell.com/documentation/lg/zenworks6) (<http://www.novell.com/documentation/lg/zenworks6>).

## Upgrading the Application Association Software

In order for a user to gain access to an application, the OnDemand Services gadgets must create an association in eDirectory between the user's object and the application's object.

On a NetWare server, the OnDemand Services gadgets use the zenapp32.nlm program that is installed with the gadgets. If your portal is on NetWare, you can skip the rest of this section and continue with the next section, **Restarting the Portal Server**.

On a Windows server, the OnDemand Services gadgets use the zennw32.dll installed with the Application Management feature of the Desktop (ZfD) Management Agent. You need to complete the following steps to enable the OnDemand Services gadgets to create application associations:

- 1** Install the Novell Client™ and the Desktop (ZfD) Management Agent to the Windows server.

For information about installing the Management Agent and the Novell Client, see the *ZfD Installation* guide at the [ZENworks for Desktops 4.0.1 documentation Web site \(http://www.novell.com/documentation/lg/zdpr\)](http://www.novell.com/documentation/lg/zdpr).

The OnDemand Services 2 gadgets used the zenapp32.dll included with ZfD 3.2. In ZfD 4, zennw32.dll replaced zenapp32.dll. After you upgrade your portal, the upgraded OnDemand Services gadgets can continue to use the zenapp32.dll for application associations, but some of the new functionality included in ZfD 4, such as application chaining, will not be supported by the gadgets until you install the Desktop (ZfD) Management Agent that includes the zennw32.dll file.

- 2 At the Windows server, log in to eDirectory through a user account that has rights to all containers where Application objects and User objects reside.

By default, your OnDemand Admin User object (ODSAdmin) has rights to all containers where Application objects and User objects reside. You can log in at the Windows server as ODSAdmin or any other user with sufficient rights (Admin, for example).

## Restarting the Portal Server

If you have not already restarted the portal server at some point after installing the portal software, you need to do so to ensure that all portal pages will be displayed correctly to users.

## Upgrading the Usage Software (Optional)

The following information applies only if you set up OnDemand Services 2 to track usage of applications. If you are not tracking application usage, skip to [“Upgrading the Maintenance Software \(Optional\)” on page 69](#).

Upgrading the usage software (UsageServer process and UsageClient), as well as the maintenance software (Maintenance process), is optional. The usage and maintenance software has not changed significantly since OnDemand Services 2 Support Pack 1. In fact, the only differences are that one UsageServer exception message has been changed and that the usage and maintenance files are now installed to the \zenworks\ondemand directory (rather than the \ondemand directory) by the ZfD server software installation program.

If you want to upgrade your usage software, you need to upgrade both the UsageServer software and the UsageClient software. The following sections provide instructions:

- ♦ [“Upgrading the UsageServer Software” on page 66](#)
- ♦ [“Upgrading the UsageClient Software” on page 68](#)

## Upgrading the UsageServer Software

The UsageServer process, along with the Maintenance process, is installed during the ZfD Server software installation. To complete the upgrade of the UsageServer and Maintenance processes, you need to do the following:

- 1 (Optional) If you do not want to run the UsageServer and Maintenance processes on the ZfD server, manually copy the \zenworks\ondemand directory to another server that meets the requirements listed in [“Network Server Requirements” on page 53](#).

For example, if you are currently running the processes on a non-ZfD server, you can manually copy the UsageServer and Maintenance files from the ZfD server to the current server, or you can choose to run the new UsageServer and Maintenance processes on the ZfD server.

## 2 Configure the new UsageServer and Maintenance processes:

### 2a Modify the \zenworks\ondemand\CommerceMaintenance.cfg file to include the settings provided in the old CommerceMaintenance.cfg file.

Because, the UsageServer files are installed to the \zenworks\ondemand directory rather than the \ondemand directory, a new CommerceMaintenance.cfg file is created. You should compare the new file to your old file to make sure the new file contains the correct settings. In particular, the following settings will need to be modified:

**ldap.host:** Specifies the IP address or DNS host name of the LDAP server used to authenticate to eDirectory. The default IP address, 127.0.0.1, provides authentication through the local LDAP server.

**ldap.loginName:** Specifies the user through which authentication occurs. Use LDAP syntax to specify the user's distinguished name (for example, cn=admin,o=novell).

If the user's distinguished name includes extended (multibyte) characters and the Maintenance process is running on a NetWare server, you need to run `commerceserver.cfg` through the Java utility `native2ascii`. At the server console, enter the following:

```
native2ascii \zenworks\ondemand\commerceserver.cfg
```

This needs to be done any time you change the user name.

**baseContext:** Specifies the container that the Maintenance process will use as its root container. The Maintenance process only processes Purchase and Usage objects located in the root container and its subcontainers. Use LDAP syntax to specify the root container distinguished name (for example, ou=apps,ou=services,o=novell).

### 2b On a NetWare server, include the \zenworks\ondemand\bin directory in the server's search path.

The ZfD server software installation program adds the search path to the `autoexec.ncf` file. However, it is commented out. To make it active, you must remove the `#` character at the beginning of the line.

### 2c (Optional) On a NetWare server, if you want the UsageServer or Maintenance process to load on server startup, uncomment the UsageServer.ncf and CommerceMaintenance.ncf commands in the autoexec.ncf file.

**NOTE:** Both the UsageServer process and Maintenance process require an LDAP authentication password (for the user specified by the `ldap.loginName` parameter in the `CommerceMaintenance.cfg` file) at startup. Regardless of whether you start the processes through the `autoexec.ncf` file or from the system console, you will need to supply the password.

## 3 Remove the old UsageServer and Maintenance processes:

### 3a Stop the UsageServer and Maintenance processes:

- ♦ On NetWare, enter **java -killUsageServer** and **java -killCommerceMaintenance**
- ♦ On Windows, close the command windows in which the processes are running.

### 3b Delete the \ondemand directory from the server.

### 3c On NetWare, if the following command is in the autoexec.ncf file, remove it:

```
SEARCH ADD SYS:\OnDemand\bin
```

## Upgrading the UsageClient Software

Usage tracking of ZfD DeFrame thin-client applications requires the UsageClient to be on each DeFrame terminal server while usage tracking of ZfD desktop applications requires the UsageClient to be on each user's workstations. The following sections provide instructions:

- ♦ “Upgrading the UsageClient Software on ZfD DeFrame Terminal Servers” on page 68
- ♦ “Upgrading the UsageClient Software on User Workstations” on page 68

### Upgrading the UsageClient Software on ZfD DeFrame Terminal Servers

You must install the UsageClient to each ZfD DeFrame terminal server. To do so:

**1** Launch setup.exe from one of the following locations:

- ♦ The \WorkstationSetup directory on the *ZENworks 6 Web Self-Service Program* CD.
- ♦ The \zenworks\ondemand\setup directory on a ZfD server.
- ♦ The \public\zenworks\thinclnt\setup directory on a ZfD server.

**2** Follow the prompts.

### Upgrading the UsageClient Software on User Workstations

For usage tracking of ZfD desktop applications, the UsageClient must be installed on each user's workstation. This happens automatically the first time a usage-based desktop application is launched from the OnDemand Services Launch Item gadget.

If you have locked down workstations so that users don't have the file system rights needed to install applications, the OnDemand Services Launch Item gadget will be unable to install the UsageClient. In this case, you will need to use one of the following installation methods:

- ♦ Log in with administrator rights at each workstation and launch a usage-based desktop application from the Launch Item gadget.
- ♦ Log in with administrator rights at each workstation and run the setup.exe program from one of the following locations:
  - ♦ The \WorkstationSetup directory on the *ZENworks 6 Web Self-Service Program* CD.
  - ♦ The \zenworks\ondemand\setup directory on a ZfD server.
  - ♦ The \public\zenworks\thinclnt\setup directory on a ZfD server.
- ♦ Use Novell Application Launcher and the predefined OnDemand Client Application object to distribute the setup.exe program. Instructions are provided below.

#### Using Novell Application Launcher and the OnDemand Client Application Object

During installation of the ZfD Server software to a network server, the ZfD Server installation program created an OnDemand Client Application object in the same eDirectory container as the ZfD Server object. This Application object launches a setup program that installs the following software components required for ZfD DeFrame and Web Self-Service:

- ♦ **DeFrame Client:** Required to launch ZfD DeFrame thin-client applications from the OnDemand Services Launch Item gadget and Novell Application Launcher.
- ♦ **ICA Client:** Required to launch ZfD DeFrame thin-client applications in an ICA client session.

- ♦ **RDP Client:** Required to launch ZfD DeFrame thin-client applications in an RDP client session.
- ♦ **NAL Plug-in:** Required to launch ZfD desktop applications from the OnDemand Services Launch Item gadget. The UsageClient files are also included with the plug-in.

If, during installation of ZfD DeFrame, you already ran the setup program on user workstations to install the DeFrame, ICA, and RDP clients, the NAL plug-in and UsageClient files were also installed. You do not need to install them again. You can skip the remainder of this section and continue with **“Upgrading the Maintenance Software (Optional)” on page 69**.

The OnDemand Client Application object is configured to run the setup program one time. The setup program requires user interaction to confirm installation of the various clients; users will not need to provide configuration information such as installation paths. You might want to manually run the setup program one time to see how users will need to interact.

The following are examples of how you can use the OnDemand Client Application object:

- ♦ You can directly associate the OnDemand Client Application object with users, in which case they will need to launch the Application object to run the setup program. For information about associating Application objects to users, see [Distributing Applications to Users and Workstations \(http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/a7dup3a.html#a7dup3a\)](http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/a7dup3a.html#a7dup3a) in Application Management in the ZENworks for Desktops *Administration* guide.
- ♦ You can assign the OnDemand Client Application object as an application dependency for usage-based desktop Application objects (or any type of Application object if you want to ensure that the files will be installed). By making the OnDemand Client Application object a dependency for usage-based desktop applications, the setup program will run the first time a user launches one of the applications. For information about setting up application dependencies, see [Setting Up Application Dependencies \(http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/af1ms6k.html#af1ms6k\)](http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/af1ms6k.html#af1ms6k) in Application Management in the ZENworks for Desktop *Administration* guide.

## Upgrading the Maintenance Software (Optional)

If you followed the instructions in **“Upgrading the UsageServer Software” on page 66**, you should have upgraded the Maintenance process at the same time you upgraded the UsageServer process.

If you do not use usage tracking and therefore skipped the Upgrading the UsageServer Software section, follow the instructions in that section to upgrade the Maintenance process.



# IV

## Administration

The following sections provide information and instructions for completing various administration tasks for Novell® ZENworks® 6 Web Self-Service.

- Chapter 9, “Administration Basics,” on page 73
- Chapter 10, “Creating Application Packages,” on page 75
- Chapter 11, “Tracking Application Usage,” on page 83
- Chapter 12, “Limiting Package Availability,” on page 87
- Chapter 13, “Organizing Packages and Applications,” on page 91
- Chapter 14, “Securing Web Applications With Novell iChain,” on page 93
- Chapter 15, “Customizing Notification Messages,” on page 99
- Chapter 16, “Creating Cost Centers,” on page 101
- Chapter 17, “Billing for Purchases,” on page 107
- Chapter 18, “Changing the Status of a Purchase,” on page 109
- Chapter 19, “Managing User Workstation Software,” on page 111
- Chapter 20, “Managing the Maintenance and UsageServer Processes,” on page 115





# 9

## Administration Basics

The following sections provide basic information that will help as you administer Novell® ZENworks® Web Self-Service.

- ♦ “OnDemand Services Gadgets” on page 73
- ♦ “Portal Administration” on page 73
- ♦ “Web Self-Service Snap-Ins for ConsoleOne” on page 73
- ♦ “Novell eDirectory Management” on page 73

### OnDemand Services Gadgets

Much of the Web Self-Service functionality is enabled through the OnDemand Services gadgets. When appropriate, the administration tasks in this documentation cover the enabling and configuring of the gadgets associated with the tasks. However, not all configuration settings are discussed in the administration tasks. You should review [Appendix A, “Gadget Configuration Settings,” on page 127](#) so that you are familiar with the functionality provided by the gadget configuration settings.

### Portal Administration

This guide provides administration information specific to the tasks required to configure and maintain the OnDemand Services gadgets. For detailed portal administration tasks and information, see the [Novell exteNd Director™ 4.1 Standard Edition documentation Web site \(http://www.novell.com/documentation/lg/nedse41\)](http://www.novell.com/documentation/lg/nedse41).

### Web Self-Service Snap-Ins for ConsoleOne

The Web Self-Service snap-ins for ConsoleOne® are installed by the ZENworks for Desktops Server installation program. You can run ConsoleOne from any ZfD server to administer the Web Self-Service eDirectory™ objects (Package objects, Purchase objects, Usage objects, and so forth). You can also run ConsoleOne from a Windows workstation by copying the ConsoleOne directory from a ZfD Server to the Windows workstation.

### Novell eDirectory Management

The Web Self-Service components (OnDemand Service gadgets, Maintenance process, UsageServer process, and so forth) perform tasks in Novell eDirectory to manage applications, packages, and users’ access to packages. In general, if a Web Self-Service component configures a setting in eDirectory, you should let Web Self-Service manage that setting. Listed below are some things you shouldn’t do in eDirectory because they adversely affect the way Web Self-Service operates.

**Disassociating Desktop and Thin-Client Applications:** When a user receives access to a package that includes desktop or thin-client applications, Web Self-Service associates the user with the applications through the Application object's Association page. When the user's access expires, Web Self-Service disassociates the user from the applications.

You should not manually disassociate a user from a packaged desktop or thin-client application. Doing so does not set the Expired flag on the Purchase object, so the application continues to appear in the user's Launch Item gadget and also in the budget holder's User Administration gadget. The user can continue to launch the application at no charge (if a cost is associated with the application's package) and the budget holder doesn't know that the user has been disassociated with the application.

For more information about application packages, see [Chapter 10, "Creating Application Packages,"](#) on page 75.

**Removing Users from iChain ACL Rules Apply To Lists:** When a user receives access to a package that includes a Web application that is secured through Novell iChain™, Web Self-Service adds the user to the Apply To list for the Web application's ACL rule. You should not manually remove a user from the Apply To list. Doing so causes the user to be unable to launch the Web application, but the Web application still shows up in the user's Launch Item gadget.

For more information about Web Self-Service integration with iChain, see [Chapter 14, "Securing Web Applications With Novell iChain,"](#) on page 93.

**Removing Purchase Objects:** When a user receives access to a package, Web Self-Service creates a Purchase object for the user under the Package object. The Launch Item gadget uses this object to know which applications to display to the user. After the user's access has expired, Web Self-Service will mark the Expired flag on the Purchase object and revoke the user's rights to the package's applications. At that point, the package's applications will no longer appear in the Launch Item gadget.

You should not manually remove the Purchase object. If you want to remove an expired Purchase object, make sure the Expired and Billed flags are marked. The Maintenance process will then remove the object (and any Usage objects associated with the Purchase object) the next time it processes the eDirectory tree.

For more information about Purchase objects, see [Chapter 18, "Changing the Status of a Purchase,"](#) on page 109. For more information about the Maintenance process, see [Chapter 20, "Managing the Maintenance and UsageServer Processes,"](#) on page 115.

# 10

## Creating Application Packages

In order for users to request applications or budget holders to assign applications, you must first include the applications in an application package.

An application package can consist of a single application or a suite of applications. For example, you could create a package that consists of Microsoft Word. Or, you could create a package that includes all the Microsoft Office applications (Word, Excel, and so forth). Users who receive access to a package can use all applications included in the package.

A package can include purchase and usage charges, or you can make the package available for free.

Refer to the following sections for information about packaging applications to make them available to users:

- ♦ “Preparing the Package’s Applications” on page 75
- ♦ “Creating the Package Object” on page 78
- ♦ “Adding Applications to the Package” on page 79
- ♦ “Establishing the Package’s Cost” on page 80

**IMPORTANT:** If a user purchases a package, but the package contents are not successfully delivered to the user, the user will still be charged for the purchase. Users will need to contact you to request that the purchase be removed.

### Preparing the Package’s Applications

Before you create the package, you need to make sure that the package’s applications are ready. Refer to the following sections:

- ♦ “Desktop Applications” on page 75
- ♦ “Thin-Client Applications” on page 76
- ♦ “Web Applications” on page 76

### Desktop Applications

Desktop applications are applications that run on the user’s desktop. The files can reside on the user’s workstation or on a network server.

Distributing Applications to Users and Workstations (<http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/a7dup3a.html>) in the Application Management part of the ZENworks® for Desktops *Administration* guide provides information about creating Application objects for desktop applications. If necessary, follow the instructions in that section to create the Application object, using the following guidelines:

- ♦ Use a UNC path (for example, \\server\_hostname\vol1\snapshot\appname) in any locations where you are prompted for a path. Use the server's hostname in all locations where the server name is required.
- ♦ Do not use any of the following characters in the Application object's name:

" (double quote)	+ (plus)
= (equals)	\ (backslash)
< (open angle bracket)	; (semicolon)
> (close angle bracket)	/ (forwardslash)
, (comma)	# (pound sign)

- ♦ Do not use the Application object's Association page to associate the application with users. Web Self-Service will create this association when a user purchases a package that includes the application.
- ♦ If you plan to track the application's usage, turn on the Enable OnDemand Services to Track Usage of This Application option. This option is located on the Application object's OnDemand tab. In addition, refer to the **Chapter 11, "Tracking Application Usage," on page 83** to ensure that usage tracking is properly configured.
- ♦ If the application's usage will be tracked (as discussed in the previous bulleted item) and the application's executable is a stub program that simply launches the real executable (for example, write.exe launches wordpad.exe), make sure that you've entered the real executable filename in the Application object's Monitor Module Name field (Run Options tab > Environment page). Otherwise, Web Self-Service will only track usage of the stub program, which will result in a very brief, inaccurate usage tracking record.

## Thin-Client Applications

Thin-client applications are applications that users access from a DeFrame™ terminal server through a terminal server client (thin-client) session. Before you can add a thin-client application to a package, you need to create the Application object. For instructions, see the ZENworks for Desktops *DeFrame Administration* guide at the [ZENworks for Desktops 4.0.1 documentation Web site \(http://www.novell.com/documentation/lg/zdpr\)](http://www.novell.com/documentation/lg/zdpr).

## Web Applications

To include a Web application in a package, you need to 1) make sure the application is located on a Web server that users can access and 2) create a Commerce Item object to define the application in Novell® eDirectory™.

You can also include Web content, such as document, video, or audio files, in addition to Web applications. The same process is required if you want to include Web content.

To create a Commerce Item object in eDirectory:

- 1 In ConsoleOne®, right-click the container where you want to create the object > click New > click Object.

You should create the Commerce Item object in the company container (as defined in the OnDemandService object) or one of its subcontainers. This ensures that OnDemand Services has sufficient rights to the object. If you create it outside of the company container or its

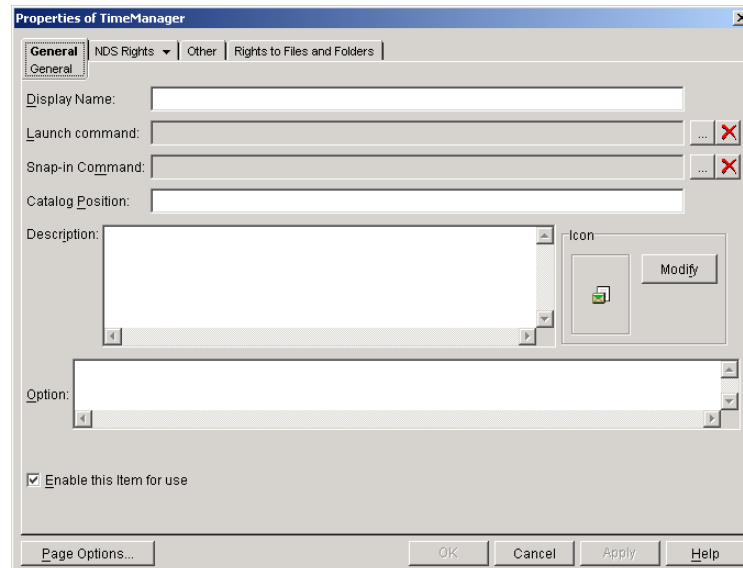
subcontainers, you will need to make the OnDemand Admin User object a trustee of the Commerce Item object. For details about the required trustee rights, see [Appendix B, “OnDemand Admin User Rights,”](#) on page 133.

- 2** In the New Object list, select the commerceItem object, then click OK.
- 3** Provide the new item with a name.

The name you supply will be used for the object’s eDirectory name and for display in the OnDemand Services gadgets if you do not supply a Display Name later. Do not use any of the following characters in the name.

" (double quote)	+ (plus)
= (equals)	\ (backslash)
< (open angle bracket)	; (semicolon)
> (close angle bracket)	/ (forwardslash)
, (comma)	

- 4** Select the Define Additional Properties box, then click OK.



- 5** On the General tab, fill in the following properties:

**Display Name:** Enter the name of the item as you want it displayed to users. If you don’t enter a display name, the Commerce Item object’s name is used.

**Launch Command:** Click the Define Command button (⋮) to display the Launch Command dialog box. Enter the URL for the Web application or content, then click OK.

**Snapiin Command:** This field applies only if you are using Novell iChain® to secure access to the Web application. If so, click the Define Command button (⋮) to display the Snapiin Command dialog box. Browse for and select the iChain ACL rule that will provide access to the secured Web application, then click OK. For more information about iChain and iChain ACL rules, see [Chapter 14, “Securing Web Applications With Novell iChain,”](#) on page 93.

**Catalog Position:** You can use this field to define a folder structure to display the item in the Launch Item gadget. The syntax uses forward slashes, as in the following example:

/Software/Wordprocessors

**Description:** Enter a description to be displayed to the user.

**Icon:** If you want to change the display icon, click Modify. You can select an icon from an executable file or .ico file.

**Option:** This field lets you specify a URL to a Web site that describes or promotes the item. Use the following syntax:

```
<attribute name='brandingURL' value='URL' />
```

Replace *URL* with the appropriate URL. Make sure to include the protocol in the URL (for example, <http://www.novell.com>).

**Enable This Item for Use:** Select this option to make the item available to users. If you deselect this option, the item will not be available, even if a user has already purchased the package.

- 6** Click OK to save the information.

## Creating the Package Object

The Package object is used to add package items and determine the package price. You can create Package objects in the catalog root or any container below the catalog root.

To create a Package object in eDirectory:

- 1** In ConsoleOne, right-click the container where you want to create the Package object > click New > click Object.

You can create the Package object anywhere in the catalog root container (as defined on the OnDemandService object) or its subcontainers. The catalog root should have been defined during installation. If it was not, see [“Configuring the Catalog Root and ApprovalFlow E-Mail Settings” on page 39](#).

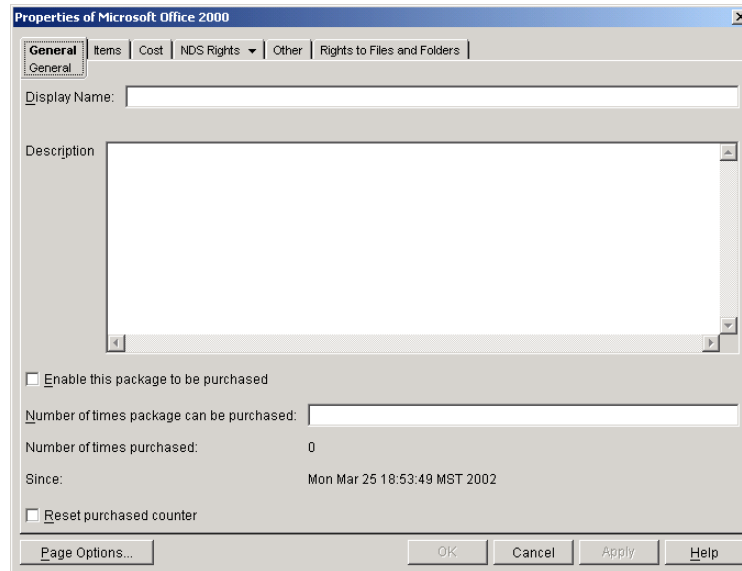
For information about how the organization of packages under the catalog root impacts users, see [“Organizing Packages and Applications” on page 91](#).

- 2** In the New Object list, select the commercePackage object, then click OK.
- 3** Provide the new package with a name.

The name you supply will be used for the object’s eDirectory name and for display in the OnDemand Services gadgets if you do not supply a Display Name later. Do not use any of the following characters in the name.

" (double quote)	+ (plus)
= (equals)	\ (backslash)
< (open angle bracket)	; (semicolon)
> (close angle bracket)	/ (forwardslash)
, (comma)	

- 4** Select the Define Additional Properties box, then click OK.



- 5 On the General tab, fill in the following properties:

**Display Name:** Enter the name of the package as you want it displayed to customers. If you don't enter a display name, the Package object's name is used.

**Description:** Enter a description to be displayed to the user.

**Enable This Package to Be Purchased:** Turn on this option when you are ready to allow users to purchase the package.

**Number of Times Package Can Be Purchased:** If you want to restrict the number of times the package can be purchased, enter the appropriate number. Leave the field empty to enable an unlimited number of purchases.

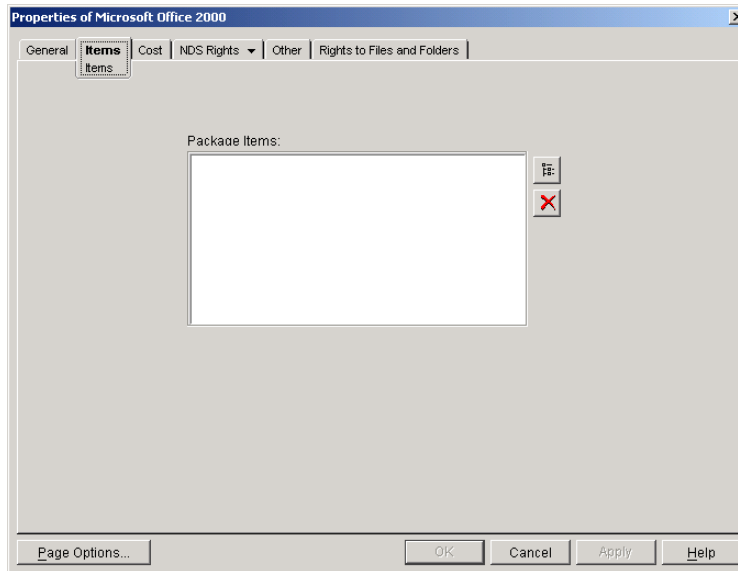
**Reset Purchased Counter:** This option applies only if you've set a limit on the number of times the package can be purchased.

Click this option, then click Apply to reset the purchase counter. This is useful if the purchase limit has been reached and you want to enable the package to be purchased again, or if you want to start tracking the number of purchases from the current day and time.

- 6 Leave the Package object's dialog box open, then continue with the next section, [Adding Applications to the Package](#).

## Adding Applications to the Package

- 1 Click the Items tab.



- 2 Click the browse button next to the Package Items list, then browse to and select the Application object or Commerce Item object to add to the package.

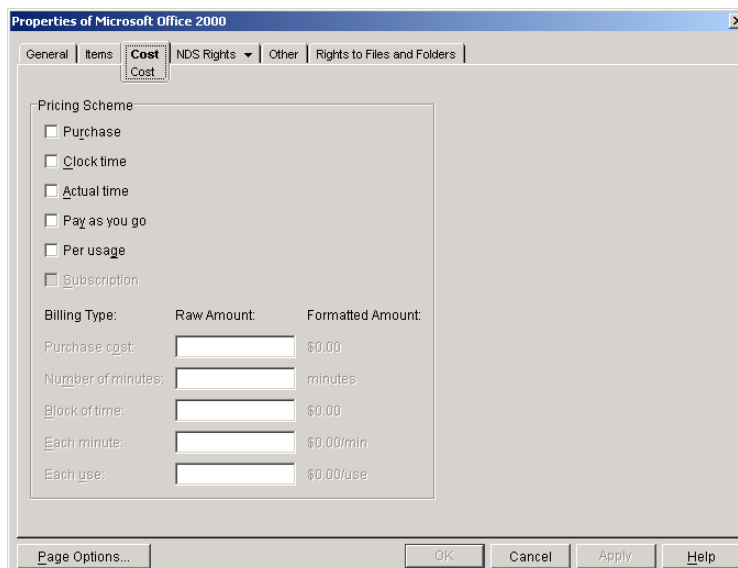
**IMPORTANT:** If you add the same application to two different packages and a user purchases both packages, only one application icon will be displayed in the Launch Item gadget. In addition, any charges associated with using the application will accrue to the most recent purchase. After that purchase expires, the charges will revert to the previous purchase.

- 3 Leave the Package object's dialog box open, then continue with the next section, [Establishing the Package's Cost](#).

## Establishing the Package's Cost

By default, no cost is associated with the package. Complete the following steps only if you want to apply purchase or usage charges to the package.

- 1 Click the Cost tab.





**2** Select one or more of the following pricing schemes:

- ♦ **Purchase:** The user pays a one-time fee for the package. The user can use the package whenever necessary.
- ♦ **Clock Time:** The user purchases a set amount of time to use the package. The time begins when the user purchases the package, runs continuously, and expires after the set amount of time.
- ♦ **Actual Time:** The user purchases a set amount of time to use the package. Only the actual time spent using the item is recorded. If the user purchases 5 hours, he or she can use 1 hour today, 3 hours tomorrow, and 1 hour next week.
- ♦ **Pay-As-You-Go:** The user pays by the minute to use the package. The time begins when the user launches a package item and ends when the item is exited.
- ♦ **Per Usage:** The user pays each time he or she uses the package.
- ♦ **Subscription:** The user pays a set amount for a subscription to the package. You determine the subscription period (7 days, 14 days, 30 days, etc.). At the end of the subscription period, the user's subscription is automatically renewed. By default, users can cancel a subscription. If you want to disable a user's ability to cancel a subscription, see [“Assigning Users” on page 37](#).

You can use a combination of these pricing schemes. For example, you could have a one-time purchase fee (Purchase pricing scheme) for a package, charge a fee for every time one of the package's applications is used (Per Usage pricing scheme), and also charge a per-minute fee for using the application. (Pay-As-You-Go pricing scheme). For information about which pricing schemes can be used together, click the Help button on the Cost page.

The Pay-As-You-Go and Actual Time schemes should not be used with Web applications because the UsageServer process is unable to track the actual time Web applications are in use. For details, see [“Usage Tracking Support for Each Application Type” on page 83](#).

**IMPORTANT:** All schemes, other than the Purchase scheme, require Web Self-Service to track usage of the package's items. If you select a usage-based scheme, you need to make sure that usage tracking is enabled and configured properly. For information, see [Chapter 11, “Tracking Application Usage,” on page 83](#).

**3** Fill in the cost information for the selected schemes.

**4** Click OK to save the information.



# 11

## Tracking Application Usage

Novell® ZENworks® Web Self-Service enables you to track the usage of applications. Usage tracking is required if you want to implement package pricing schemes that are based on the number of times the package is used or the number of minutes it is used. If you don't want to implement usage-based pricing schemes for your packages, usage information can still help you decide if you need to increase or decrease the resources (hardware, Help Desk personnel, etc.) supporting applications.

- ♦ [“Usage Tracking Support for Each Application Type” on page 83](#)
- ♦ [“Enabling Usage Tracking of an Application” on page 84](#)
- ♦ [“Ensuring Accurate Usage Tracking for Thin-Client Applications” on page 84](#)
- ♦ [“Starting the UsageServer and Maintenance Processes” on page 85](#)

### Usage Tracking Support for Each Application Type

Web Self-Service provides full usage tracking support for thin-client and desktop applications. You can track both the number of times (per-use) and the number of minutes (per-minute) a thin-client or desktop application is used.

Web Self-Service provides partial usage tracking support for Web applications. You can track per-use but not per-minute usage.

Because per-minute usage is not supported for Web applications, you should not include Web applications in any package whose usage-based pricing scheme is reliant on tracking time that the application is open. The Actual Time and Pay-As-You-Go pricing schemes fall into this category. The following table summarizes the application types and the package pricing schemes you can use with packages that contain those application types.

	Web Application	Thin-Client Application	Desktop Application
Clock Time	Yes	Yes	Yes
Actual Time	No*	Yes	Yes
Pay-As-You-Go	No*	Yes	Yes
Per Usage	Yes	Yes	Yes
Subscription	Yes	Yes	Yes

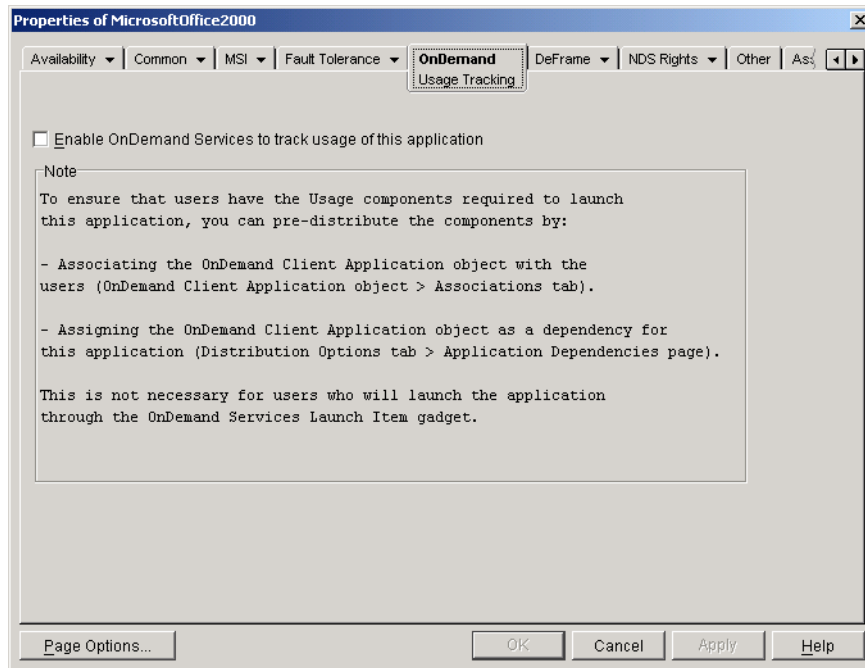
\* You can actually include Web applications in packages that use these pricing schemes. With Actual Time, however, using the package's Web applications will not increment the time accrued; only thin-client and desktop applications included in the package will increment the time. With

Pay-As-You-Go, use of the package's Web applications will be free; only thin-client and desktop applications included in the package will be charged a per-minute usage fee.

## Enabling Usage Tracking of an Application

By default, Web applications are enabled for usage tracking, but desktop and thin-client applications are not. To enable a desktop or thin-client application:

- 1** In ConsoleOne<sup>®</sup>, right-click the Application object.
- 2** Click the OnDemand tab to display the Usage Tracking page.



- 3** Click the Enable OnDemand Services to Track Usage of This Application check box to turn on the option.
- 4** If you want OnDemand Services to terminate the application if its usage expires while the user is running it, complete the following substeps. Otherwise, the user will be able to use the application until he or she exits.
  - 4a** Click the Availability tab, then click Termination.
  - 4b** Select the desired termination behavior, then click OK. .

For a description of each behavior, click the Help button on the Termination page

- 5** Click OK to save your changes.

## Ensuring Accurate Usage Tracking for Thin-Client Applications

To properly terminate usage tracking of a thin-client application, users need to use the application's Exit feature (for example, File menu > Exit) rather than clicking the Close button in the upper-right corner of the session window. Using Exit correctly exits the application, logs the user out of the session, and terminates usage tracking.

If, rather than using the application's Exit feature, the user closes the terminal server session by clicking the Close button in the upper-right corner of the session window, the user is disconnected from the session but not logged out. The session remains open on the terminal server, and the user can reconnect to the same session by launching the application again. This means that the application was never really exited, so usage tracking continues.

To ensure that disconnected sessions are closed out, you can use either of the following methods:

- ♦ Configure each terminal server to end disconnected sessions after a certain period of time. For instructions, see your terminal server documentation.
- ♦ Enable a ZfD Terminal Server policy (User Policy Package > Win2000 Terminal Server Policies > Windows Terminal Server Policy > Connection page) and apply the User policy package to the appropriate users. The Windows Terminal Server policy should configure a timeout interval for disconnected sessions (Disconnection Timeout setting) and reset the connection on timed-out sessions (Reset Broken or Timed-Out Sessions setting). For more information, see the [ZENworks for Desktops 4.0.1 documentation \(http://www.novell.com/documentation/lg/zdpr\)](http://www.novell.com/documentation/lg/zdpr).

## Starting the UsageServer and Maintenance Processes

For usage tracking to occur, you need to make sure the UsageServer process is running. If the UsageServer process is not installed, or for information about configuring the process, refer to [“Managing the Maintenance and UsageServer Processes” on page 115](#).

In addition, the Maintenance process also needs to be running. If the UsageServer becomes unavailable, the Maintenance process closes the usage records for any applications that were being tracked. Even if the users continue to use the applications, they are only charged for the time up to the point that the UsageServer became unavailable. This ensures that they will not be charged for time they did not use.

- ♦ [“Starting the UsageServer Process” on page 85](#)
- ♦ [“Starting the Maintenance Process” on page 86](#)

## Starting the UsageServer Process

To start the UsageServer process:

- 1** At the NetWare® server console, enter **usageserver.ncf**.

or

At the Windows server command prompt, change to the directory where the UsageServer process was installed (by default, \zenworks\ondemand\bin), then enter **usageserver.bat**.

- 2** Enter the password for the Admin or Admin-equivalent user that the Usage Server will use to authenticate to Novell eDirectory™ through the LDAP server.

If you can't remember which user is being used for authentication, the user's context is stored in the `commerceserver.cfg` file, located in the \ondemand directory.

You can install additional UsageServer processes or change the default configuration for this process. For information, see [Chapter 20, “Managing the Maintenance and UsageServer Processes,” on page 115](#).

## Starting the Maintenance Process

- 1 At the NetWare server console, enter **commercemaintenance.ncf**.

or

At the Windows server command prompt, change to the directory where the Maintenance process was installed (by default, \zenworks\ondemand\bin), then enter

**commercemaintenance.bat**.

- 2 Enter the password for the Admin or Admin-equivalent user that the Maintenance process will use to authenticate to eDirectory through the LDAP server.

If you can't remember which user is being used for authentication, the user's context is stored in the `commerceserver.cfg` file, located in the \ondemand directory.

You can install additional Maintenance processes or change the default configuration for this process. For information, see [Chapter 20, "Managing the Maintenance and UsageServer Processes,"](#) on page 115.

# 12 Limiting Package Availability

By default, Novell® ZENworks® Web Self-Service is configured to allow all packages to be available to all users. You can, however, configure Web Self-Service so that packages are available only to the users who have been granted rights specifically to the packages.

The OnDemand Admin User object is a trustee of the catalog root container. The OnDemand Services Package Request gadget is configured to access the catalog as the OnDemand Admin user, which means that it has rights to see all packages contained within the catalog. To limit package availability, you configure the Package Request gadget to receive its catalog rights through the logged-in user rather than the OnDemand Admin user. The user will then see only the packages that you specifically give him or her rights to.

The following sections provide instructions for completing the tasks discussed above:

- ♦ “Configuring the Package Request Gadget to Limit Package Availability” on page 87
- ♦ “Giving Users Trustee Rights to Packages” on page 88

## Configuring the Package Request Gadget to Limit Package Availability

The Package Request gadget’s Limit Package Availability configuration setting determines whether the gadget displays all packages or only the packages the user has been specifically granted rights to.

To configure the gadget to limit availability:

- 1** Log in to the portal as a portal administrator.
- 2** Click Administer the Portal.
- 3** Click Gadgets.
- 4** Select the OD\_PackageRequestGadget, then click Edit.
- 5** Click Configuration.
- 6** Change the Limit Package Availability setting from FALSE to TRUE.
- 7** Click Continue, click Save, then click OK.

# Giving Users Trustee Rights to Packages

With the Package Request gadget configured to limit package availability, users will only have rights to a package if you make them trustees of the Package object and the package's container object. If you have many users who you want to make the package available to, you can add those users to a group and then make the group a trustee of a Package object and its container object. The following sections provide instructions:

- ♦ “Making Users Trustees of a Package Object” on page 88
- ♦ “Making Users Trustees of a Container Object” on page 88

## Making Users Trustees of a Package Object

- 1 In ConsoleOne®, right-click the Package object > click Trustees of This Object to display the Trustees of This Object page.
- 2 Click Add Trustee, browse for and select the user or group who you want the package to be available to, then click OK to display the Rights Assigned to Selected Objects dialog box.  
  
To make all objects in a container trustees of the Package object, you can select the container object rather than the individual User and Group objects.
- 3 Click OK to accept the default rights and add the user to the list of trustees.
- 4 Repeat **Step 2** and **Step 3** to assign additional users as trustees.
- 5 When finished adding users, click OK to save your changes.
- 6 Repeat **Step 1** through **Step 5** to assign users rights to another package.

## Making Users Trustees of a Container Object

Regardless of the trustee rights that a user has to a Package object, if he or she does not have trustee rights to the package's container object, the package will not be available.

You need to assign rights beginning with the company location container (as specified on the OnDemandService object) down to the package's container. For example, if the company location container is NOVELL, the catalog root container is NOVELL\PACKAGES, and the NOVELL\PACKAGES\APPLICATIONS container is where the package resides that you're making available, you need to assign trustee rights to the NOVELL, PACKAGES, and APPLICATIONS container.

**NOTE:** If desired, you can simply assign rights to the company location container (NOVELL, in the above example) and mark the rights as inheritable. However, this would cause the user's Package Request gadget to display all containers under the catalog root container (PACKAGES, in the above example), even if a container does not have any applications available to the user.

To make a user a trustee of a container:

- 1 In ConsoleOne, right-click the container object > click Trustees of This Object.
- 2 Click Add Trustee, browse for and select the user or group, then click OK to display the Rights Assigned to Selected Objects dialog box.  
  
To make all objects in a container trustees of the Package object, you can select the container object rather than the individual User and Group objects.
- 3 In the Property list, click [Entry Rights], then deselect Browse (make sure all rights are deselected).



- 4** In the Property list, click [All Attributes Rights], then deselect Compare and Read (make sure all rights are deselected).
- 5** Click Add Property, select OU, then click OK to add it to the Property list. Leave the rights set to the default (Compare and Read).
- 6** Click OK to add the user to the list of trustees.
- 7** Repeat **Step 2** through **Step 6** to make additional users trustees of the container.
- 8** When finished adding users, click OK to save your changes.
- 9** Repeat **Step 1** through **Step 8** to make users trustees of another container.



# 13

## Organizing Packages and Applications

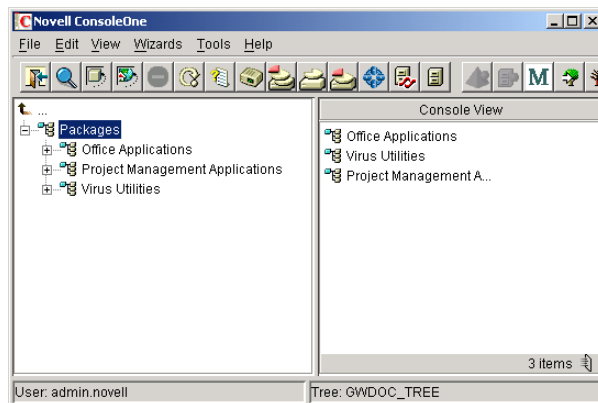
You can organize how packages are displayed in the Package Request gadget and how applications are displayed in the Launch Item gadget.

- ♦ “Organizing Packages in the Package Request Gadget” on page 91
- ♦ “Organizing Applications in the Launch Item Gadget” on page 92

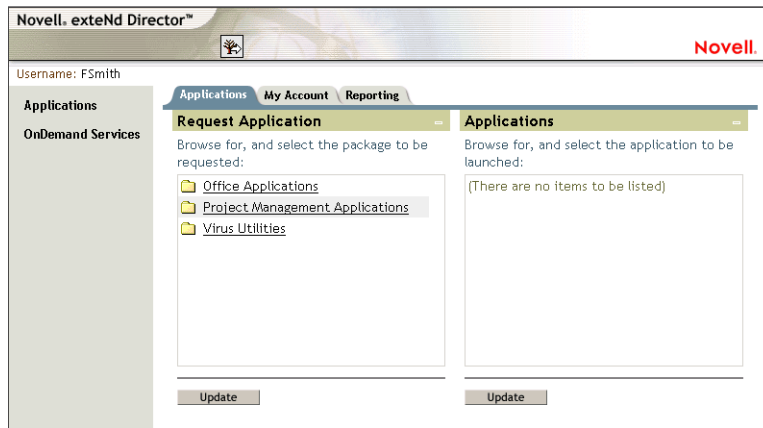
### Organizing Packages in the Package Request Gadget

The Package Request gadget displays packages exactly as they are organized in the package catalog in Novell® eDirectory™. The package catalog is the container structure where packages need to reside in order to be available to users. The location of the package catalog is determined by the catalog root container, which is specified in the OnDemandService object.

All packages can reside in the catalog root container, or you can organize your package catalog by creating containers under the catalog root. In the following example, Packages is the catalog root container and Office Applications, Virus Utilities, and Project Management Applications are three subcontainers.



The following example shows the resulting folder structure for the Package Request gadget. Notice that the name of the catalog root container (Packages, in the above example) is not displayed to users.



Because the Package Request gadget displays the catalog exactly as it is organized in eDirectory, you will want to organize the catalog in a manner that makes sense to both you and your users.

## Organizing Applications in the Launch Item Gadget

The Launch Item gadget displays desktop and thin-client applications in the locations designated in the Folders list of their Application objects (Application objects > Identification tab > Folders page). For information about using the Folders list, see Application Management in the [ZENworks for Desktops Administration guide](http://www.novell.com/documentation/lg/zdpr) (<http://www.novell.com/documentation/lg/zdpr>).

The Launch Item gadget displays Web applications in the location specified in the Catalog Position field on a Web Applications Commerce Item object (Commerce Item object > General tab > General page). For information about using the Catalog Position field, see “[Web Applications](#)” on [page 76](#).

# 14

## Securing Web Applications With Novell iChain

You can use Novell® iChain® 2.x to secure the Web applications you are provisioning through Novell ZENworks® Web Self-Service. Doing so ensures that a user will only have access to a Web application if he or she has purchased (or been assigned access to) the Web application's package.

To integrate Web Self-Service with iChain, you set up your Web applications as protected resources with an access level of Secure. You then create access control list (ACL) rules that allow access to the Web applications in these secured areas. When a user purchases a package that includes a Web application, Web Self-Service adds the user to the membership list of the ACL rule associated with the Web application. This gives the user access to the Web application until the purchase expires, at which time Web Self-Service removes the user from the rule's membership list.

Complete the tasks in the following sections to secure your Web applications through iChain:

- ♦ “Setting Up Protected Resources” on page 93
- ♦ “Creating ACL Rules” on page 94
- ♦ “Configuring iChain Server ACL List Refresh” on page 95
- ♦ “Making the OnDemand Admin User Object a Trustee of ACL Rules” on page 96
- ♦ “Associating ACL Rules with Web Applications” on page 97

For additional information about iChain, see the Novell iChain 2.0 documentation at the [Novell Documentation Web site \(http://www.novell.com/documentation\)](http://www.novell.com/documentation).

### Setting Up Protected Resources

If you have not already set up your Web applications as Secure access protected resources, you need to do so. In general, you should define the Web application folders, not the application files themselves, as protected resources. For example:

- ♦ If you have a single Web application folder structure (for example, the Web Application Server's webapp folder and subfolders), you can define that single Web application folder as a protected resource. Its files and subfolders become secured and are accessible only to users included in the membership lists of ACL rules that allow access.
- ♦ If your Web applications are located in different folders and do not have a common parent folder (or don't have a common parent folder that you want to define as a protected resource with Secure access level), you can define each Web application's folder as a protected resource. This accomplishes the same purpose as setting up a single protected resource; it simply involves a little more work for you.

Protected resources are configured through the iChain Service object (ISO) in Novell eDirectory™. To define a protected resource.

- 1** In ConsoleOne<sup>®</sup>, right-click the iChain Service object > click Properties to display the property pages.
- 2** Click the Protected Resources tab.
- 3** On the Protected Resource page, click the Add button to display the Add New Protected Resource dialog box.
- 4** Fill in the following fields:
  - Resource Name:** Enter a name to identify the protected resource. The name differentiates the resources in the Protected Resources list and must be unique within the list. You can, if desired, enter the resource's URL as its name.
  - URL Prefix:** Enter the resource's URL. For example,  
`www.novell.com/webapps/ *`  
would define the webapps directory and subdirectories as a protected resource. On the other hand,  
`www.novell.com/webapps/index.html`  
would define only the index.html file as a protected resource.
  - Access:** Select Secure so that access will be controlled through ACL rules.
  - Pass Parameters in the Query String:** Leave this option selected.
  - Pass Parameters as Header Variables:** Leave this option unselected.
- 5** Click OK to add the resource to the Protected Resource list.
- 6** Repeat **Step 3** through **Step 5** to set up additional protected resources.
- 7** When you've finished setting up protected resources, click OK to save your changes.

## Creating ACL Rules

When a Web application is part of a protected resource, it is accessible only through an ACL rule that has been configured to allow access to the Web application.

You can configure an ACL rule to provide access to multiple Web applications. Generally, however, you should create separate ACL rules for each Web application. If you inadvertently use the same ACL rule for Web applications in different packages, when a user purchases one of the packages he or she will also gain access to the Web applications in the other package. Although the user couldn't launch the unpurchased Web applications through the OnDemand Services Launch Item gadget, he or she could manipulate a purchased Web application's URL and possibly gain access to the unpurchased Web application.

To create an ACL rule:

- 1** In ConsoleOne, if you are using iChain 2.0, right-click the container where you want to create the object, click New, click Object to display the New Object dialog box, select iChain Access Control Rule, then click OK to display the New iChain Access Control Rule dialog box.  
or  
If you are using iChain 2.1, right-click the container where you want to create the object, click New, click iChain Object, select iChain Access Control Rule, then click OK to display the New iChain Access Control Rule dialog box.
- 2** Enter a name for the ACL Rule object, select Define Additional Properties, then click OK.

- 3** Click the Access Control tab.
- 4** Click the Add button on the right side of the Allowed URLs list to display the Add New Resource dialog box.
- 5** Fill in the following fields:
  - Resource Name:** Browse for and select the protected resource that the Web application is part of. The protected resource's URL forms the basis of the Web application's URL.
  - URL Postfix:** Enter the information that, when appended to the protected resource's URL, completes the Web application's URL.

An asterisk (\*) as the last character provides access to the folder content and all subfolders. A question mark (?) as the last character provides access to the folder contents but not the subfolders.

For example, if the protected resource's URL were `www.mycompany.com/webapps`, entering `/timemanager/?` would give the user access to all files in the `www.mycompany.com/webapps/timemanager` folder. An asterisk used in place of the question mark would grant access to that folder and its subfolders.
- 6** Click OK to add the URL to the Allowed URLs list.
- 7** Click OK to save the ACL rule.
- 8** Repeat **Step 1** through **Step 7** to create ACL rules that allow access to other Web applications.

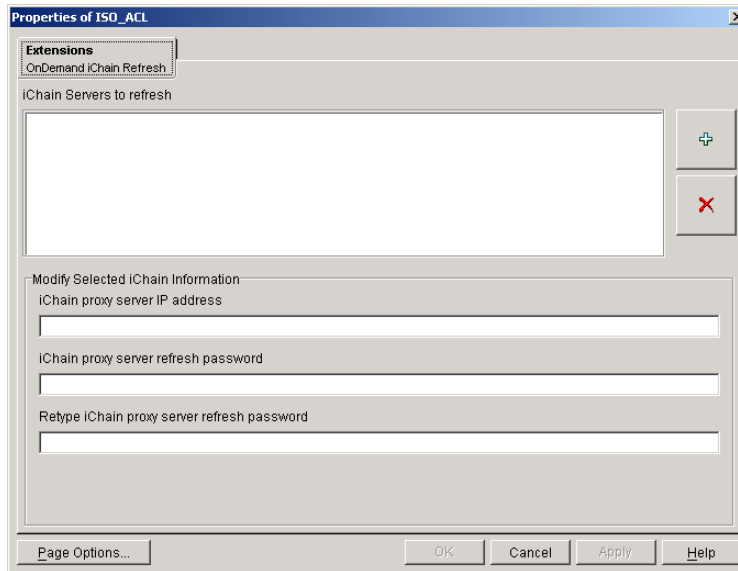
## Configuring iChain Server ACL List Refresh

Whenever Web Self-Service adds a user to an ACL rule's membership list or removes a user from the list, each iChain server where the ACL rule is being applied needs to be refreshed immediately so that the membership list changes will be enforced. Otherwise, a user who has just purchased a package with a Web application won't be able to run the application until the iChain server refreshes its ACL rules. Likewise, a user whose package has expired will be able to continue to use the package's Web applications until the iChain server refreshes.

Web Self-Service can immediately refresh the ACL rules of one or more iChain servers. For each ACL rule you create, you define the iChain servers you want Web Self-Service to refresh. When a user is added or removed from the ACL rule's membership list, Web Self-Service refreshes the specified iChain servers.

To define the iChain servers that Web Self-Service will refresh when an ACL rule's membership list is modified:

- 1** Right-click the ACL Rule object, then click Extensions of This Object.
- 2** Click Add Extension, select the `commerceProxySecretStore` auxiliary class extension, then click OK to display the OnDemand iChain Refresh page.



- 3 Click the Add (+ sign) button to add an address to the iChain Servers to Refresh list.
- 4 In the list, select the address that was just added to display it in the iChain Proxy Server Address field.
- 5 Modify the following fields:
 

**iChain Proxy Server Address:** Change the sample address to the iChain server's address. Press Enter after you've modified the address so that the list will reflect the change. If you don't press Enter, the change will not be made.

**iChain Proxy Server Refresh Password:** Type the existing password required to refresh ACL rules. This is the same password used by the iChain Config user.

**Retype iChain Proxy Server Refresh Password:** Confirm the password by retyping it.
- 6 Click Apply to save the changes.
- 7 Repeat [Step 3](#) through [Step 6](#) to define additional iChain server's whose ACL rules list should be refreshed when OnDemand Services updates the ACL rule.
- 8 When finished defining iChain servers, click OK, then click Close to close the extensions dialog box.

## Making the OnDemand Admin User Object a Trustee of ACL Rules

The OnDemand Admin User object must be a trustee of the ACL rules so that the OnDemand Services gadgets can read and update the rules.


- 1 Right-click the ACL Rule object, then click Trustees of This Object.
- 2 Click Add Trustee, browse for and select the OnDemand Admin User object, then click OK to display the Rights Assigned to Selected Objects dialog box.
- 3 Leave [Entry Rights] set to the defaults.
- 4 Select [All Attributes Rights], then click Write. Compare, Read, and Write should all be selected.
- 5 Click OK to add the OnDemand Admin User object to the list of trustees.
- 6 Click OK to close the Trustees of This Object page.



# Associating ACL Rules with Web Applications

For Web Self-Service to add a user to the correct ACL rule's membership list when the user purchases a package that contains a Web application, you need to add the ACL rule to the Snapin Command field in the Web Application's Commerce Item object.

To associate an ACL rule with a Web application:

- 1** Right-click the Web application's Commerce Item object > click properties.
- 2** In the Snapin Command field, click the Define Command button () to display the Snapin Command dialog box.
- 3** Browse for and select the ACL rule that allows access to the Web application, then click OK to close the Snapin Command dialog box.
- 4** Click OK to save your changes.



# 15 Customizing Notification Messages

Novell® ZENworks® Web Self-Service creates the following e-mail notifications:

- ♦ Package Request: Sent to a cost center's budget holder to inform him or her of a user's request for a package.
- ♦ Request Response: Sent to a user in response to his or her package request. Indicates whether the budget holder approved or denied the request.
- ♦ Purchase Information: Sent to a user to inform him or her that he or she has been assigned a package.

These e-mail notifications include text in the message subject and the message body that you can customize to meet your own needs.

- ♦ [“Modifying the Message Subject” on page 99](#)
- ♦ [“Modifying the Message Body” on page 99](#)

**IMPORTANT:** You must enable users to receive notification messages. This is done through the Receive Email Notifications setting on a User object (User object > OnDemand tab > Purchases page). For more detailed instructions, see [“Assigning Users to the Cost Center” on page 103](#). In addition, the E-Mail Address setting for each user and budget holder needs to be defined on their User objects (User object > General tab > Identification page).

## Modifying the Message Subject

The Subject line for all three notification messages must be the same. The default Subject line text is "Subject: notification of subscription".

To modify the Subject line:

- 1 In ConsoleOne®, right-click the OnDemandService object > click Properties.
- 2 In the Mail Subject field, type the text you want to use for the Subject line.
- 3 Click OK to save your changes.

## Modifying the Message Body

The CommerceMailMsg.properties file contains the body text for each notification message. The file is included in the CommerceLdapCmd.jar, located in the Tomcat webapps\mps\web-inf\lib directory.

To modify the message text:

- 1 Extract the CommerceMailMsg.properties file from the CommerceLdapCmd.jar file.
- or

If you are using a zip utility such as WinZip that lets you edit a file without first extracting the file, open the CommerceLdapCmd.jar file in that utility.

- 2** Open the CommerceMailMsg.properties file in a text editor.

The text for the notification messages is found in the following three sections:

```
## E-mail: Purchase Message
## E-mail: Request Approval Message
## E-mail: Response to Request Message
```

You can include any text you want by entering it. You can add comments to the file by entering text preceded by a pound sign (#).

The predefined arguments you can use are listed at the beginning of the file. For example, you can use argument 0 to include the current date and time in the text.

- 3** Edit the message text and save the changes.
- 4** If you extracted the CommerceMailMsg.properties file from the .jar file to edit it, add the edited file back into the .jar file.

# 16

## Creating Cost Centers

A Novell® ZENworks® Web Self-Service cost center performs two main functions:

- ♦ Supports user self-service. A user must belong to a cost center in order to request access to applications, and the cost center must have a least one budget holder who is responsible for approving or denying requests.
- ♦ Supports application fees. If you assign purchase and usage fees to applications, the application fees are charged to the user's cost center.

During installation, you should have created at least one cost center and assigned budget holders and users. The tasks required to create additional cost centers are identical to the ones described in installation. If you need help, refer to the following sections:

- ♦ [“Creating a Cost Center” on page 101](#)
- ♦ [“Adding Budget Holders” on page 102](#)
- ♦ [“Assigning Users to the Cost Center” on page 103](#)

## Creating a Cost Center

You create cost centers in Novell eDirectory™ by using Organizational Role objects. Each user you add as an occupant to the organizational role becomes a budget holder for the cost center.

- 1** In ConsoleOne®, right-click the container where you want to create the cost center, click New, then click Object.
- 2** Select Organizational Role in the Class list, then click OK to display the New Organizational Role dialog box.
- 3** Enter a name for the cost center (for example, DEPT12345), then click OK to create the object.  
**IMPORTANT:** Do not use an & character in the name.
- 4** Right-click the newly created object, then click Extensions of this Object.
- 5** In the Extensions of this Object dialog box, click Add Extension.
- 6** In the list of auxiliary class extensions, select commerceBudgetHolderRole, then click OK to display the BudgetHolder page.
- 7** Fill in the following fields:

**Current Budget Holder:** The cost center can have more than one budget holder. This allows for reassignment of the budget holder responsibilities from one person to another (for example, if a budget holder goes on vacation). The current budget holder determines which budget holder is active. Select the user you want to initially assign as the current budget holder.

**User Containers:** This field remains for backwards compatibility with previous versions of Novell ZENworks OnDemand Services. It is not used with Web Self-Service. Leave it blank.

**8** Click OK to close the page, then click Close to close the Extensions of this Object dialog box.

**9** To create additional cost centers, repeat **Step 1** through **Step 8**.

or

To add additional users as budget holders for the cost center, continue with the next section, **Adding Budget Holders**.

or

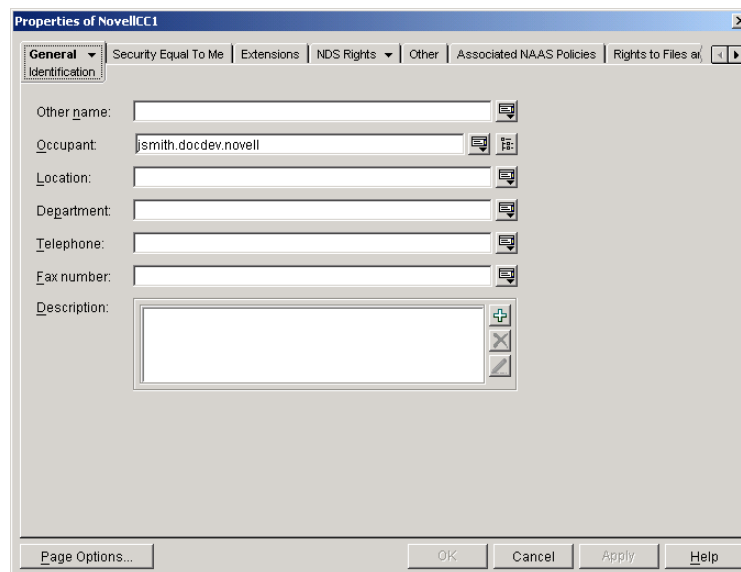
To assign users to the cost center, skip to “**Assigning Users to the Cost Center**” on page 103.

## Adding Budget Holders

When you create a cost center (see the previous section, **Creating a Cost Center**), you assign a user to be the current budget holder. Only the current budget holder can approve requests for the cost center. You can, however, add additional budget holders who can be assigned as the current budget holder at any time. The current budget holder assignment can be changed in ConsoleOne, or any of the budget holders can use the Current Budget Holder gadget to change the assignment.

To add budget holders to a cost center:

**1** In ConsoleOne, right-click the cost center’s object, then click Properties.

The screenshot shows a Windows-style dialog box titled "Properties of NovellCC1". It has several tabs: "General", "Security Equal To Me", "Extensions", "NDS Rights", "Other", "Associated NAAS Policies", and "Rights to Files and Folders". The "General" tab is selected, and within it, the "Identification" sub-tab is active. The form contains several fields: "Other name:" (empty), "Occupant:" (containing "jsmith.docdev.novell"), "Location:" (empty), "Department:" (empty), "Telephone:" (empty), "Fax number:" (empty), and "Description:" (a large empty text area). Each text field has a small icon to its right. At the bottom of the dialog are buttons for "Page Options...", "OK", "Cancel", "Apply", and "Help".

**2** On the Identification page, use the Occupant field to add the users who will be the cost center’s budget holders.

**3** Click OK to save your budget holder changes.

**4** Continue with the next section, **Assigning Users to the Cost Center**.

# Assigning Users to the Cost Center

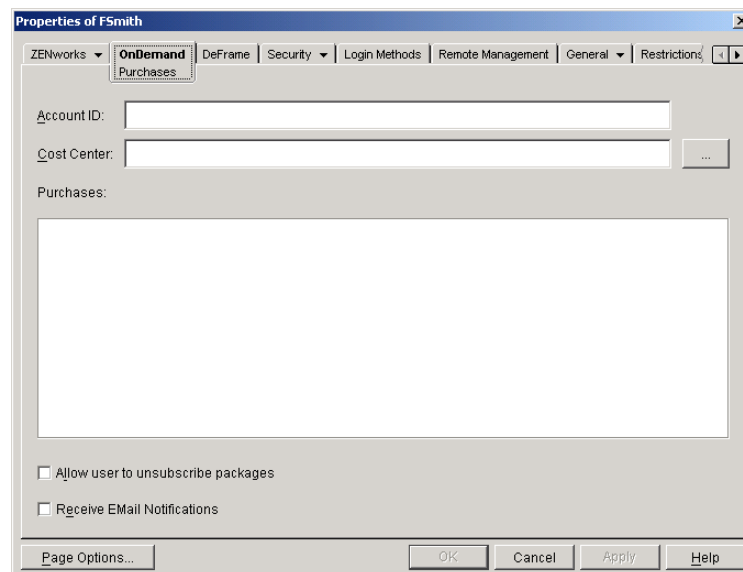
After you've created a cost center and assigned its budget holders, you need to assign users to the cost center.

You can use ConsoleOne to individually assign users to the cost center, or you can use the OnDemand User Configuration Utility to assign multiple users to a cost center at one time. The following sections provide instructions for both methods:

- ♦ “Using ConsoleOne to Individually Assign Users” on page 103
- ♦ “Using the OnDemand User Configuration Utility to Assign Multiple Users” on page 104

## Using ConsoleOne to Individually Assign Users

- 1 In ConsoleOne, right-click the User object, then click Properties.
- 2 Click the OnDemand tab to display the Purchases page.

The screenshot shows a Windows-style dialog box titled "Properties of FSmith". It has several tabs: ZENworks, OnDemand (selected), DeFrame, Security, Login Methods, Remote Management, General, and Restrictions. The "OnDemand" tab is active, showing a "Purchases" section. This section contains two text input fields: "Account ID:" and "Cost Center:". The "Cost Center:" field has a small browse button (three dots) to its right. Below these fields is a large, empty rectangular area labeled "Purchases:". At the bottom of the dialog, there are two checkboxes: "Allow user to unsubscribe packages" and "Receive EMail Notifications", both of which are currently unchecked. At the very bottom are buttons for "Page Options...", "OK", "Cancel", "Apply", and "Help".

- 3 Fill in the following fields:

**Account ID:** The user's account ID is recorded on each Purchase and Usage object that he or she generates. If you have a third-party billing system or reporting system, you can base your billing or reporting on the account ID.

The account ID can be a unique identifier, such as the user's employee workforce number, or it can be a common identifier, such as the user's cost center ID as it is defined in the billing or reporting system.

**Cost Center:** Browse to and select the cost center that you want the user assigned to. Cost centers are represented by Organizational Role objects.

**Allow User to Unsubscribe Packages:** If you want the user to be able to cancel his or her subscription to a package, select this option. If you do so, the user can use the Workflow Tracking gadget to view an approved subscription purchase and cancel the subscription.

**Receive E-Mail Notifications:** Select this option to enable the user to receive e-mail messages notifying him or her of approved or denied purchases.

- 4 Click OK to save the information.

## Using the OnDemand User Configuration Utility to Assign Multiple Users

The OnDemand User Configuration utility lets you do the following:

- ◆ Configure the cost center and account ID.
- ◆ Enable users as ZfD DeFrame™ users. If you've already enabled users as DeFrame users, the utility will not affect the DeFrame settings.

This Java-based utility is included on the *ZENworks 6 Companion 1* CD. Before you can use it, you will need to install it to a server or workstation. You will then need to run it from a Windows workstation/server that has a Java Runtime Engine (JRE) installed.

- 1** Copy the files from the `zenworksfordesktops\odusrcfg` directory on the the *ZENworks 6 Companion 1* CD to a server or workstation directory.

The files can be copied to any server or workstation location. One recommended location is the ConsoleOne directory.

- 2** From a Windows workstation/server that has a JRE installed, run `odusrcfg.bat` from the directory to display the OnDemand User Configuration Utility Introduction page.

- 3** Click Next to display the Novell eDirectory Authentication page.

- 4** Fill in the following fields:

**LDAP Server Hostname or IP Address:** Enter the hostname or IP address of an LDAP server that provides access to eDirectory. If the LDAP server's port is not 389, include the port number.

**Admin Name:** Using LDAP syntax, specify the context of a user that has admin-equivalent eDirectory rights.

**Password:** Specify the user's password.

- 5** Click Next to authenticate to eDirectory and display the User Selection page.

- 6** Select the Enable ApprovalFlow option.

- 7** Select the users you want to assign to a cost center.

All users you select will be assigned to the same cost center. You can select individual users, or you can select a container to add all the container's users.

- 8** Click Next to display the ApprovalFlow Settings page.

- 9** Fill in the following fields:

**Account ID:** Enter an account ID to identify the user. The user's account ID is recorded on each Purchase and Usage object that he or she generates. If you have a third-party billing system or reporting system, you can base your billing or reporting on the account ID.

**Cost Center:** Browse to and select the cost center that you want the user assigned to. Cost centers are represented by Organizational Role objects.

**Enable User to Receive E-Mail Notifications:** Select this option to enable the user to receive e-mail messages notifying him or her of approved or denied purchases.

**Enable User to Cancel Package Subscriptions:** If you want the user to be able to cancel his or her subscription to an application package, select this option. If you do so, the user can use the Workflow Tracking gadget to view an approved subscription purchase and cancel the subscription.



**Overwrite Existing ApprovalFlow Settings:** Select this option to have these settings overwrite any ApprovalFlow settings currently assigned to the selected users.

**10** Click Finish, then click Yes to confirm that you want to configure the selected users.

**11** When the user configuration is complete, click Finish to exit the utility.



# 17

## Billing for Purchases

To be able to bill for purchases, you need to extract the purchase information from Novell® eDirectory™ to an existing billing system. If you already have an interface (for example, a DirXML® or JDBC\* driver) for accessing and managing information from eDirectory, you can modify that interface to also extract and manage the purchase information. If you don't have an interface, you will need to develop one or have one developed for you.

- ♦ “Developing Your Own Interface” on page 107
- ♦ “Getting Development Help” on page 107

### Developing Your Own Interface

The [Novell Developer Kit site \(http://developer.novell.com/ndk\)](http://developer.novell.com/ndk) contains developer kits to help you access and manage information in eDirectory. The *eDirectory SDK for Java* kit contains the libraries, tools, sample code, and documentation necessary for you to fully integrate a Java application with eDirectory and the *eDirectory SDK for C* kit contains similar items for you to integrate a C application.

As you build your interface, keep in mind the following:

- ♦ The Purchase objects and Usage objects contain the information (cost center, user name, user account ID, and so forth) you will want to extract for billing purposes.
- ♦ To decide which Purchase objects to bill for, you might want to use the Expired status or date in the Purchase object.
- ♦ After you have extracted a purchase's information from eDirectory into your billing system, you'll want to remove the Purchase object and its Usage objects. If you mark the Purchase object as Expired and Billed, the Maintenance process will remove it and its Usage objects. Because a Purchase object cannot be deleted until after its Usage objects have been deleted, it might take the Maintenance process two cycles to remove the objects: the first to remove the Usage objects and the second to remove the Purchase object.
- ♦ You can develop Java classes that perform tasks on Purchase and Usage objects, and then have the Maintenance process run the classes. For information, see the [purchaseClassx](#) and [usageClassx](#) entries under “Modifying the Maintenance and UsageServer Configuration File (Commerceserver.cfg)” on page 121.

### Getting Development Help

If you want help developing an interface, Novell provides several options:

- ♦ **Developer Labs:** Developer Labs will work side-by-side with you to develop the interface between your billing system and eDirectory. For information, see the [Novell Developer Support site \(http://developer.novell.com/support\)](http://developer.novell.com/support).

- ♦ **Novell Consulting:** Novell Consulting<sup>SM</sup> will work with you to design, develop, and implement a solution that best fits your environment and needs. For information, see the [Novell Consulting site \(http://www.novell.com/solutions/ngage\)](http://www.novell.com/solutions/ngage)

# 18

## Changing the Status of a Purchase

When a user receives access to a package, Novell® ZENworks® Web Self-Service creates a Purchase object under the Package object in Novell eDirectory™. In addition, if the purchase is usage-based, when the user runs any of the package's applications the UsageServer process creates a Usage object under the Purchase object.

Throughout the lifetime of a user's purchase, the following statuses are recorded on the Purchase object to track and manage the purchase:

- ♦ **Billed:** The user's cost center has been billed for the purchase. This status is not marked automatically. To change this status, you must manually mark it or have another application (such as a Java application that provides an interface between your billing system and eDirectory) mark it.
- ♦ **Expired:** The purchase has expired. Web Self-Service automatically marks this status whenever the purchased time has been used or the ending date for the purchase has passed.
- ♦ **Deletable:** The purchase is 30 days past its expiration date and has been billed. Web Self-Service automatically marks this status. After the purchase is marked deletable, the Maintenance process will delete the Purchase object and its Usage objects. Because a Purchase object cannot be deleted until after its Usage objects have been deleted, it might take the Maintenance process two cycles to remove the objects: the first to remove the Usage objects and the second to remove the Purchase object.

Thirty (30) is the default number of days. You can change this number in the `commerceserver.cfg` file, located in the `sys:\zenworks\ondemand` directory on a NetWare® server and the `\zenworks\ondemand` directory on a Windows server. You can also determine how often the Maintenance process runs; the default is every 10 minutes. For additional information, see [“Managing the Maintenance and UsageServer Processes” on page 115](#).

- ♦ **Subscription:** The purchase is a subscription purchase. When this subscription period expires, Web Self-Service will automatically renew the subscription and create a new Purchase object for that subscription period.

To modify the status of a purchase:

- 1** In ConsoleOne®, right-click the Purchase object > click Properties.  
A Purchase object is located under its Package object.
- 2** On the General page, select or deselect the status you want to change.
- 3** Click OK.



# 19

## Managing User Workstation Software

In order for the Novell® ZENworks® OnDemand Services Launch Item gadget and Novell Application Launcher™ to launch desktop and DeFrame™ thin-client applications, the following workstation software components are used:

- ♦ **DeFrame Client:** Required to launch DeFrame thin-client applications from the Launch Item gadget and Novell Application Launcher.
- ♦ **ICA Web Client:** Required to launch DeFrame thin-client applications in an ICA client session. Version 6.30.1050.0 is used.
- ♦ **RDP Web Client:** Required to launch DeFrame thin-client applications in an RDP client session. Version 5.0.2221.1 is used.
- ♦ **NAL Plug-In:** Required to launch desktop applications from the Launch Item gadget. The UsageClient files are also included with the plug-in.

The following sections provide information to help you better understand and manage the workstation software:

- ♦ [“Installing the Workstation Software” on page 111](#)
- ♦ [“Updating the ICA Web Client” on page 113](#)

### Installing the Workstation Software

In order for a user to launch applications from the Launch Item gadget, the appropriate workstation software components must be installed on the user’s workstation. There are several ways to install the components, as explained in the following sections:

- ♦ [“Using the OnDemand Services Launch Item Gadget to Install the Workstation Software” on page 111](#)
- ♦ [“Using Novell Application Launcher to Install the Workstation Software” on page 112](#)
- ♦ [“Manually Installing the Workstation Software” on page 113](#)

### Using the OnDemand Services Launch Item Gadget to Install the Workstation Software

The easiest way to distribute the workstation software is through the Launch Item gadget. The first time a user launches an application, the Launch Item gadget automatically installs the appropriate components to the workstation. For example, when the user launches a desktop application, the NAL plug-in is installed. When the user launches a DeFrame thin-client application configured for an RDP client session, the RDP client and the DeFrame client (if necessary) are installed.

The clients are packaged as the following .cab files:

- ♦ **dappx.cab:** Contains the DeFrame client. The Launch Item gadget will install this client only if a user launches a thin-client application and the client is not already installed on the user's workstation.
- ♦ **wficat.cab:** Contains the ICA Web client. The .cab file contains the 6.30.1050.0 version of the ICA Web client that replaces all versions prior to May 2002. The Launch Item gadget installs the ICA Web client the first time a user launches an application configured for an ICA session.
- ♦ **nmstsc.cab:** Contains the RDP Web client, version 5.0.2221.1. The Launch Item gadget installs the RDP Web client the first time a user launches an application configured for an RDP session.
- ♦ **nalexec.cab:** Contains the NAL plug-in and UsageClient. The Launch Item gadget installs these files the first time a user launches a desktop application or a usage-based application (desktop, thin-client, or Web).

For the software to be installed on a Windows 95/98 workstation, the user must be logged into Windows on the workstation.

For the software to be installed on a Windows 2000/XP workstation, the user must be logged in as a member of the Administrators or Power Users group. Logging in as Administrator or a Power User is only required the first time; users can log in through a non-Administrator account thereafter.

The .cab files are located in the following directory:

```
tomcat\webapps\nps\portal\gadgets\com.novell.ondemand.gadgets.LaunchItemGadget\bin
```

## Using Novell Application Launcher to Install the Workstation Software

During installation of the ZfD Server software to a network server, the installation program created an OnDemand Client Application object in the same eDirectory container as the ZfD Server object.

The OnDemand Client Application object is configured to run the Workstation setup program one time. The setup program requires user interaction to confirm installation of the various clients (DeFrame client, ICA client, RDP client, and NAL plug-in); users will not need to provide configuration information such as installation paths. You might want to manually run the setup program one time to see how users will need to interact.

The following are examples of how you can use the OnDemand Client Application object:

- ♦ You can directly associate the OnDemand Client Application object with users, in which case they will need to launch the Application object to run the setup program. For information about associating Application objects to users, see [Distributing Applications to Users and Workstations \(http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/a7dup3a.html#a7dup3a\)](http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/a7dup3a.html#a7dup3a) in Application Management in the ZENworks for Desktops *Administration* guide.
- ♦ You can assign the OnDemand Client Application object as an application dependency for DeFrame thin-client Application objects (or any type of Application objects if you want to ensure that the files are installed). By making the OnDemand Client Application object a dependency for the thin-client applications, the setup program will run the first time a user launches one of the thin-client applications. For information about setting up application dependencies, see [Setting Up Application Dependencies \(http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/aflms6k.html#aflms6k\)](http://www.novell.com/documentation/lg/zdpr/zdpradmn/data/aflms6k.html#aflms6k) in Application Management in the ZENworks for Desktops *Administration* guide.



## Manually Installing the Workstation Software

- 1** Make sure the user has the rights required to install software to his or her workstation.
  - ♦ On a Windows 2000/XP workstation, the user must be logged in as a member of the Administrators or Power Users group. Logging in as Administrator or a Power User is only required the first time; users can log in through a non-Administrator account thereafter.
  - ♦ On a Windows 95/98 workstation, the user must be logged into Windows on the workstation.
- 2** Run the setup.exe program from one of the following locations:
  - ♦ The \WorkstationSetup directory on the *ZENworks 6 Web Self-Service Program* CD.
  - ♦ The \DFWorkstationSetup directory on the *ZENworks for Desktops 4.0.1 Companion* CD.
  - ♦ The \zenworks\ondemand\setup directory on a ZfD 4.0.1 server.
  - ♦ The \public\zenworks\thinclnt\setup directory on a ZfD 4.0.1 server.

## Updating the ICA Web Client

You can update the ICA Web client whenever Citrix\* releases a new version. This is not supported for the RDP Web client.

- 1** Download the new ICA Web client .cab file from the [Citrix ICA client download site \(http://www.citrix.com/download/ica\\_clients.asp\)](http://www.citrix.com/download/ica_clients.asp).
- 2** Copy the file to the Launch Item gadget's bin directory. The full path is:  
`tomcat\webapps\nps\portal\gadgets\com.novell.ondemand.gadgets.LaunchItemGadget\bin`
- 3** Change the ICAVersion property in main.xsl to the new version number:
  - 3a** Open the file in an editor. The file is located in the following directory:  
`tomcat\webapps\nps\portal\gadgets\com.novell.ondemand.gadgets.LaunchItemGadget`
  - 3b** Locate the following line:  
`<!ENTITY ICAVersion "6,30,1050,0">`
  - 3c** Change the version to the new version number. If you are not sure of the new version number, you can open the .cab file, extract one of the .DLL files, then check the version information on the file properties.
  - 3d** Save the changes to the file.
- 4** Refresh the portal style sheets:
  - 4a** Log in to the portal as a portal admin
  - 4b** Click Portal Administration > Administer the Portal to display the Portal Administration page.
  - 4c** Click Portal > Refresh Portal.
  - 4d** Under Caching, select Stylesheets for Backend Rendering, then click Refresh.

The next time a user launches an ICA thin-client application through the Launch Item gadget, the new ICA Web client is installed to the user's workstation.



# 20

## Managing the Maintenance and UsageServer Processes

Novell® ZENworks® Web Self-Service includes two Java processes, Maintenance and UsageServer, that provide Novell eDirectory™ maintenance and application usage tracking. The following sections provide information to help you understand and manage the two processes:

- ♦ [“Understanding the Maintenance Process” on page 115](#)
- ♦ [“Understanding the UsageServer Process” on page 115](#)
- ♦ [“Explicitly Defining the UsageServer’s Address” on page 118](#)
- ♦ [“Installing Additional Maintenance and UsageServer Processes” on page 120](#)
- ♦ [“Modifying the Maintenance and UsageServer Configuration File \(Commerceserver.cfg\)” on page 121](#)
- ♦ [“Modifying Logging Settings” on page 123](#)

### Understanding the Maintenance Process

The Maintenance process monitors the Purchase and Usage objects associated with packages stored in eDirectory. When a purchase expires, the Maintenance process marks it as Expired. Thirty days later (by default), the Maintenance process deletes any expired Purchase objects (and the associated Usage objects) provided that you’ve marked them as Billed. The Maintenance process will not mark a Purchase object as billed; you must do it manually (see [Chapter 18, “Changing the Status of a Purchase,” on page 109](#)) or create a utility to do so (see [Chapter 17, “Billing for Purchases,” on page 107](#)).

You can run multiple Maintenance processes if necessary. However, each process must be responsible for a specific portion of the eDirectory tree. In other words, you cannot have two Maintenance processes maintaining the same Purchase and Usage objects. You control this by using the baseContext setting in the commerceserver.cfg file to specify the base container for the Maintenance process. The Maintenance process will only maintain Purchase and Usage objects located within the base container and its subcontainers. For information about the commerceserver.cfg file, see [“Modifying the Maintenance and UsageServer Configuration File \(Commerceserver.cfg\)” on page 121](#)

### Understanding the UsageServer Process

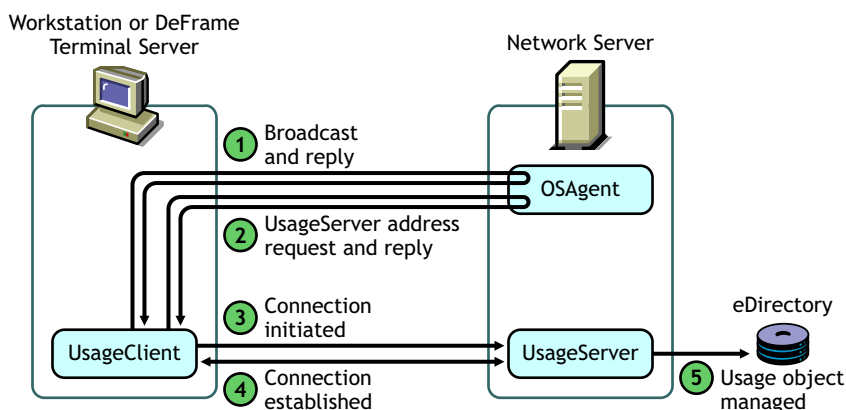
The UsageServer combines with the UsageClient to track application usage. You must run the UsageServer if any packages use a pricing scheme based on time or use (see [“Establishing the Package’s Cost” on page 80](#)). The following sections provide information about the UsageServer process:

- ♦ [“How the UsageServer and UsageClient Work” on page 116](#)

- ♦ “What to Do If the UsageServer and the UsageClient Are in Different Subnets” on page 116
- ♦ “Multiple UsageServers and Load Balancing” on page 118

## How the UsageServer and UsageClient Work

The UsageServer consists of two components, the OSAgent and the UsageServer process, both of which run on the same server. The OSAgent is responsible for facilitating a connection between the UsageClient (running on the user’s workstation or the DeFrame terminal server) and the UsageServer process. The UsageServer is responsible for maintaining usage information in eDirectory. The following diagram illustrates this process.



1. When a user launches a usage-based application, the UsageClient broadcasts a request looking for available OSAgents. The OSAgent replies with its IP address.
2. The UsageClient requests the UsageServer’s IP address from the OSAgent. The OSAgent passes the UsageServer’s IP address to the UsageClient.
3. The UsageClient initiates a connection with the UsageServer.
4. The UsageClient and UsageServer establish a connection.
5. Based on information passed to it by the UsageClient, the UsageServer then accesses eDirectory to create the Usage object associated with the application.

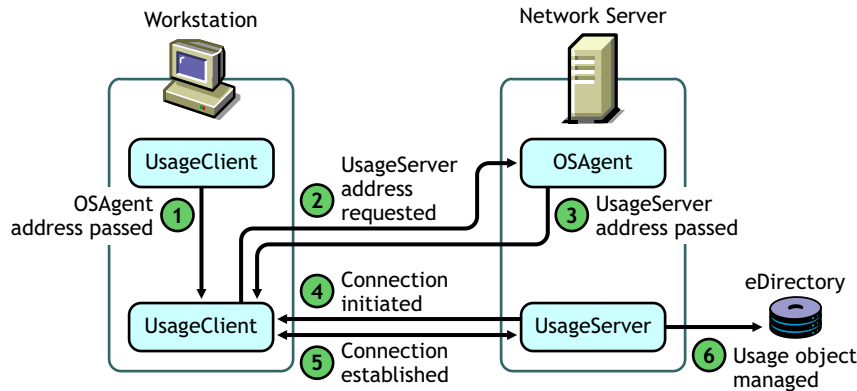
## What to Do If the UsageServer and the UsageClient Are in Different Subnets

For a UsageClient broadcast to find an OSAgent, as shown in the diagram in the previous section, [How the UsageServer and UsageClient Work](#), the UsageClient and OSAgent must be in the same subnet. If you have user workstations or DeFrame™ terminal servers that will not be in the same subnet as the OSAgent, you can explicitly provide the UsageClient with the OSAgent’s IP address.

### Usage-Based Desktop Applications

With desktop applications, you use the Launch Item gadget’s UsageServer setting to explicitly define the IP address of the OSAgent you want the UsageClient to connect to. The following diagram illustrates the desktop application usage process when the UsageServer’s address is explicitly defined in the Launch Item gadget.

## Desktop Application

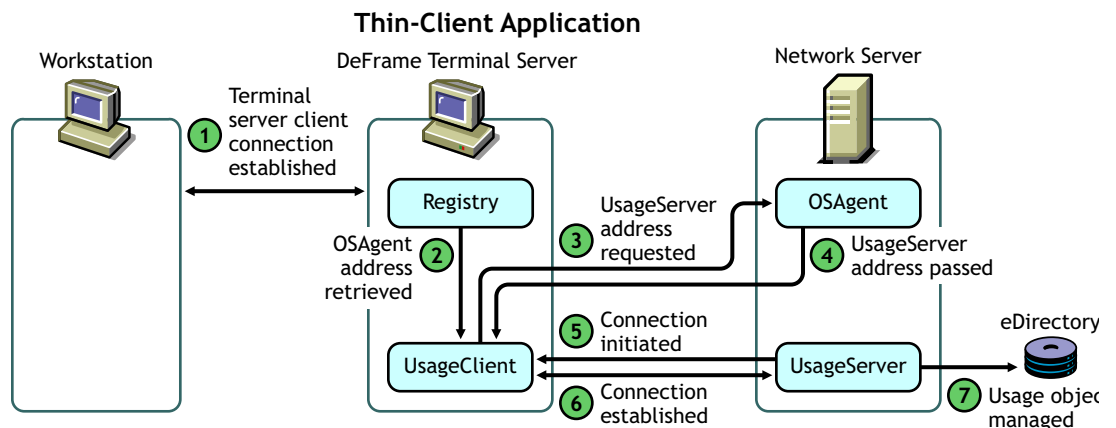


1. When a user launches a usage-based desktop application, the Launch Item gadget passes the OSAgent's IP address to the UsageClient.
2. The UsageClient requests the UsageServer's IP address from the OSAgent.
3. The OSAgent passes the UsageServer's IP address to the UsageClient.
4. The UsageClient initiates a connection with the UsageServer.
5. The UsageClient and UsageServer establish a connection.
6. Based on information passed to it by the UsageClient, the UsageServer accesses eDirectory to create a Usage object associated with the application.

For information about configuring the Launch Item gadget to provide the UsageServer address, see [“Adding the UsageServer's Address to the Launch Item Gadget or Novell Application Launcher” on page 119.](#)

## Usage-Based Thin-Client Applications

With DeFrame thin-client applications, the DeFrame server's UsageClient, not the workstation's UsageClient, tracks application usage. The DeFrame server's UsageClient does not receive the OSAgent's IP address from the Launch Item gadget. Instead, you need to add the OSAgent's IP address to the DeFrame server's Windows registry. The UsageClient then pulls the IP address from the registry. The following diagram illustrates the thin-client application usage process when the UsageServer's address is explicitly defined in the DeFrame server's registry.



1. When a user launches a usage-based thin-client application, a client session is established with the DeFrame server.
2. The UsageClient, running on the DeFrame server, retrieves the OSAgent's IP address from the Windows registry.
3. The UsageClient requests the UsageServer's IP address from the OSAgent.
4. The OSAgent passes the UsageServer's IP address to the UsageClient.
5. The UsageClient initiates a connection with the UsageServer.
6. The UsageClient and UsageServer establish a connection.
7. Based on information passed to it by the UsageClient, the UsageServer then accesses eDirectory to create the Usage object associated with the application.

For information about adding the UsageServer's address to a DeFrame server's Windows registry, see [“Adding the UsageServer's Address to a DeFrame Server's Registry” on page 120](#).

## Multiple UsageServers and Load Balancing

If your usage tracking requirements are placing a heavy load on a single UsageServer, you can set up multiple UsageServers to lessen the server load. For information about installing additional UsageServers, see [“Installing Additional Maintenance and UsageServer Processes” on page 120](#).

The UsageServer load balancing capabilities are limited to round-robinning the usage sessions. For example, if you have three UsageServers (U1, U2, and U3), the first session will be assigned to U1, the second to U2, and the third to U3. The fourth session will then be assigned to U1, the fifth to U2, and so forth. Because the usage sessions handled by one UsageServer might end before another UsageServer's sessions and new sessions are added in this round-robin manner, true load balancing might not be achieved. Overall, however, the workload will be lessened on any one server.

In a load-balanced UsageServer environment, it doesn't matter which UsageServer makes the initial usage session connection. The UsageClient will be given the address of the UsageServer that is supposed to handle the next session.

For UsageServer load balancing to work, the following conditions must be met:

- ♦ All UsageServers must be located in the same eDirectory tree.
- ♦ All UsageServers must be located in the same network subnet. If they are not, the UsageServer's OSAgents will not know about each other.

## Explicitly Defining the UsageServer's Address

If you have user workstations or DeFrame servers that are not in the same subnet as the UsageServer's OSAgent, the OSAgent will not receive the UsageClient's broadcast. This will result in the usage-based application failing to launch because the usage session could not be opened. To resolve this problem, you need to explicitly provide the UsageClient with the OSAgent's address.

The UsageClient receives the OSAgent's address from the Launch Item gadget or Novell Application Launcher for usage-based desktop applications or from the DeFrame server's Windows registry for DeFrame thin-client applications. The following sections provide instructions for explicitly defining the UsageServer's OSAgent's address in the appropriate

locations. If you have both usage-based desktop applications and usage-based thin-client applications, you will need to complete the tasks in both sections.

- ♦ “Adding the UsageServer’s Address to the Launch Item Gadget or Novell Application Launcher” on page 119
- ♦ “Adding the UsageServer’s Address to a DeFrame Server’s Registry” on page 120

## Adding the UsageServer’s Address to the Launch Item Gadget or Novell Application Launcher

With desktop applications, you can use the Launch Item gadget’s UsageServer setting or the Novell Application Launcher’s UsageServer setting to explicitly define the IP address or host name of the UsageServer you want the UsageClient to connect to. When the Launch Item gadget or Novell Application Launcher launches a usage-based desktop application, it passes the IP address or host name to the UsageClient on the user’s workstation. The UsageClient then connects directly to the UsageServer’s OSAgent.

If you need to specify the same UsageServer address for all users, you can add it to the Launch Item gadget. If you need to specify different UsageServer addresses for different users, you will need to use the Novell Application Launcher settings.

### Adding the UsageServer’s Address to the Launch Item Gadget

- 1** Log in to Novell Portal Services (NPS) as a portal administrator.
- 2** Click Administer the Portal.
- 3** Click Gadgets.
- 4** Select the OD\_LaunchItemGadget, then click Edit.
- 5** Click Configuration.
- 6** In the UsageServer field, enter the IP address or DNS host name of the network server where the UsageServer process is installed.
- 7** Click Continue, click Save, then click OK.

### Adding the UsageServer’s Address to Novell Application Launcher

- 1** In ConsoleOne, right-click the User object to which you want to apply new settings, then click Properties.  
  
You can define the UsageServer’s address for multiple users by selecting the users’ container object rather than the individual User objects.
- 2** Click the ZENworks tab, then click Launcher Configuration to display the Launcher Configuration page.  
  
The Launcher Configuration page provides three modes you can use to view the configuration settings for the current object. By default, the View/Edit Object’s Custom Configuration mode is selected.
- 3** In the Mode list, make sure the View/Edit Object’s Custom Configuration mode is selected.
- 4** Click Add to display the Launcher Configuration dialog box.
- 5** On the User tab, select the OnDemand UsageServer setting in the Settings list.

- 6** In the Values field, select Custom, then enter the IP address or DNS host name of the network server where the UsageServer process is installed.
- 7** Click OK to save the setting and close the Launcher Configuration dialog box.
- 8** Click OK to close the object's property pages.

## Adding the UsageServer's Address to a DeFrame Server's Registry

With DeFrame thin-client applications, the UsageClient on the DeFrame server, not the UsageClient on the user's workstation, tracks application usage. The DeFrame server's UsageClient does not receive the IP address from the Launch Item gadget or Novell Application Launcher. Instead, if you want to explicitly define the UsageServer's address, you need to add the IP address or hostname to the DeFrame server's Windows registry.

- 1** At the DeFrame server, use regedit.exe to open the registry.
- 2** Add the following key:  
`HKEY_LOCAL_MACHINE\SOFTWARE\Novell\on-demand\Objects`
- 3** Add a String value to the Objects key, using the following information:

**Value Name:** -ORBagentaddr

**Value Data:** Enter the IP address or DNS host name of the network server where the UsageServer process is installed.

- 4** Save the registry changes.

## Installing Additional Maintenance and UsageServer Processes

The Maintenance and UsageServer processes can be run from multiple locations at the same time, enabling you to balance the workload among several servers.

The two processes are installed by the ZfD Server installation program. If you want to run the processes on a server that is not a ZfD server, you can copy the files from the ZfD server.

- 1** Manually copy the \zenworks\ondemand directory from a ZfD server to another server that meets the requirements listed in [“Network Server Requirements” on page 53](#).
- 2** Modify the following settings in the \zenworks\ondemand\CommerceMaintenance.cfg file:

**ldap.host:** Specifies the IP address or DNS host name of the LDAP server used to authenticate to eDirectory. The default IP address, 127.0.0.1, provides authentication through the local LDAP server.

**ldap.loginName:** Specifies the user through which authentication occurs. Use LDAP syntax to specify the user's distinguished name (for example, cn=admin,o=novell).

If the user's distinguished name includes extended (multibyte) characters and the Maintenance process is running on a NetWare® server, you need to run commerceserver.cfg through the Java utility native2ascii. At the server console, enter the following:

```
native2ascii \zenworks\ondemand\commerceserver.cfg
```

This needs to be done any time you change the user name.

**baseContext:** Specifies the container that the Maintenance process will use as its root container. The Maintenance process only processes Purchase and Usage objects located in the



root container and its subcontainers. Use LDAP syntax to specify the root container distinguished name (for example, ou=apps,ou=services,o=novell).

**IMPORTANT:** You need to make sure that no two Maintenance processes are configured to maintain the same part of the eDirectory tree. If necessary, modify the CommerceMaintenance.cfg files for your other processes to ensure unique base contexts.

- 3** If you want to modify other settings in the CommerceMaintenance.cfg file, refer to [“Modifying the Maintenance and UsageServer Configuration File \(Commerceserver.cfg\)” on page 121](#) for descriptions of the settings.
- 4** To start the Maintenance process: At the NetWare server console, enter  
**sys:\zenworks\ondemand\bin\commercemaintenance.ncf.**  
or  
At the Windows server command prompt, change to the Maintenance process directory (by default, \zenworks\ondemand\bin), then enter **commercemaintenance.bat.**
- 5** To start the Usage Server process: At the NetWare server console, enter  
**sys:\zenworks\ondemand\bin\usageserver.ncf.**  
or  
At the Windows server command prompt, change to the directory where the Usage Server was installed (by default, \zenworks\ondemand\bin), then enter **usageserver.bat.**

## Modifying the Maintenance and UsageServer Configuration File (Commerceserver.cfg)

The Maintenance and UsageServer processes share the same configuration file. The configuration file, named commerceserver.cfg, is located in the \zenworks\ondemand directory of the server where the two processes are installed.

You can use any text editor to edit the file. The configuration settings are described below.

**ldap.host:** Specifies the IP address or DNS host name of the LDAP server through which the Maintenance and UsageServer processes access eDirectory.

**ldap.loginName:** Specifies the distinguished name of the user through which the Maintenance and UsageServer processes will authenticate to eDirectory. Use LDAP syntax to specify the name (for example, cn=admin,o=novell).

If the user name includes extended (multibyte) characters and the Maintenance and UsageServer processes are running on a NetWare server, you need to run commerceserver.cfg through the Java utility native2ascii. At the server console, enter the following:

```
native2ascii \ondemand\commerceserver.cfg
```

This needs to be done any time you change the user name.

**ldap.useSSL:** Specifies whether or not SSL will be used for the connection to the LDAP server. Values are TRUE and FALSE.

**ldap.port.SSL:** Specifies the SSL LDAP port number being used by the LDAP server. This port will be used if ldap.useSSL is set to TRUE. The standard SSL LDAP port is 636.

**ldap.port.clearText:** Specifies the non-SSL LDAP port number being used by the LDAP server. This port will be used if ldap.useSSL is set to FALSE. The standard non-SSL LDAP port is 389.

**UsageHeartbeatMinutes:** This setting is used by both the Maintenance and UsageServer processes to ensure accurate purchase records for packages with usage-based pricing schemes.

When a user launches a package's application, a Usage object is created under the Purchase object associated with the package. The Usage object contains a beginning time stamp created by the UsageServer. After each 10 minutes of use (the default), the UsageServer adds an interim time stamp called the heartbeat time stamp.

When the user exits the application, the UsageServer removes the heartbeat time stamp and creates the ending time stamp. However, if something happens during the time the user is using the application so that the UsageServer is unable to create the ending time stamp, the last heartbeat time stamp remains.

The Maintenance process, during its maintenance task, looks for any Usage objects with heartbeat time stamps that are older than 10 minutes. If this situation exists, it uses the heartbeat time stamp as the ending time stamp. This ensures that the user is not billed for more time than the UsageServer was actually able to record

**MaintainEveryMinutes:** Specifies how often the Maintenance process will read eDirectory to process any Purchase and Usage records. The default is every 10 minutes.

Take care when configuring this setting. One of the tasks the Maintenance process performs is to revoke a user's package rights when the user's purchase expires. If you set a long maintenance interval (for example, 600 minutes or 10 hours) and the purchase pricing scheme is Clock Time, a user will continue to have access to the purchased package for up to 10 hours after the user's purchased time ended (provided the user does not log out of the portal or refresh the Launch Item gadget's list). The extra time will depend on where the Maintenance process is in its time interval when the user's purchased time runs out. For example, if it performed maintenance 4 hours prior to the user's time running out, the user could continue to use the package for another 6 hours if he or she doesn't log out of the portal or update the Launch Item gadget.

**DeletePurchasesPostExpirationDays:** Specifies how long the Maintenance process will wait to delete a Purchase object after the purchase has expired and been billed for (as indicated by the Expired and Billed status boxes checked on the Purchase object). The default is 1 day. Enter 0 to have Purchase objects deleted immediately upon expiration and billing. Enter -1 to never delete expired Purchase objects.

**DeleteUsagePostExpirationDays:** Specifies how long the Maintenance process will wait to delete a Usage object after it has expired. The default is 1 day. Enter 0 to have Usage objects deleted immediately upon expiration. Enter -1 to never delete expired Usage objects.

**DeletePendingRequestExpirationDays:** Specifies how long the Maintenance process will wait to delete a Purchase object created for a purchase that is still pending approval or denial. The default is 1 day. Enter 0 to have pending Purchase objects deleted immediately. Enter -1 to never delete pending Purchase objects.

**DeleteRequestInfoExpirationDays:** This setting applies to subscription-based purchases only. It specifies how long the Maintenance process will wait to delete the purchase request information stored in the Purchase object. The default is 1 day. Enter 0 to have purchase request information deleted immediately upon approval or denial. Enter -1 to have purchase request information deleted only when the Purchase object is deleted.

**baseContext:** Specifies the context of the container that the Maintenance process will use as its root container. The Maintenance process will only process purchases located within the root container and its subcontainers. Use LDAP syntax to specify the root container (for example, ou=apps,ou=services,o=novell).

**purchaseClassx:** The Maintenance process is multi-threaded. One thread is used to process purchases (Purchase objects) and another thread is used to process usages (Usage objects). You can use the purchaseClassx setting to specify the Java classes that you want run during the purchase thread. Currently, there are two pre-defined classes that are used:

- ♦ RequestMaintain: Used to process request information generated by the ApprovalFlow™ process. If you don't use ApprovalFlow, you don't need to have the Maintenance process run this class.
- ♦ PurchaseMaintain: Used to process Purchase objects.

The order in which you specify the classes in the commerceserver.cfg file is the order in which they will be processed. In the following example, the purchaseClass1 entry is processed before the purchaseClass2 entry:

```
purchaseClass1 = RequestMaintain
purchaseClass2 = PurchaseMaintain
```

You can also develop your own Java classes and tie them in through the purchaseClassx property. For example, if you have a third-party billing or reporting system, you could write a Java class to read the Purchase object attributes and export them to the billing or reporting database. By entering the following property definition, the Maintenance process would run the Java class after the first two classes:

```
purchaseClass3 = PurchaseInfoExport
```

**usageClassx:** Specifies the Java classes that you want run during the Maintenance processes usage thread. Currently, there is one predefined class that is used:

- ♦ UsageMaintain: Used to process Usage objects.

The syntax is:

```
usageClass1 = UsageMaintain
```

As with the purchaseClassx property, you can tie in additional classes to perform tasks on Usage objects.

## Modifying Logging Settings

The Maintenance and Usage Server processes use Jakarta Log4j by the Apache Software Foundation for logging. Each process requires its own log configuration file, although each configuration file contains the same settings.

The files, log4j\_maintenance.cfg and log4j\_usage.cfg, are located in the \zenworks\ondemand directory on the same server volume as the Maintenance and UsageServer processes. Using the configuration settings in the files, you can set the logging information level (INFO, WARN, ERROR, FATAL, and DEBUG) and determine the output method (screen, rolling log file, and so forth).

The files contain brief descriptions of the configuration settings. For more information about the configuration settings, see the [Jakarta Log4j Web site \(http://jakarta.apache.org/log4j/docs/documentation\)](http://jakarta.apache.org/log4j/docs/documentation).



# V

## Appendixes

Appendix A, “Gadget Configuration Settings,” on page 127

Appendix B, “OnDemand Admin User Rights,” on page 133

Appendix C, “Maintaining User Access to the Portal During Upgrade,” on page 135



# A

## Gadget Configuration Settings

Each Novell® ZENworks® Web Self-Service gadget has configuration settings that you can modify to change the behavior of the gadget. The following sections provide descriptions of each gadget's configuration settings:

- ♦ [“Approval Gadget \(OD\\_ApprovalGadget\)” on page 128](#)
- ♦ [“Configuration Wizard Gadget \(OD\\_ConfigWizardGadget\)” on page 128](#)
- ♦ [“Current Budget Holder Gadget \(OD\\_CurrentBHGadget\)” on page 129](#)
- ♦ [“Edit Account Gadget \(OD\\_EditAccountGadget\)” on page 129](#)
- ♦ [“Launch Item Gadget \(OD\\_LaunchItemGadget\)” on page 129](#)
- ♦ [“Package Request Gadget \(OD\\_PackageRequestGadget\)” on page 130](#)
- ♦ [“Report Gadget \(OD\\_ReportGadget\)” on page 130](#)
- ♦ [“User Administration Gadget \(OD\\_UserAdminGadget\)” on page 131](#)
- ♦ [“Workflow Tracking Gadget \(OD\\_WorkflowTrackingGadget\)” on page 132](#)

## Accessing Gadget Configuration Settings

There are two places that you can modify gadget configuration settings: the gadget assignment or the gadget object.

A gadget assignment is simply a gadget that has been assigned to a page. The configuration settings you enter on a gadget assignment apply only to that assignment.

A gadget object is the base gadget before you assign it to a page. If you configure a gadget object's settings, the configuration settings remain with it when you assign the gadget to a page.

- ♦ [“Configuring Gadget Assignment Settings” on page 127](#)
- ♦ [“Configuring Gadget Object Settings” on page 128](#)

## Configuring Gadget Assignment Settings

The portal installation creates four pages (Cost Center Management, Purchase Approval, Applications, and OnDemand Services) and assigns the appropriate gadgets to the pages. You can modify the gadget assignment settings on all pages except the OnDemand Services page. To modify the configuration settings for the assigned gadgets assigned to the OnDemand Services page, you must modify the gadget objects' settings.

To configure the settings for a gadget assignment:

- 1** Log in to the portal as a portal administrator.
- 2** Click Administer the Portal.

- 3** Click Pages.
- 4** Select the page with the gadget assignment whose settings you want to configure, then click Edit.
- 5** Select the gadget assignment, then click Edit.

## Configuring Gadget Object Settings

- 1** Log in to the portal as a portal administrator.
- 2** Click Administer the Portal.
- 3** Click Gadgets.
- 4** Select the gadget whose configuration settings you want to modify, then click Edit.
- 5** Click Configuration.

## Approval Gadget (OD\_ApprovalGadget)

The Approval Gadget is used by budget holders to approve or deny users' requests to purchase packages. It has the following configuration settings:

**Display Name:** Specifies the name that will be used when displaying the gadget on the portal page. This field is available only if you are editing a gadget assignment rather than the gadget object.

**OnDemandService Object:** Specifies the distinguished name of the OnDemandService object that was created in Novell eDirectory™ during installation. Use LDAP syntax to specify the name (for example, cn=ondemandservice,ou=services,o=novell).

**Include URL in Email:** Specifies whether or not to include the portal's URL in notification messages sent to users. Including the URL allows users to launch the portal directly from the e-mail message rather than going to their Web browsers. The default setting is TRUE.

**iChain URL:** If you are using Novell iChain® to provide secure authentication and access to your portal, enter the appropriate iChain URL, including the protocol (for example, http://www.novell.com/nps or https://123.456.78.910). Instead of the portal's URL, this iChain URL will be used in notification messages sent to users.

**Window Height:** Specifies the height of the window that lists the purchase requests. If the list of requests is too long to fit in the window, the window will include a vertical scroll bar. The default setting is 172.

## Configuration Wizard Gadget (OD\_ConfigWizardGadget)

The Configuration Wizard gadget stores the OnDemand Admin User object's context and password in a secure place where the other gadgets can get it in order to authenticate to eDirectory. You only need to use this gadget if you change the location or password of the OnDemand Admin User object. It has the following configuration settings:

**Display Name:** Specifies the name that will be used when displaying the gadget on the portal page. This field is available only if you are editing a gadget assignment rather than the gadget object.



## Current Budget Holder Gadget (OD\_CurrentBHGadget)

The Current Budget Holder gadget is used by budget holders to select the currently active budget holder for the cost center and to add or remove budget holders. It uses the following settings:

**Display Name:** Specifies the name that will be used when displaying the gadget on the portal page. This field is available only if you are editing a gadget assignment rather than the gadget object.

**OnDemandService Object:** Specifies the distinguished name of the OnDemandService object that was created in eDirectory during installation. Use LDAP syntax to specify the name (for example, cn=ondemandservice,ou=services,o=novell).

**Window Height:** Specifies the height of the window that lists the budget holders. If the budget holder list is too long to fit in the window, the window will include a vertical scroll bar. The default setting is 172.

## Edit Account Gadget (OD\_EditAccountGadget)

The Edit Account gadget is no longer supported. In previous versions, this gadget allowed users to edit their account information (for example, name, password, and e-mail address). It is only included to support ZENworks OnDemand Services 2 customers.

## Launch Item Gadget (OD\_LaunchItemGadget)

The Launch Item gadget displays the applications from purchased packages and launches the applications. It has the following configuration settings:

**Display Name:** Specifies the name that will be used when displaying the gadget on the portal page. This field is available only if you are editing a gadget assignment rather than the gadget object.

**Window Height:** Specifies the height of the window that lists the applications. If the application list is too long to fit in the window, the window will include a vertical scroll bar. The default setting is 150.

**XTier Authentication Host Server:** Specifies the IP address or DNS host name of the server where the ZENworks for Desktops (ZfD) Middle Tier server is installed. The ZfD Middle Tier servers enables access to Novell eDirectory without the Novell Client™. This setting will override (and change) the user's current Middle Tier server configuration setting.

**Novell Client Authentication Host Server:** Specifies the IP address or DNS host name of the server to be used when authenticating to eDirectory through the Novell Client. This setting will override (and change) the user's current Novell Client configuration setting.

**XTier Port:** Specifies the number of the Middle Tier server. This setting will override (and change) the user's current Middle Tier server configuration setting.

**UsageServer:** Specifies the IP address or DNS host name of the network server where the UsageServer process is installed.

**Display Thin-Client Applications:** Specifies whether or not to display DeFrame thin-client applications in the list. The default is TRUE.

**Display Web Content:** Specifies whether or not to display Web applications (or Web content) in the list. The default is TRUE.

**Display Workstation Applications:** Specifies whether or not to display desktop applications in the list. The default is TRUE.

**Display Large Icons:** Specifies whether application icons are displayed in the Windows small icon format or in large icon format. The default is FALSE, which means small icon format is used.

**Number of Columns:** This setting applies only if Display Large Icons is set to TRUE. It specifies the number of columns to use when displaying the icons. The default is 4.

**iFolder Server Object:** If you are using Novell iFolder® to enable users to save files from thin-client applications to network storage locations, this setting specifies the iFolder servers you want available to users. Use typeless, leading dot delimited notation to specify the iFolder Server object's distinguished name (for example, .ifolderserver.services.novell). You can click New Value to specify more than one iFolder server.

This setting applies only if iFolder servers are not already associated with the user (User object > DeFrame tab > iFolder Servers page). If the user is already associated with iFolder servers, those servers are used.

## Package Request Gadget (OD\_PackageRequestGadget)

The Package Request gadget displays available application packages and enables users to request the packages. It has the following configuration settings:

**Display Name:** Specifies the name that will be used when displaying the gadget on the portal page. This field is available only if you are editing a gadget assignment rather than the gadget object.

**OnDemandService Object:** Specifies the distinguished name of the OnDemandService object that was created in eDirectory during installation. Use LDAP syntax to specify the name (for example, cn=ondemandservice,ou=services,o=novell).

**Include URL in Email:** Specifies whether or not to include the portal's URL in notification messages sent to budget holders after users request packages.

**iChain URL:** If you are using Novell iChain to provide secure authentication and access to your portal, enter the appropriate iChain URL, including the protocol (for example, http://www.novell.com/nps or https://123.456.78.910). Instead of the portal's URL, this iChain URL will be used in budget holder notification messages.

**Window Height:** Determines the height of the window used to list available packages. If the package list is too long to fit in the window, the window will include a vertical scroll bar. The default setting is 150.

**Limit Package Availability:** The default setting, FALSE, causes all packages to be available to users. Setting this to TRUE causes users to have access only to the packages that they have specifically been given rights to (through eDirectory). If you set this to TRUE, you need to configure user rights to packages, otherwise users will not have any packages available to them. For instructions, see [Chapter 12, "Limiting Package Availability," on page 87](#).

## Report Gadget (OD\_ReportGadget)

The Report gadget is used to create reports detailing the charges accrued by users for package they've purchased. It has the following configuration settings:

**Display Name:** Specifies the name that will be used when displaying the gadget on the portal page. This field is available only if you are editing a gadget assignment rather than the gadget object.

**ASP Admin:** Specifies the distinguished names of the users who can generate reports for all companies located in the eDirectory tree. Use LDAP syntax to specify a name (for example, cn=sgreen,o=novell). You can click New Value to add more than one user. Make sure the user contexts you enter are valid. The contexts will not be checked during entry for validity.

**Company Admin:** Specifies the distinguished names of the users who can generate reports for all cost centers in a single company. Use LDAP syntax to specify a name (for example, cn=jsmith,ou=services,o=novell). You can click New Value to add more than one user. Make sure the user contexts you enter are valid. The contexts will not be checked during entry for validity.

**OnDemandService Object:** Specifies the distinguished name of the OnDemandService object that was created in eDirectory during installation. Use LDAP syntax to specify the name (for example, cn=ondemandservice,ou=services,o=novell).

**Window Height:** Specifies the height of the window that lists the packages included in the report. If the package list is too long to fit in the window, the window will include a vertical scroll bar. The default setting is 200.

## Tabbed Container Gadget (OD\_TabbedContainerGadget)

The Tabbed Container gadget is used to create the OnDemand Services page. It has no gadget specific configuration settings.

## User Administration Gadget (OD\_UserAdminGadget)

The User Administration gadget is used by budget holders to assign packages to users. It has the following configuration settings:

**Display Name:** Specifies the name that will be used when displaying the gadget on the portal page. This field is available only if you are editing a gadget assignment rather than the gadget object.

**Allow Users to Unsubscribe:** This setting is no longer used. In previous versions, it populated the user's Disable User's Ability to Unsubscribe Package's setting in eDirectory (User object > OnDemand tab > Purchases page > Disable User's Ability to Unsubscribe Packages). It is only included to support ZENworks OnDemand Services 2 customers.

**Include URL in Email:** This setting applies when assigning packages to users. It specifies whether or not to include the portal's URL in e-mail notifications sent to users. Notification messages let users know when they have been given access to packages. Including the URL in the messages enables users to launch the portal directly from the messages rather than going to their Web browsers.

**iChain URL:** This setting applies when assigning packages to users. If you are using Novell iChain to provide secure authentication and access to your portal, enter the appropriate iChain URL, including the protocol (for example, <http://www.novell.com/nps> or <https://123.456.78.910>). Instead of the portal's URL, this iChain URL will be used in notification messages sent to users.

**Window Height:** Specifies the height of the window that lists the users. If the user list is too long to fit in the window, the window will include a vertical scroll bar. The default setting is 200.

**iFolder Server Object:** This setting applies when creating new users through the User Administration gadget. If you are using Novell iFolder to enable users to save files from thin-client applications to network storage locations, this setting specifies the iFolder servers you want available to users. You must enter the distinguished name of the iFolder Server object, in LDAP syntax:

```
cn=ifolderserver,ou=services,o=novell
```

This setting populates the user's iFolder Servers setting in eDirectory (User object > DeFrame tab > iFolder Servers page > iFolder Servers setting).

**Available Templates:** This setting applies when creating new users through the User Administration gadget. You can use this setting to specify User templates that budget holders can choose from when creating user accounts. You must specify the distinguished name of the Template object, in LDAP syntax:

```
cn=usertemplate,ou=users,o=novell
```

## Workflow Tracking Gadget (OD\_WorkflowTrackingGadget)

The Workflow Tracking gadget is used by users to check the status of their package purchase requests and to cancel requests. It has the following configuration settings:

**Display Name:** Specifies the name that will be used when displaying the gadget on the portal page. This field is available only if you are editing a gadget assignment rather than the gadget object.

**OnDemandService Object:** Specifies the distinguished name of the OnDemandService object that was created in eDirectory during installation. Use LDAP syntax to specify the name (for example, cn=ondemandservice,ou=services,o=novell).

**Window Height:** Specifies the height of the window that lists the user's current requests. If the request list is too long to fit in the window, the window will include a vertical scroll bar. The default setting is 150.

# B

## OnDemand Admin User Rights

The Novell® ZENworks® Web Self-Service components authenticate to Novell eDirectory™ through the OnDemand Admin User object. The table below summarizes all trustee rights required by the User object.

During installation, the OnDemand Admin User object is assigned Supervisor rights to the container where it resides. As long as the objects listed in the following table reside in or beneath the OnDemand Admin User object's container, the OnDemand Services gadgets will have the eDirectory rights they need.

If all objects do not reside beneath the OnDemand Admin User object's container, you can select a base container that encompasses all of the objects and give the OnDemand Admin User object Supervisor rights to the container. Or, you can use the following table to assign appropriate rights to individual objects or containers.

Object	Trustee Rights	Explanation
Company root container	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read	The company root container (as defined on the OnDemandService object) and all subcontainers.
User containers	[Entry Rights] - Browse, Create [All Attributes Rights] - Compare, Read, Write	All containers where User objects reside.
Catalog container	[Entry Rights] - Browse, Create [All Attributes Rights] - Compare, Read, Write	The catalog root container (as defined on the OnDemandService object) and all subcontainers.
Application objects containers	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read, Write	All containers where Application objects (desktop and DeFrame thin-client) reside.
Commerce Item objects containers	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read	All containers where Commerce Item (Web application) objects reside.
iChain ACL Rule objects	[Entry Rights] - Browse, Create [All Attributes Rights] - Compare, Read, Write	All iChain ACL rules that control access to OnDemand Services Web applications.
OnDemandService object	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read	
Cost Center objects	[Entry Rights] - Browse [All Attributes Rights] - Compare, Read, Write	All cost center (Organizational Role) objects



# C

## Maintaining User Access to the Portal During Upgrade

The following sections provide information about how to minimize user impact by maintaining user access to your current portal during the upgrade process. This information should be used in combination with the upgrade instructions provided in [Chapter 8, “Upgrading to Web Self-Service,” on page 57](#).

- ♦ [“Maintaining Access to a Windows Portal Server” on page 135](#)
- ♦ [“Maintaining Access to a NetWare Portal Server” on page 135](#)

### Maintaining Access to a Windows Portal Server

For a Novell® Portal Services 1.5 portal running on a Windows server, the following options are available to maintain user access to the portal during the upgrade:

- ♦ If your portal installation includes multiple portal heads (servers), bring down one or more of the portal heads, upgrade those heads, bring the upgraded portal heads online, then upgrade the remaining portal heads. To upgrade a portal head, follow the instructions in [“Upgrading the Portal Software” on page 57](#).
- ♦ If your portal installation includes only one portal head (server), install the portal files to a new server. To do so, follow the instructions in [“Upgrading the Portal Software” on page 57](#). After your new portal is configured, remove the old portal server and use only the new portal server.

For information about managing multiple portal heads, see the [Novell Portal Services 1.5 documentation \(http://www.novell.com/documentation/lg/portal\)](#) and the [Novell exteNd Director™ 4.1 Standard Edition documentation \(http://www.novell.com/documentation/lg/nedse41\)](#).

**IMPORTANT:** DeFrame™ thin-client applications will not work in the new portal until you upgrade DeFrame to ZfD 4.0.1 DeFrame. This is the version included with ZENworks 6 Desktop Management. After you upgrade to ZfD 4.0.1 DeFrame, DeFrame thin-client applications will not work in your old portal, so you should not upgrade to ZfD 4.0.1 DeFrame until you are ready to start using the new portal.

### Maintaining Access to a NetWare Portal Server

For a Novell® Portal Services 1.5 portal running on a NetWare server, the following options are available to maintain user access to the portal during the upgrade:

- ♦ If your portal installation includes multiple portal heads (servers), bring down one or more of the portal heads, upgrade those heads, bring the upgraded portal heads online, then upgrade the remaining portal heads. To upgrade a portal head, follow the instructions in [“Upgrading the Portal Software” on page 57](#).

For information about managing multiple portal heads, see the [Novell Portal Services 1.5 documentation \(http://www.novell.com/documentation/lg/portal\)](#) and the [Novell exteNd](#)

Director™ 4.1 Standard Edition documentation (<http://www.novell.com/documentation/lg/nedse41>).

- ♦ If your portal installation includes only one portal head (server), use one of the following methods:
  - ♦ Install the portal files to a new server that meets the software requirements listed in “[Web Server Requirements](#)” on page 53. To do so, follow the instructions in “[Upgrading the Portal Software](#)” on page 57. After your new portal is configured, remove the old portal server and use only the new portal server.
  - ♦ Install the portal files to the existing portal server. Because you will be managing two Tomcat installations on the server, detailed instructions are provided in [Single-Server Upgrade Procedures](#) below.

**IMPORTANT:** DeFrame™ thin-client applications will not work in the new portal until you upgrade DeFrame to ZfD 4.0.1 DeFrame. This is the version included with ZENworks 6 Desktop Management. After you upgrade to ZfD 4.0.1 DeFrame, DeFrame thin-client applications will not work in your old portal, so you should not upgrade to ZfD 4.0.1 DeFrame until you are ready to start using the new portal.

## Single-Server Upgrade Procedures

To maintain access to a portal while you upgrade the portal, follow the instructions in the following sections:

- ♦ “[Upgrading the Server Software](#)” on page 136
- ♦ “[Upgrading the Portal Software](#)” on page 136
- ♦ “[Establishing Parallel Portals](#)” on page 136
- ♦ “[Finishing the Upgrade](#)” on page 137
- ♦ “[Switching to the New Portal](#)” on page 138

### Upgrading the Server Software

Upgrade the server software to meet the requirements listed in “[Web Server Requirements](#)” on page 53. When you upgrade to Tomcat 4, do not overwrite the current Tomcat 3.3 directory. Make a new directory at the same level as the Tomcat 3.3 directory (for example, `sys:\tomcat\33` and `sys:\tomcat\4`).

### Upgrading the Portal Software

Follow the instructions in “[Upgrading the Portal Software](#)” on page 57 to upgrade the portal software.

### Establishing Parallel Portals

After upgrading the portal software, complete the following steps to configure user access to the old portal and administrator access to the new portal. After you complete the steps, users will still have access to the old portal through the same URL they’ve always used and you’ll have access to the new portal by specifying port 8080 in the URL.

- 1** If you are currently logged in to the new Novell exteNd Director 4.1 portal, exit the portal.
- 2** Open the `Apache\Conf\AdminServ.conf` file and do the following:
  - 2a** Make sure the NPS 1.5 Tomcat configuration information is not commented out.



The NPS 1.5 configuration information might be contained in a NOVELL PORTAL SERVICES section in the file. Or, it might be in another file that is referenced with an Include statement similar to the following:

```
#Include Tomcat33_root_directory_path\conf\nps-apache.conf
```

- 2b** Comment out the following line, located at the end of the file, so that the

```
#Include Tomcat4_root_directory_path\conf\nps-apache.conf
```

- 3** In the *tomcat33\conf\server.xml* file, change the listener for port 8080 to another port number (for example, 8081).

This enables Tomcat 4 to listen on port 8080. Tomcat33 will listen on the new port assignment.

- 4** In the server's *autoexec.ncf* file, include search path and load commands for both Tomcat 3.3 and Tomcat 4. For example:

```
search add sys:\tomcat33\bin
tomcat33
search add sys:\tomcat4\bin
tomcat4
```

- 5** At the system console prompt, enter **nvxadmdn** to stop the Apache Web server, then enter **nvxadmup** to restart it.

- 6** Stop Tomcat 3.3 and Tomcat 4.

You can unload Java to stop both versions, or you can use the following command for each version:

```
java -killxxx
```

The *xxx* variable indicates the process ID. You can discover the process ID by entering `java -show` at the system console prompt.

- 7** At the system console prompt, enter **tomcat33** to restart Tomcat 3.3, then enter **tomcat4** to restart Tomcat 4.

You should now be able to access both your existing NPS 1.5 portal and your new exteNd Director 4.1 portal using the following URLs:

```
(Existing NPS 1.5 portal) http://server_address/nps
(New exteNd Director 4.1 portal) http://server_address:8080/nps
```

The *server\_address* variable represents the IP address or DNS hostname of the portal server. Designating port 8080 in the URL for the exteNd Director 4.1 portal bypasses the Apache Web server, which is handling the NPS 1.5 portal, and routes to Tomcat 4, which is handling the exteNd Director 4.1 portal.

## Finishing the Upgrade

To finish configuring the new portal, complete the remaining tasks in “[Upgrading the Portal Software](#)” on page 57.

To upgrade the UsageServer and Maintenance processes, complete the tasks in “[Upgrading the Usage Software \(Optional\)](#)” on page 66 and “[Upgrading the Maintenance Software \(Optional\)](#)” on page 69.

## Switching to the New Portal

After you've completed the upgrade tasks and are ready to switch users to the new portal, complete the following steps:

- 1** In the `Apache\Conf\AdminServ.conf` file, remove the NPS configuration information and uncomment the following line located at the end of the file:

```
Include Tomcat4_root_directory_path\conf\nps-apache.conf
```

- 2** Restart the Apache Web server.

When users access the portal, they will get the new portal rather than the old portal.