Oracle® Workflow

Administrator's Guide

Release 12

Part No. B31431-01

December 2006

ORACLE®

Oracle Workflow Administrator's Guide, Release 12

Part No. B31431-01

Contents

Send Us Your Comments

Preface

1 Overview of Oracle Workflow

2 Setting Up Oracle Workflow

3   Oracle Workflow Security

4   Oracle Workflow Home Page

5   Monitoring Workflow Processes

6 Administering Notifications

7 Oracle Workflow Manager

8 Oracle Workflow Loaders

9 Workflow Administration Scripts

A Oracle Workflow Administrator Navigation Paths

B Oracle Workflow Administrator Personalizations

C Oracle Workflow Performance Concepts

D   Oracle Workflow Profile Options

E   Oracle Workflow Diagnostic Tests

Glossary

Index

Send Us Your Comments

Oracle Workflow Administrator's Guide, Release 12

Part No. B31431-01

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document. Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Applications Release Online Documentation CD available on Oracle MetaLink and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: appsdoc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Preface

Intended Audience

Welcome to Release 12 of the *Oracle Workflow Administrator's Guide*.

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.

- The Oracle Applications graphical user interface.

    To learn more about the Oracle Applications graphical user interface, read the *Oracle Applications User's Guide*.

- Operating system concepts.

- Oracle Database, Oracle Application Server, and PL/SQL technology.

    If you have never used these products, Oracle suggests you attend training classes available through Oracle University.

See Related Information Sources on page xiii for more Oracle Applications product information.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our

documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/ .

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Structure

1 Overview of Oracle Workflow

This chapter introduces you to the concept of a workflow process and to the major features of Oracle Workflow.

2 Setting Up Oracle Workflow

This chapter describes the requirements for Oracle Workflow and the steps necessary to set up Oracle Workflow at your site.

3 Oracle Workflow Security

This chapter describes the architecture and configuration of security for Oracle Workflow.

4 Oracle Workflow Home Page

This chapter discusses the Oracle Workflow home page, where administrators can centrally access the Web-based features of Oracle Workflow.

5 Monitoring Workflow Processes

This chapter discusses how to monitor an instance of a workflow process.

6 Administering Notifications

This chapter describes how to administer users' notifications.

7 Oracle Workflow Manager

This chapter describes how to use the Oracle Workflow Manager component of Oracle Applications Manager.

8 Oracle Workflow Loaders

This chapter describes access protection for workflow object definitions and how to load those definitions between a database and a flat file.

9 Workflow Administration Scripts

This chapter describes the SQL scripts that workflow administrators can run against an Oracle Workflow server installation.

A Oracle Workflow Administrator Navigation Paths

This appendix lists the navigation paths to Oracle Workflow administrator Web pages in the seeded Oracle Workflow responsibilities for Oracle Applications.

B Oracle Workflow Administrator Personalizations

This appendix lists features that you can add to Oracle Workflow administrator Web pages through Oracle Application Framework Personalization.

C Oracle Workflow Performance Concepts

This appendix describes concepts and techniques that you can use to enhance performance when running Oracle Workflow.

D Oracle Workflow Profile Options

This appendix lists the profile options that you can set to configure Oracle Workflow.

E Oracle Workflow Diagnostic Tests

This appendix describes the diagnostic tests that workflow administrators can run to check the setup of Oracle Workflow.

Glossary

Related Information Sources

This book is included on the Oracle Applications Documentation Library, which is supplied in the Release 12 Media Pack. You can download soft-copy documentation as PDF files from the Oracle Technology Network at http://otn.oracle.com/documentation, or you can purchase hard-copy documentation from the Oracle Store at http://oraclestore.oracle.com. The Oracle Applications Release 12 Documentation Library contains the latest information, including any documents that have changed significantly between releases. If substantial changes to this book are necessary, a revised version will be made available on the "virtual" documentation library on Oracle *MetaLink*.

For a full list of documentation resources for Oracle Applications Release 12, see *Oracle Applications Documentation Resources, Release 12*, Oracle*MetaLink* Document 394692.1.

If this guide refers you to other Oracle Applications documentation, use only the Release 12 versions of those guides.

Online Documentation

All Oracle Applications documentation is available online (HTML or PDF).

- **Online Help** - Online help patches (HTML) are available on Oracle*MetaLink*.

- **PDF Documentation** - See the Oracle Applications Documentation Library for

current PDF documentation for your product with each release. The Oracle Applications Documentation Library is also available on Oracle*MetaLink* and is updated frequently.

- **Oracle Electronic Technical Reference Manual -** The Oracle Electronic Technical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for each Oracle Applications product. This information helps you convert data from your existing applications and integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. The Oracle eTRM is available on Oracle *MetaLink*.

Related Guides

You should have the following related books on hand. Depending on the requirements of your particular installation, you may also need additional manuals or guides.

Oracle Alert User's Guide

This guide explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

Oracle Applications Concepts

This book is intended for all those planning to deploy Oracle E-Business Suite Release 12, or contemplating significant changes to a configuration. After describing the Oracle Applications architecture and technology stack, it focuses on strategic topics, giving a broad outline of the actions needed to achieve a particular goal, plus the installation and configuration choices that may be available.

Oracle Applications Developer's Guide

This guide contains the coding standards followed by the Oracle Applications development staff. It describes the Oracle Application Object Library components needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. It also provides information to help you build your custom Oracle Forms Developer forms so that they integrate with Oracle Applications.

Oracle Application Framework Developer's Guide

This guide contains the coding standards followed by the Oracle Applications development staff to produce applications built with Oracle Application Framework. This guide is available in PDF format on Oracle*MetaLink* and as online documentation in JDeveloper 10*g* with Oracle Application Extension.

Oracle Application Framework Personalization Guide

This guide covers the design-time and run-time aspects of personalizing applications built with Oracle Application Framework.

Oracle Applications Installation Guide: Using Rapid Install

This book is intended for use by anyone who is responsible for installing or upgrading

Oracle Applications. It provides instructions for running Rapid Install either to carry out a fresh installation of Oracle Applications Release 12, or as part of an upgrade from Release 11*i* to Release 12. The book also describes the steps needed to install the technology stack components only, for the special situations where this is applicable.

Oracle Applications Supportability Guide

This manual contains information on Oracle Diagnostics and the Logging Framework for system administrators and custom developers.

Oracle Applications System Administrator's Guide Documentation Set

This documentation set provides planning and reference information for the Oracle Applications System Administrator. *Oracle Applications System Administrator's Guide - Configuration* contains information on system configuration steps, including defining concurrent programs and managers, enabling Oracle Applications Manager features, and setting up printers and online help. *Oracle Applications System Administrator's Guide - Maintenance* provides information for frequent tasks such as monitoring your system with Oracle Applications Manager, managing concurrent managers and reports, using diagnostic utilities, managing profile options, and using alerts. *Oracle Applications System Administrator's Guide - Security* describes User Management, data security, function security, auditing, and security configurations.

Oracle Applications User's Guide

This guide explains how to navigate, enter data, query, and run reports using the user interface (UI) of Oracle Applications. This guide also includes information on setting user profiles, as well as running and reviewing concurrent requests.

Oracle Integration Repository User's Guide

This guide covers the employment of Oracle Integration Repository in researching and deploying business interfaces to produce integrations between applications.

Oracle Workflow Developer's Guide

This guide explains how to define new workflow business processes and customize existing Oracle Applications-embedded workflow processes. It also describes how to define and customize business events and event subscriptions.

Oracle Workflow User's Guide

This guide describes how users can view and respond to workflow notifications and monitor the progress of their workflow processes.

Oracle Workflow API Reference

This guide describes the APIs provided for developers and administrators to access Oracle Workflow.

Oracle XML Gateway User's Guide

This guide describes Oracle XML Gateway functionality and each component of the Oracle XML Gateway architecture, including Message Designer, Oracle XML Gateway Setup, Execution Engine, Message Queues, and Oracle Transport Agent. The

integrations with Oracle Workflow Business Event System and the Business-to-Business transactions are also addressed in this guide.

Integration Repository

The Oracle Integration Repository is a compilation of information about the service endpoints exposed by the Oracle E-Business Suite of applications. It provides a complete catalog of Oracle E-Business Suite's business service interfaces. The tool lets users easily discover and deploy the appropriate business service interface for integration with any system, application, or business partner.

The Oracle Integration Repository is shipped as part of the E-Business Suite. As your instance is patched, the repository is automatically updated with content appropriate for the precise revisions of interfaces in your environment.

Do Not Use Database Tools to Modify Oracle Applications Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using an Oracle Applications form can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

1

Overview of Oracle Workflow

This chapter introduces you to the concept of a workflow process and to the major features of Oracle Workflow.

This chapter covers the following topics:

- Overview of Oracle Workflow for Administrators

Overview of Oracle Workflow for Administrators

Oracle Workflow delivers a complete workflow management system that supports business process based integration. Its technology enables modeling, automation, and continuous improvement of business processes, routing information of any type according to user-defined business rules.

E-business is accelerating the demand for integration of applications within the enterprise as well as integration of a company's systems with trading partners and business-to-business exchanges. Oracle Workflow automates and streamlines business processes both within and beyond your enterprise, supporting traditional applications based workflow as well as e-business integration workflow. Oracle Workflow is unique in providing a workflow solution for both internal processes and business process coordination between applications.

Routing Information

Business processes today involve getting many types of information to multiple people according to rules that are constantly changing. With so much information available, and in so many different forms, how do you get the right information to the right people? Oracle Workflow lets you provide each person with all the information they need to take action. Oracle Workflow can route supporting information to each decision maker in a business process, including people both inside and outside your enterprise.

Defining and Modifying Business Rules

Oracle Workflow lets you define and continuously improve your business processes using a drag-and-drop process designer.

Unlike workflow systems that simply route documents from one user to another with some approval steps, Oracle Workflow lets you model sophisticated business processes. You can define processes that loop, branch into parallel flows and then rendezvous, decompose into subflows, and more. Because Oracle Workflow can decide which path to take based on the result of a stored procedure, you can use the power of Java and of PL/SQL, the language of the Oracle Database, to express any business rule that affects a workflow process. See: Workflow Processes, page 1-4.

Delivering Electronic Notifications

Oracle Workflow extends the reach of business process automation throughout the enterprise and beyond to include any e-mail or Internet user. Oracle Workflow lets people receive notifications of items awaiting their attention via e-mail, and act based on their e-mail responses. You can even view your list of things to do, including necessary supporting information, and take action using a standard Web browser.

Integrating Systems

Oracle Workflow lets you set up subscriptions to business events which can launch workflows or enable messages to be propagated from one system to another when business events occur. You can communicate events among systems within your own enterprise and with external systems as well. In this way, you can implement point-to-point messaging integration or use Oracle Workflow as a messaging hub for more complex system integration scenarios. You can model business processes that include complex routing and processing rules to handle events powerfully and flexibly.

Major Features and Definitions

Oracle Workflow Builder

Oracle Workflow Builder is a graphical tool that lets you create, view, or modify a business process with simple drag and drop operations. Using the Workflow Builder, you can create and modify all workflow objects, including activities, item types, and messages. See: Workflow Processes, page 1-4.

At any time you can add, remove, or change workflow activities, or set up new prerequisite relationships among activities. You can easily work with a summary-level model of your workflow, expanding activities within the workflow as needed to greater levels of detail. And, you can operate Oracle Workflow Builder from a desktop PC or from a disconnected laptop PC.

Workflow Engine

The Workflow Engine embedded in the Oracle Database implements process definitions at runtime. The Workflow Engine monitors workflow states and coordinates the routing of activities for a process. Changes in workflow state, such as the completion of workflow activities, are signaled to the engine via a PL/SQL API or a Java API. Based on flexibly-defined workflow rules, the engine determines which activities are eligible to run, and then runs them. The Workflow Engine supports sophisticated workflow rules, including looping, branching, parallel flows, and subflows.

Business Event System

The Business Event System is an application service that uses the Oracle Advanced Queuing (AQ) infrastructure to communicate business events between systems. The Business Event System consists of the Event Manager, which lets you register subscriptions to significant events, and event activities, which let you model business events within workflow processes.

When a local event occurs, the subscribing code is executed in the same transaction as the code that raised the event. Subscription processing can include executing custom code on the event information, sending event information to a workflow process, and sending event information to other queues or systems.

Workflow Definitions Loader

The Workflow Definitions Loader is a utility program that moves workflow definitions between database and corresponding flat file representations. You can use it to move workflow definitions from a development to a production database, or to apply upgrades to existing definitions. In addition to being a standalone server program, the Workflow Definitions Loader is also integrated into Oracle Workflow Builder, allowing you to open and save workflow definitions in both a database and file.

Complete Programmatic Extensibility

Oracle Workflow lets you include your own PL/SQL procedures or external functions as activities in your workflows. Without modifying your application code, you can have your own program run whenever the Workflow Engine detects that your program's prerequisites are satisfied.

Electronic Notifications

Oracle Workflow lets you include users in your workflows to handle activities that cannot be automated, such as approvals for requisitions or sales orders. The Notification System sends notifications to and processes responses from users in a workflow. Electronic notifications are routed to a role, which can be an individual user or a group of users. Any user associated with that role can act on the notification.

Each notification includes a message that contains all the information a user needs to

make a decision. The information may be embedded in the message body or attached as a separate document. Oracle Workflow interprets each notification activity response to decide how to move on to the next workflow activity.

Electronic Mail Integration

Electronic mail (e-mail) users can receive notifications of outstanding work items and can respond to those notifications using their e-mail application of choice. An e-mail notification can include an attachment that provides another means of responding to the notification.

Internet-Enabled Workflow

Any user with access to a standard Web browser can be included in a workflow. Web users can access a Notification Web page to see their outstanding work items, then navigate to additional pages to see more details or provide a response.

Monitoring and Administration

Workflow administrators and users can view the progress of a work item in a workflow process by connecting to the Workflow Monitor using a standard Web browser that supports Java. The Workflow Monitor displays an annotated view of the process diagram for a particular instance of a workflow process, so that users can get a graphical depiction of their work item status. The Workflow Monitor also displays a separate status summary for the work item, the process, and each activity in the process.

You can also use the Oracle Workflow Manager component of Oracle Applications Manager as an additional administration tool for Oracle Workflow. Oracle Applications Manager is a tool that provides administrative and diagnostic capabilities for concurrent processing, Oracle Workflow, and other functionality in Oracle Applications.

Workflow Processes

Oracle Workflow manages business processes according to rules that you define. The rules, which we call a workflow process definition, include the activities that occur in the process and the relationship between those activities. An activity in a process definition can be an automated function defined by a PL/SQL stored procedure or an external function, a notification to a user or role that may optionally request a response, a business event, or a subflow that itself is made up of a more granular set of activities.

A workflow process is initiated when an application calls a set of Oracle Workflow Engine APIs. The Workflow Engine takes over by driving the relevant work item defined by the application, through a specific workflow process definition. According to the workflow process definition, the Workflow Engine performs automated steps and invokes appropriate agents when external processing is required.

The following diagram depicts a simplified workflow process definition that routes a

requisition to a manager or set of managers for approval.

Sample Workflow Process in Oracle Workflow Builder
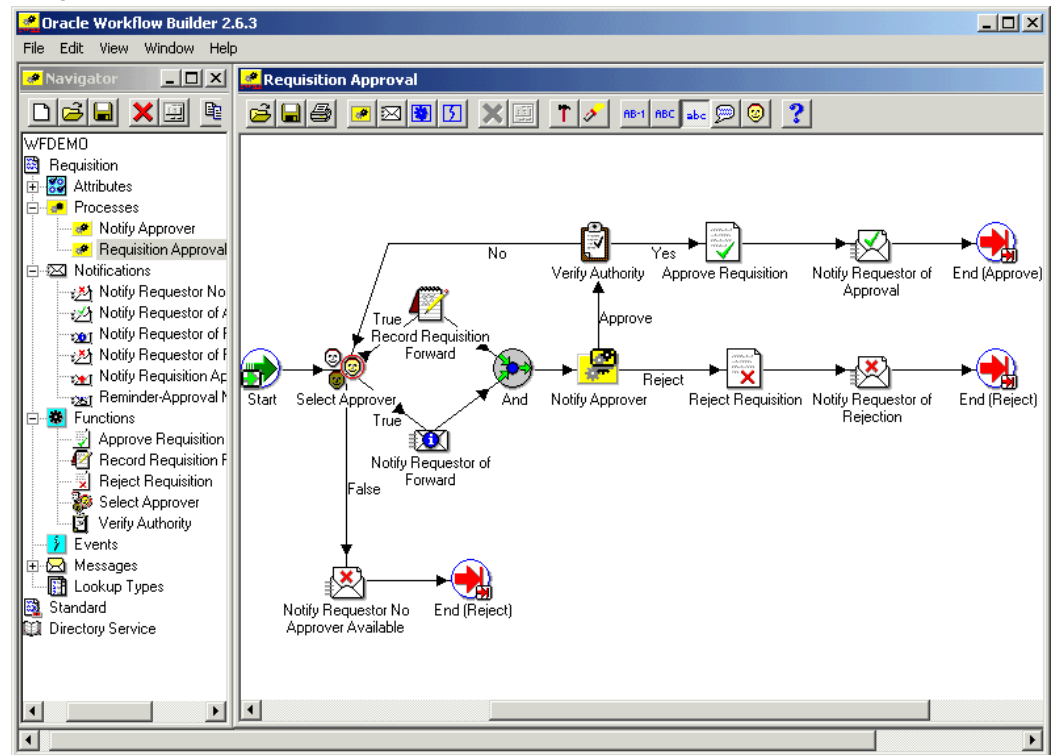
We refer to the whole drawing as a process or process diagram. The icons represent activities, and the arrows represent the transitions between the activities. In the above example, new items are created for the process when a user creates and submits a requisition in the appropriate application.

This process contains several workflow activities implemented as PL/SQL stored procedures, including:

- Select Approver - To select, according to your business rules, who should approve the requisition.

- Verify Authority - To verify that a selected approver has the spending authority to approve the requisition.

2

Setting Up Oracle Workflow

This chapter describes the requirements for Oracle Workflow and the steps necessary to set up Oracle Workflow at your site.

This chapter covers the following topics:

- Overview of Setting Up
- Step 1: Partitioning Workflow Tables
- Step 2: Setting Global User Preferences
- Step 3: Setting Up an Oracle Workflow Directory Service
- Step 4: Setting Up Additional Languages
- Step 5: Setting Up Background Workflow Engines
- Step 6: Implementing Notification Mailers
- Step 7: Modifying Your Message Templates
- Step 8: Adding Worklist Functions to User Responsibilities
- Step 9: Setting the WF: Notification Reassign Mode Profile Option
- Step 10: Setting Up Notification Handling Options
- Step 11: Setting Up for Electronic Signatures
- Step 12: Customizing the Logo on Oracle Workflow's Web Pages
- Step 13: Adding Custom Icons to Oracle Workflow
- Step 14: Setting Up the Business Event System

Overview of Setting Up

After you install Oracle Workflow, implement it for your site by setting up the preferences and components appropriate for your enterprise.

Related Topics

Oracle Workflow Hardware and Software Requirements, page 2-2

Overview of Required Setup Steps for Oracle Workflow, page 2-3

Optional Setup Steps, page 2-4

Other Workflow Features, page 2-5

Identifying the Version of Your Oracle Workflow Server, page 2-5

Oracle Workflow Setup Checklist, page 2-5

Oracle Workflow Hardware and Software Requirements

The components of Oracle Workflow require the following hardware and software configurations.

- Oracle Workflow Builder is installed using Oracle Universal Installer. The Oracle Workflow Builder installation includes the Oracle Net Services and Required Support Files which it requires. You should install Oracle Workflow Builder on an IBM, Compaq or 100% compatible personal computer with the following:

  - A Pentium processor or better

  - Clock speed of 66 Mhz or greater (90 Mhz or greater is recommended)

  - Network card

  - SVGA color monitor

  - Microsoft Windows 2000 or Windows XP

  - At least 65 Mb of available disk space to install Oracle Workflow Builder, Oracle Net Services, and Required Support Files.

  - At least 32 Mb of memory, 64 Mb recommended

    Note: Oracle Net Services require and only support the use of Microsoft's TCP/IP drivers.

- The Oracle Workflow Server requires the following:

  - Oracle Database 10*g* Release 2 or higher, Enterprise or Standard Edition. See: *Oracle Applications Installation Guide: Using Rapid Install*.

  - At least 40 Mb of available disk space for Oracle Workflow Server once it is installed in your Oracle Home

- At least 128 Mb of memory, 256 Mb recommended

- Oracle Net Services, corresponding to the version of your Oracle Database

- SQL*Plus, corresponding to the version of your Oracle Database

- To send and receive e-mail notifications, you must have an SMTP mail server set up for outbound messages and an IMAP4 compliant mail server set up for inbound messages.

  To connect to the SMTP server over Secure Sockets Layer (SSL), you must have Stunnel installed on the SMTP server.

- To send and respond to e-mail notifications with HTML attachments, your e-mail application should support HTML attachments and you should have a Web browser application that supports JavaScript and Frames to view the attachment.

- The Oracle Workflow Web pages require Oracle HTTP Server to be installed on a server machine. Oracle HTTP Server is included with Oracle Application Server. See: *Oracle Applications Installation Guide: Using Rapid Install*.

  To view Workflow Web pages, users need a Web browser application supported for Oracle Applications, such as Microsoft Internet Explorer version 6.0 or Mozilla Firefox version 1.5.

- To sign notification responses with certificate-based digital signatures, you must use a PC with Microsoft Windows 2000 or Windows XP. Additionally, you must access the notifications in the Oracle Workflow Web pages using Microsoft Internet Explorer version 6.0.

- To use the Workflow XML Loader, you must have Java Development Kit (JDK) Version 5.0 installed.

Overview of Required Setup Steps

1. Set up the default Oracle Workflow user preferences for your entire enterprise using the Workflow Configuration page. The Workflow Configuration page also lets you define your workflow administrator role. See: Setting Global User Preferences, page 2-9.

2. Ensure that a directory service is set up to provide information about the individuals and roles in your organization who may utilize Oracle Workflow functionality and receive workflow notifications. During installation of Oracle Applications, directory service views for users and roles from the unified Oracle Applications environment are automatically implemented for you. See: Setting Up an Oracle Workflow Directory Service, page 2-14.

3. Set up background Workflow Engines to control the load and throughput of the primary Workflow Engine on your system. You can specify the cost threshold level of your primary and background engines to determine the activities an engine processes and the activities an engine defers. See: Setting Up Background Workflow Engines, page 2-39.

4. Set up the Business Event System to communicate business events between systems using event subscription processing and Workflow process event activities. See: Setting Up the Business Event System, page 2-145.

Optional Setup Steps

1. You can partition the WF_ITEM_ACTIVITY_STATUSES, WF_ITEM_ACTIVITY_STATUSES_H, WF_ITEM_ATTRIBUTE_VALUES, and WF_ITEMS tables for performance gain. See: Partitioning Workflow Tables, page 2-8.

2. Set up additional languages if you want to use Oracle Workflow in languages other than English. See: Setting Up Additional Languages, page 2-36.

3. Set up one or more notification mailers if you want to allow your users to receive notifications by e-mail. See: Implementing Notification Mailers, page 2-43.

4. You can modify the templates for your electronic mail notifications. See: Modifying Your Message Templates, page 2-84.

5. You can give users access to the Advanced Worklist, Personal Worklist, and Notification Search Web pages from any responsibility you choose. See: Adding Worklist Functions to User Responsibilities, page 2-136.

6. You can use the WF: Notification Reassign Mode profile option to control which reassign modes are available to users from the Notification Details page. See: Setting the WF: Notification Reassign Mode Profile Option, page 2-137.

7. You can control the item types for which users can define vacation rules and grant worklist access, using the WF: Vacation Rule Item Types lookup type and the WF: Vacation Rules - Allow All profile option. See: Setting Up Notification Handling Options, page 2-138.

8. You can set up users to enable electronic signatures in notification responses. See: Setting Up for Electronic Signatures, page 2-140.

9. You can customize the company logo that appears in the Oracle Workflow Web pages. See: Customizing the Logo on Oracle Workflow's Web Pages, page 2-145.

10. You can include additional icons to your Oracle Workflow Icons subdirectory to

customize the diagrammatic representation of your workflow processes. Use custom symbols for each activity you define. See: Adding Custom Icons to Oracle Workflow, page 2-145.

Other Workflow Features

Before deploying Oracle Workflow and custom process definitions to other branches of your enterprise, you can protect your data from further modification by determining the level of access your users have to the data. See: Overview of Oracle Workflow Access Protection, page 8-1.

You can also use the Workflow Definitions Loader to load workflow process definitions from flat files to the database without using Oracle Workflow Builder. See: Using the Workflow Definitions Loader, page 8-8.

Additioinally, you can use the Workflow XML Loader to load XML definitions for Business Event System objects between a database and a flat file. See: Using the Workflow XML Loader, page 8-11.

For details about configuring Oracle Workflow security, see: Oracle Workflow Security, page 3-1.

Identifying the Version of Your Oracle Workflow Server

If you ever need to determine the version of the Oracle Workflow server you are running, you can connect to your Oracle Workflow database account using SQL*Plus and run a script called `wfver.sql`. See: wfver.sql, page 9-14.

In addition, all Oracle Workflow modules, such as the Workflow Definitions Loader, Oracle Workflow Builder, notification mailers, and the Workflow Monitor, automatically verify that the module is compatible with the version of the Oracle Workflow server that it is operating against. This version compatibility check helps to prevent problems such as running Oracle Workflow Builder 2.6.3 against an Oracle Workflow 2.0.3 database.

Oracle Workflow Setup Checklist

The following table lists Oracle Workflow setup steps and shows whether each step is required or optional. You need to perform optional steps only if you plan to use the related feature or complete certain business functions.

Setup Checklist

Step Number	Requirement	Step
Step 1	Optional	Partitioning Workflow Tables, page 2-8
Step 2	Required	Setting Global User Preferences, page 2-9
Step 3	Required	Setting Up an Oracle Workflow Directory Service, page 2-14
Step 4	Optional	Setting Up Additional Languages, page 2-36
Step 5	Required	Setting Up Background Workflow Engines, page 2-39
Step 6	Optional	Implementing Notification Mailers, page 2-43
Step 7	Optional	Modifying Your Message Templates, page 2-84
Step 8	Optional	Adding Worklist Functions to User Responsibilities, page 2-136
Step 9	Optional	Setting the WF: Notification Reassign Mode Profile Option, page 2-137
Step 10	Optional	Setting Up Notification Handling Options, page 2-138
Step 11	Optional	Setting Up for Electronic Signatures, page 2-140
Step 12	Optional	Customizing the Logo on Oracle Workflow's Web Pages, page 2-145

Step Number	Requirement	Step
Step 13	Optional	Adding Custom Icons to Oracle Workflow, page 2-145
Step 14	Required	Setting Up the Business Event System, page 2-145

The following flowchart shows the Oracle Workflow setup steps in a graphical format, indicating which steps are required and which are optional.

Setup Flowchart
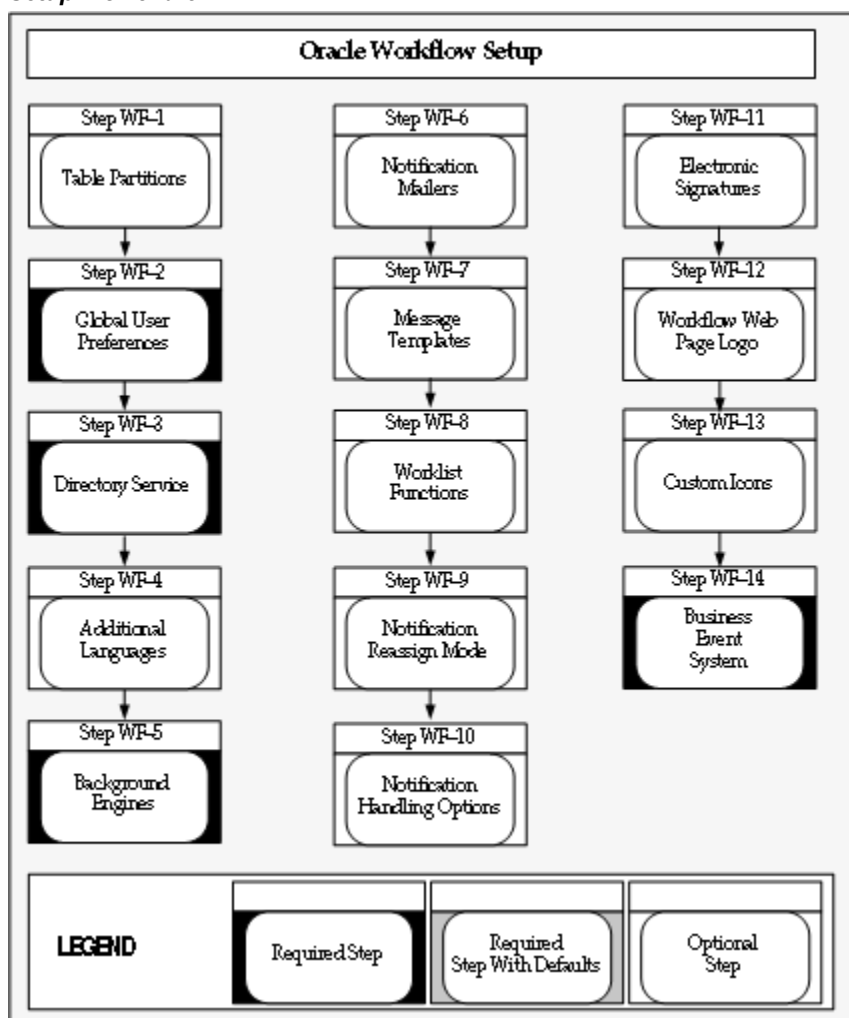
Step 1: Partitioning Workflow Tables

Partitioning addresses key issues in supporting very large tables and indexes by letting you decompose them into smaller and more manageable pieces called partitions. SQL queries and DML statements do not need to be modified in order to access partitioned tables. However, once partitions are defined, DDL statements can access and manipulate individual partitions rather than entire tables or indexes. In this way, partitioning can simplify the manageability of large database objects. Also, partitioning is entirely transparent to applications.

You can optionally run a script called `wfupartb.sql` to partition certain Workflow tables that store runtime status data. This step is highly recommended for performance gain.

The script partitions four Workflow tables and recreates the associated indexes. The following table shows the Workflow tables and indexes on which the script runs.

Partitioned Tables and Associated Indexes

Table	Indexes
WF_ITEM_ACTIVITY_STATUSES	WF_ITEM_ACTIVITY_STATUSES_PK, WF_ITEM_ACTIVITY_STATUSES_N1, and WF_ITEM_ACTIVITY_STATUSES_N2
WF_ITEM_ACTIVITY_STATUSES_H	WF_ITEM_ACTIVITY_STATUSES_H_N1 and WF_ITEM_ACTIVITY_STATUSES_H_N2
WF_ITEM_ATTRIBUTE_VALUES	WF_ITEM_ATTRIBUTE_VALUES_PK
WF_ITEMS	WF_ITEMS_PK, WF_ITEMS_N1, WF_ITEMS_N2, and WF_ITEMS_N3

Before running the partitioning script, you should back up these four tables so that you can restore them in case the script fails.

To run the script, you must have sufficient free space on the table and index tablespaces. During the creation of the partitioned tables, the script requires slightly more diskspace than the underlying tables, in the same tablespace where the underlying tables are located. Similarly, sufficient free space is required for the index tablespace.

Additionally, you should allow sufficient time for the script to run. The amount of time needed depends on the amount of data in the tables. When the tables already contain existing data, such as after an upgrade from a previous release, the script requires more time than it does when the tables are empty, such as after a fresh installation of Oracle

Workflow. To minimize the time required, run the script as early as possible in your setup process.

> **Important:** If you are running the partitioning script through Oracle Net Services, then you must set the TWO_TASK variable before you begin.

The `wfupartb.sql` script is located in the `admin/sql` subdirectory under `$FND_TOP`. Use the script as follows:

```
sqlplus <apps_user>/<apps_passwd> @wfupartb <fnd_user>
<fnd_passwd> <apps_user> <apps_passwd>
```

For example:

```
sqlplus apps/apps @wfupartb applsys apps apps apps
```

If the partitioning script fails, you must perform any necessary cleanup manually. Since the script's operations are DDL operations running in `nologging` mode, rollback is not possible.

Related Topics

Partitioning for Performance, page C-7

Step 2: Setting Global User Preferences

Users can control how they interact with Oracle Workflow by specifying user preferences. As a workflow administrator, you also have access to set default user preference values globally for the entire enterprise, using the Workflow Configuration page. An individual user can override a default user preference at any time by modifying his or her preference setting, using the Preferences page in Oracle Applications.

See: Set Preferences, *Oracle Applications User's Guide*.

To Set Global Preferences:

1. Use a Web browser to navigate to the Workflow Configuration page, using a responsibility and navigation path specified by your system administrator. See: Oracle Workflow Administrator Navigation Paths, page A-1.

   > **Note:** You must have workflow administrator privileges to set global workflow preferences in the Workflow Configuration page. If you do not have administrator privileges, you can view global workflow preferences, but you cannot modify them. After installation of Oracle Applications, workflow administrator privileges are initially assigned to the SYSADMIN user by default.

You can change that assignment in this page.

2. In the Workflow System Administrator field, select the role to which you want to assign workflow administrator privileges. Any user associated with this role can set global workflow preferences in this page, view and respond to any user's notifications, define rules to handle notifications automatically in a user's absence, view workflows owned by any user and perform administrative operations in the Status Monitor, run test workflows in the Developer Studio, and maintain Business Event System objects and raise test events in the Event Manager. See: Setting Up an Oracle Workflow Directory Service, page 2-14.

If you want all users and roles to have workflow administrator privileges, such as in a development environment, enter an asterisk (*) in the Workflow System Administrator field.

> **Note:** To find out which role currently has workflow administrator privileges, without accessing the Workflow Configuration page, you can use the following command:

```
select text
from wf_resources
where name = 'WF_ADMIN_ROLE';
```

After installing Oracle Workflow, you should change the Workflow System Administrator preference from the default setting to the role that you want to have administrator privileges. The default setting after installation is SYSADMIN. You must log in as the SYSADMIN user to access the Workflow Configuration page and specify the preferences you want.

> **Note:** The SYSADMIN role is different than the role associated with the System Administrator responsibility in Oracle Applications. If you want to assign workflow administrator privileges to this or any other Oracle Applications responsibility, you must set the Workflow System Administrator preference to the internal name of the workflow role associated with that responsibility.
>
> You can query the WF_ROLES view to find the role name for a responsibility. For example, to find the role names for various administrator responsibilities in Oracle Applications, use the following command:
>
> ```
> select name, display_name
> from wf_roles
> where display_name like '%Admin%';
> ```
>
> If you set the Workflow System Administrator preference to the role name of a responsibility, then any Oracle Applications user

with that responsibility will have workflow administrator privileges.

> **Note:** The user through which a notification mailer accesses Oracle Application Framework content must have workflow administrator privileges in order to access the content for every user's notifications. If you change the Workflow System Administrator preference, ensure that the Framework User mailer parameter is set to a user that is a member of the Workflow System Administrator role, or, if you set the Workflow System Administrator preference to a responsibility, ensure that the Framework User mailer parameter is set to a user that has that responsibility. See: Notification Mailer Configuration Wizard, page 7-30.

3. If you are integrating with Oracle Internet Directory, specify the Lightweight Directory Access Protocol (LDAP) server information for the LDAP directory to which you will connect. If you already configured these parameters while installing Oracle Application Server with Oracle E-Business Suite, Oracle Workflow displays those values here. For more information, see: Installing Oracle Application Server 10*g* with Oracle E-Business Suite Release 12 (Oracle*MetaLink* note 376811.1) and Oracle Single Sign-On Integration, *Oracle Applications System Administrator's Guide - Security*.

   - Host - The host on which the LDAP directory resides.

   - Port - The port on the host.

   - Username - The LDAP user account used to connect to the LDAP server. This user name must have write privileges and is required to bind to the LDAP directory. For example:

     ```
     cn=orcladmin
     ```

   - Old Password - Enter your current LDAP password. Oracle Workflow validates this password before letting you change it.

   - New Password - Enter the new LDAP password you want to use. The password must be at least five characters long.

   - Repeat Password - Enter your new LDAP password again in this field to confirm it. You must enter exactly the same value that you entered in the New LDAP Password field.

     > **Note:** LDAP password values are masked as asterisks in the

display and are stored in encrypted form.

- Change Log Base Directory - The LDAP node under which change logs are located. For example:

  ```
  cn=changelog
  ```

- User Base Directory - The LDAP node under which user records can be found. For example:

  ```
  cn=Users,dc=oracle,dc=com
  ```

4. Specify details about the local system that identifies this installation of Oracle Workflow in the Business Event System. See: Systems, *Oracle Workflow Developer's Guide*.

   - System Name - The system name for the database where this installation of Oracle Workflow is located. Oracle Workflow automatically creates the system definition for this database in the Event Manager during installation.

   - Status - Select the execution status for the local system.

     - Enabled - Subscriptions are executed on all events. Oracle Workflow sets the system status to Enabled by default.

     - Local Only - Subscriptions are executed only on events raised on the local system.

     - External Only - Subscriptions are executed only on events received by inbound agents on the local system.

     - Disabled - No subscriptions are executed on any events.

       Note: The local system settings are specific to this installation of Oracle Workflow and are not included when Business Event System data is replicated to other systems.

5. Specify default workflow preferences for your users.

   - Notification Style - Specify whether Oracle Workflow should send e-mail notifications to users, and if so, in what format. A user can override this default setting by specifying a different notification style in his or her individual Oracle E-Business Suite preferences.

     - HTML mail with attachments - Send notifications as HTML-formatted e-mail with attached links to the Notification Details page. Users must read

their e-mail using an HTML e-mail client.

- Plain text mail with HTML attachments - Send notifications as plain text e-mail but include the HTML-formatted version of the notifications and links to the Notification Details page as attachments.

- Plain text mail - Send notifications as plain text e-mail.

- Plain text summary mail - Send a summary of all notifications as plain text e-mail. Users must use the Worklist Web pages to view and take action on individual notifications.

- Do not send me mail - Do not send the notifications as e-mail. Users must use the Worklist Web pages to view and take action on their notifications.

- HTML mail - Send notifications as HTML-formatted e-mail only, without any standard attachments. If custom attachments have been defined for individual notifications in a process, however, those attachments will still be included. Users must read their e-mail using an HTML e-mail client.

- HTML summary mail - Send a summary of all notifications as HTML-formatted e-mail, with a link to the Worklist page as well as links to each notification in the Notification Details page. Users must use the Worklist Web pages to view and take action on individual notifications.

    Note: To send e-mail notifications, you must configure and run a notification mailer. Additionally, users who are to receive e-mail notifications must have an e-mail address defined. You can run a diagnostic test through Oracle Diagnostics to check that all users with a notification preference to receive e-mail have an e-mail address defined. See: Implementing Notification Mailers, page 2-43, Setting Up an Oracle Workflow Directory Service, page 2-14, and Oracle Workflow Diagnostic Tests, page E-1.

    Users can always access their notifications through the Worklist Web pages, even if their notification preference also includes e-mail notifications.

    Note: Oracle Workflow also uses a notification preference called Disabled, which is set automatically at the user preference level for a user with an invalid e-mail address. Do not set the global notification preference to this value.

- Browser Signing DLL Location - The location of the `Capicom.dll` file that is used for Web page operations with encryption in the Microsoft Internet Explorer browser. This preference is required only if you plan to use certificate-based digital signatures to confirm notification responses, and your users access Oracle Applications with Microsoft Internet Explorer.

  By default, this preference is set to a URL at which the `Capicom.dll` file can be downloaded from Microsoft's Web site. In most cases, you do not need to change this setting. However, you can update this preference if the location of the `Capicom.dll` file changes, or if you choose to store a copy of the file on your local network and point to that location instead.

  For more information about setting up for certificate-based signatures, see: Loading Certificates for Digital Signatures, page 2-140.

6. Review details about the JInitiator plugin in your Oracle Applications installation. Oracle Workflow uses JInitiator to launch Oracle Applications forms linked to notifications.

   - Class ID - The class identifier for this version of JInitiator.

   - Download Location - The location where the JInitiator executable is staged for download to users' client machines.

   - Version - The JInitiator version number.

     Note: The global notification and DLL location preferences are saved to the Oracle Workflow preferences table for a special user name called ‑WF_DEFAULT‑. The workflow administrator role, LDAP, local system, and JInitiator information is saved to the Oracle Workflow resources table.

Step 3: Setting Up an Oracle Workflow Directory Service

Oracle Workflow requires a directory service to provide information about the individuals and roles in your organization who may utilize Oracle Workflow functionality and receive workflow notifications. Oracle Workflow references this user and role information through the following views.

- WF_USERS - Individual users.

- WF_ROLES - Roles, which can have one or more users as members.

- WF_USER_ROLES - Associations of users with the roles of which they are members.

> **Note:** A role can contain only individual users as its members. It cannot contain another role. However, roles can be related to each other in a hierarchy so that users assigned to one role automatically inherit membership in its superior roles as well.

- WF_USER_ROLE_ASSIGNMENTS_V - Assignments of users to roles, both direct and inherited through role hierarchy relationships.

See: Workflow Directory Service Views, page 2-24.

Oracle Workflow provides a predefined directory service for you that is implemented by default during installation, with directory service views for users and roles from the unified Oracle Applications environment. See: Setting Up a Directory Service for Oracle Workflow, page 2-16.

You can also create your own directory service by defining custom views with the required columns. However, note that only the predefined directory service provided by Oracle Workflow is supported by Oracle. See: Oracle Workflow Support Policy, *Oracle Workflow Developer's Guide*.

Oracle Workflow provides local directory repository tables called WF_LOCAL_ROLES and WF_LOCAL_USER_ROLES. These tables should always be included in any implementation of the WF_USERS, WF_ROLES, and WF_USER_ROLES views.

- WF_LOCAL_ROLES stores role information, including a user flag to mark those roles that also represent individual users. This table contains columns similar to those required in the WF_USERS and WF_ROLES views.

- WF_LOCAL_USER_ROLES stores information about the associations of users with roles. This table contains columns similar to those required in the WF_USER_ROLES view.

Oracle Workflow also provides tables to support extended directory service features.

- WF_LOCAL_ROLES_TL stores translated display name and description values for multiple language support (MLS) in the WF_USERS and WF_ROLES views.

- WF_ROLE_HIERARCHIES stores information about the hierarchical relationships between roles.

- WF_USER_ROLE_ASSIGNMENTS stores information about the direct and inherited assignments of users to roles.

The Workflow local tables store denormalized user and role information originating from various other Oracle Applications modules, so that the directory service views can access this information with good performance. You can also use these tables to store ad hoc users and roles by calling the appropriate Workflow directory service PL/SQL APIs.

You should periodically purge ad hoc users and roles from the Workflow local tables

after they have expired in order to improve performance. See: Directory, *Oracle Workflow API Reference*.

For more information, see: Workflow Directory Service APIs, *Oracle Workflow API Reference*, Ad Hoc Users and Roles, *Oracle Workflow Developer's Guide*, and Oracle Workflow Security, page 3-1.

Setting Up a Directory Service for Oracle Workflow within Oracle Applications:

Oracle Workflow uses a directory service model in which denormalized information is maintained in the Workflow local tables for performance gain. The Workflow local tables store user and role information originating from various other Oracle Applications modules, as well as ad hoc users and roles, so that the WF_USERS, WF_ROLES, WF_USER_ROLES, and WF_USER_ROLE_ASSIGNMENTS_V views can access this information with good performance. You should maintain synchronization between the user and role information stored in application tables by the source modules and the information stored in the Workflow local tables.

Directory Service Views

The predefined WF_USERS, WF_ROLES, WF_USER_ROLES, and WF_USER_ROLE_ASSIGNMENTS_V directory service views for Oracle Workflow are based solely on the Workflow local tables where the denormalized information is stored. These view definitions are automatically created for you during installation. See: Workflow Directory Service Views, page 2-24.

- WF_USERS is based on WF_LOCAL_ROLES where the user flag is set to Y and on WF_LOCAL_ROLES_TL.

- WF_ROLES is based on WF_LOCAL_ROLES and on WF_LOCAL_ROLES_TL.

- WF_USER_ROLES is based on WF_LOCAL_USER_ROLES.

- WF_USER_ROLE_ASSIGNMENTS_V is based on WF_USER_ROLE_ASSIGNMENTS.

> **Note:** You can customize your directory service by creating your own custom view definitions, provided that you define the required columns and map to the Workflow local tables. However, note that only the predefined directory service views provided by Oracle Workflow are supported by Oracle. See: Oracle Workflow Support Policy, *Oracle Workflow Developer's Guide*.

The only roles in WF_LOCAL_ROLES that are marked as individual users with the user flag set to Y are roles that represent Oracle Applications users, originating from the FND_USER table, roles that represent Oracle Trading Community Architecture (TCA) person parties, roles that represent TCA contacts (relationship parties), or roles that

represent ad hoc users. Records originating from other application tables are treated solely as roles, with the user flag set to `N`. The WF_LOCAL_USER_ROLES table is used to associate Oracle Applications users, TCA person parties, and TCA contacts with roles defined by other applications.

> **Note:** An Oracle Applications user may be associated with an Oracle Human Resources person. In this case, some person information is combined into the user's record in WF_LOCAL_ROLES. In such a combined record, the originating system is changed from `FND_USR` to `PER`, and the display name is taken from Oracle Human Resources, while the internal name is the Oracle Applications user name from FND_USER, and the user flag is still set to `Y`.

> Each Oracle Human Resources person is also represented in WF_LOCAL_ROLES as a role with the originating system PER_ROLE and the user flag set to `N`. This record remains unaffected whether the person is linked to an Oracle Applications user or not.

> The following table summarizes the different ways in which Oracle Applications users and Oracle Human Resources people are stored in WF_LOCAL_ROLES.

> *Oracle Applications Users and Oracle Human Resources People in WF_LOCAL_ROLES*

> | Type of Role | Orig_System | User_Flag |
> |---|---|---|
> | Oracle Applications user, not linked to an Oracle Human Resources person | FND_USR | Y |
> | Oracle Applications user linked to an Oracle Human Resources person | PER | Y |
> | Oracle Human Resources person | PER_ROLE | N |

> To link an Oracle Applications user to an Oracle Human Resources person, navigate to the Users window in Oracle Applications and select the appropriate person name in the Person field for that user. See: Users Window, *Oracle Applications System Administrator's Guide*.

> You should only link an Oracle Human Resources person to one Oracle

Applications user. If a person is linked to more than one user, notifications for that person may become inaccessible, and workflow processes may be halted while waiting for those notifications to be completed. Additionally, assigning a person to multiple users may cause errors in other Oracle Applications modules as well. For this reason, you must not link an Oracle Human Resources person to more than one Oracle Applications user.

The WF_LOCAL_ROLES and WF_LOCAL_USER_ROLES tables are partitioned by the originating system within Oracle Applications that was the source of the denormalized information. This partitioning provides faster data access and also allows each originating system to be synchronized with the Workflow local tables individually. Each table also includes a separate partition that contains ad hoc users and roles as well as data from any system that does not have its own partition.

The partition information for each originating system is stored in the WF_DIRECTORY_PARTITIONS table. There are partitions for the following systems:

- `WF_LOCAL_ROLES` - Ad hoc users and roles, as well as data from any originating system that does not have its own partition

- `FND_USR` - FND users, which may or may not be linked to Oracle Human Resources people

- `FND_RESP` - FND responsibilities

- `PER_ROLE` - HR people

- `POS` - HR positions

- `AMV_APPR` - MarketView approvals

- `AMV_CHN` - MarketView channels

- `ENG_LIST` - Engineering approval list

- `HZ_GROUP` - TCA groups

- `HZ_PARTY` - TCA person parties and contacts

- `GBX` - Federal HR group boxes

- `HTB_SEC` - This partition does not participate in bulk synchronization.

- `PQH_ROLES` - Position Control roles

- `UMX` - User Management roles. This partition does not participate in bulk

synchronization.

> **Note:** Normally each partition contains only records that originate from the corresponding system. However, the FND_USR partition can contain both roles with an originating system value of FND_USR, which are unlinked Oracle Applications users, and roles with an originating system value of PER, which are Oracle Applications users that are linked to Oracle Human Resources people.

See: Ad Hoc Users and Roles, *Oracle Workflow Developer's Guide*.

Synchronizing Workflow User and Role Information

For each Oracle Applications module that is a source of Oracle Workflow user and role information, the information stored in the source application tables must be synchronized with the denormalized information in the Workflow local tables. The Workflow local synchronization APIs are used to perform this synchronization.

Incremental Synchronization

Oracle Workflow automatically performs an initial synchronization of the user and role information in all the related originating systems during installation. Subsequently, you must continue synchronizing the user and role information from the source modules with the Workflow local tables. Each Oracle Applications module that stores user and role information in its application tables automatically synchronizes that information with the information in the Workflow local tables on an incremental basis, using the Workflow local synchronization APIs.

Bulk Synchronization

If necessary, you can run a concurrent program named Synchronize WF LOCAL Tables to perform synchronization in bulk, periodically refreshing the information in the Workflow local tables for the affected modules. This concurrent program is provided for troubleshooting and diagnostic purposes or to synchronize the Workflow local tables with the user and role information stored in the product application tables in case incremental synchronization is not being performed automatically for any reason.

Oracle Workflow provides a request set named Synchronize Workflow LOCAL Tables that contains ten instances of the Synchronize WF Local Tables program, one for each originating system. You can use this request set to submit requests for all the originating systems at once. Note that because this program is incompatible with itself, each request is defined as a separate stage and the stages will run sequentially. By default, this request set is scheduled to run once a day to provide a minimal level of synchronization. You can modify the schedule for the request set to perform synchronization more frequently.

You only need to run the bulk synchronization program for products that are not successfully performing incremental synchronization. If you want to run the Synchronize Workflow LOCAL Tables request set for a subset of products, you can

remove the program instances for other products' originating systems from the Synchronize Workflow LOCAL Tables request set. See: Defining Request Sets, *Oracle Applications System Administrator's Guide*.

> **Note:** Products that use role hierarchies do not participate in bulk synchronization. These products must perform incremental synchronization.

To submit the Synchronize Workflow LOCAL Tables request set

1. Navigate to the Submit Requests form in Oracle Applications (System Administrator: Requests > Run). See: Running Reports and Programs, *Oracle Applications User's Guide*.

2. Choose to run a request set and select Synchronize Workflow LOCAL Tables as the request set to run.

3. Enter the values you want for the program parameters. See: To submit a single request for the Synchronize WF LOCAL Tables concurrent program, page 2-20.

4. Select the print and run options you want to define the schedule for this request set, and choose Submit to submit the requests.

To submit a single request for the Synchronize WF LOCAL Tables concurrent program

1. Navigate to the Submit Requests form in Oracle Applications (System Administrator: Requests > Run). See: Running Reports and Programs, *Oracle Applications User's Guide*.

2. Choose to run a single request and select the Synchronize WF LOCAL Tables concurrent program as the request to run.

3. In the Parameters window, enter the values you want for the following parameters:

    • Orig System - Select the name of the originating system whose user and role information you want to synchronize with the WF_LOCAL tables. These system names are stored in the `FND_WF_ORIG_SYSTEMS` lookup type. You can select the following systems:

        • `ALL` - All originating systems

        • `AMV_APPR` - MarketView Approvals

        • `AMV_CHN` - MarketView Channels

        • `ECX` - XML Gateway Ad Hoc Users

- ENG_LIST - Engineering Approval List

- FND_RESP - FND Responsibilities

- FND_USR - FND Users

- GBX - Federal HR Group Boxes

- HZ_GROUP - TCA Groups

- HZ_PARTY - TCA Parties

- PER_ROLE - HR People

- POS - HR Positions

- PQH_ROLES - Position Control Roles

- Parallel Processes - Enter the number of parallel processes to run. Usually this parameter is set to 1. However, if your hardware resources allow, you can optionally set this parameter to a higher value in order to parallelize the queries during execution of the program.

- Logging Mode - Select the logging mode you want. This mode determines whether redo log data is generated for database operations performed by the bulk synchronization process. The default value for this parameter is LOGGING, which generates redo log data normally. You can optionally set the logging mode to NOLOGGING to suppress redo log data, obtaining a performance gain. Without this redo log data, no media recovery is possible for the Workflow directory tables and indexes, requiring that you re-run the bulk synchronization process in the event of a media recovery scenario. One additional scenario that would require the process to be re-run is when you open a standby database, whose maintenance also depends on redo log data.

- Temporary Tablespace - Select the temporary tablespace the program should use.

- Raise Errors - Select Yes or No to indicate whether the program should raise an exception if it encounters an error. Usually you can leave this parameter set to the default value, which is Yes.

4. Select the print and run options you want to define the schedule for this request, and choose Submit to submit the request. You can submit multiple requests for this program to perform synchronization for different originating systems at different frequencies. However, note that because this program is incompatible with itself, only one request for the program can run at a time.

> **Note:** Additionally, you must not run bulk synchronization using APIs or scripts from SQL*Plus while you are running the Synchronize WF LOCAL Tables concurrent program or the Synchronize Workflow LOCAL Tables request set, as the two processes will interfere with each other.

How Bulk Synchronization Is Performed

The bulk synchronization program retrieves user and role information from an originating system through views that present the information that was formerly included in the previous implementation of the Workflow directory service views. Each originating system provides two new views, one that contains the same columns as WF_ROLES and one that contains the same columns as WF_USER_ROLES.

> **Note:** Originating systems that use role hierarchies do not participate in bulk synchronization. These originating systems must perform incremental synchronization.

For backward compatibility, the originating systems' synchronization views must present exactly the same user and role information that was included for that system in the previous implementation of the Workflow directory service views. The information must be presented in the format required by Oracle Workflow, with no duplicates. For example, the internal name for a user or role must be sourced from a column that is no longer than 320 characters. It is recommended that internal names be all uppercase. If the source table in the originating system does not have a column that meets these criteria, the internal name should be defined to be *<orig_system>*: *<orig_system_id>* instead, so that Oracle Workflow can reference the original base table where users or roles are stored and a unique user or role in that table.

> **Note:** If internal names in all uppercase are used, the names should be initially entered in the database in all uppercase. Forcing the names to uppercase in the view definition results in poor performance when accessing these views.

> **Note:** You can customize these originating system synchronization view definitions to specify the data you want to include in bulk synchronization, provided that your customized views meet the requirements listed above. However, note that the originating systems that have implemented incremental synchronization will also be propagating user and role information to the Workflow local tables automatically, so the synchronization views used for bulk synchronization are not the only source of data for Oracle Workflow. Also note that only the predefined synchronization views provided by

Oracle Applications are supported by Oracle. See: Oracle Workflow Support Policy, *Oracle Workflow Developer's Guide*.

When you run the bulk synchronization program for a particular originating system, the program extracts the role and user/role association information from that system's synchronization views and loads the information into staging tables. The program then performs a partition exchange between the staging tables and the WF_LOCAL_ROLES, WF_LOCAL_ROLES_TL, WF_LOCAL_USER_ROLES, and WF_USER_ROLE_ASSIGNMENTS tables to update the partitions for that system in the Workflow local tables. Finally, the staging tables are truncated.

> **Note:** The bulk synchronization program does not store or modify any information in the `WF_LOCAL_ROLES` partitions within the WF_LOCAL_ROLES and WF_LOCAL_USER_ROLES tables that contain ad hoc users and roles.

Role Hierarchies

Roles can be related to each other in a hierarchy so that users assigned to one role automatically inherit membership in its superior roles as well. Role hierarchies enable role-based access control in Oracle Applications.

For example, a company could define a role hierarchy with three roles: sales manager, sales representative, and employee. A user with the sales manager role automatically inherits the sales representative role, and a user with the sales representative role automatically inherits the employee role. If user A is assigned directly to the sales representative role, then user A will also have an inherited assignment to the employee role. If user B is assigned directly to the sales manager role, user B will also have inherited assignments to both the sales representative role and the employee role.

Oracle Workflow stores hierarchical relationships between roles in the WF_ROLE_HIERARCHIES table. Oracle Workflow also stores denormalized information about direct and inherited assignments of users to roles in the WF_USER_ROLE_ASSIGNMENTS table for performance gain. If a user is associated with a certain role through more than one direct or inherited assignment, the WF_USER_ROLE_ASSIGNMENTS table tracks which assignments are currently valid and expires the user/role association only when all assignments have ended.

> **Note:** Originating systems that use role hierarchies do not participate in bulk synchronization. These originating systems must perform incremental synchronization.

Validating Directory Service Information

You can run a diagnostic test through Oracle Diagnostics to check that there are no duplicate roles in the WF_LOCAL_ROLES table. See: Oracle Workflow Diagnostic

Tests, page E-1.

If you encounter inconsistencies in your directory service information, Oracle Support may direct you to run the Workflow Directory Services User/Role Validation concurrent program (FNDWFDSURV) to validate and correct the information about user/role associations. You should also investigate the cause of any inconsistencies. Use Standard Request Submission to run this program, specifying the following parameters.

- p_BatchSize - The number of records to process in one transaction before the program commits data. If your resources allow, you can increase this value to enhance performance. The default value is 10,000 records.

- p_Check_Dangling - Select Yes to check that all users and roles referenced by user/role associations exist in the WF_LOCAL_ROLES table. If the user or role or both are missing for any user/role association, the program removes that user/role association from the WF_LOCAL_USER_ROLES table. Select No to skip this check. The default value is No.

- Add missing user/role assignments - Select Yes to check that all user/role associations in the WF_LOCAL_USER_ROLES table have corresponding user/role assignments in the WF_USER_ROLE_ASSIGNMENTS table and add any missing direct assignments. Select No to skip this check. The default value is No.

- Update Who columns in WF tables - Select No to preserve the existing values in the LAST_UPDATED_BY and LAST_UPDATE_DATE standard Who columns when the program updates corrupt records in the Workflow directory service tables. Select Yes if you want the program to update the standard Who columns. The recommended value, which is also the default value, is No.

See: Running Reports and Programs, *Oracle Applications User's Guide*.

Workflow Directory Service Views:

Oracle Workflow relies on views named WF_USERS, WF_ROLES, WF_USER_ROLES, and WF_USER_ROLE_ASSIGNMENTS_V to reference user and role information. Other views provide further access to Workflow directory service data, including WF_ALL_ROLES_VL, WF_ALL_USER_ROLES, and WF_ALL_USER_ROLE_ASSIGNMENTS. These directory service views for the unified Oracle Applications environment are automatically defined for you during installation.

> **Note:** An expiration date can be assigned to each role in WF_LOCAL_ROLES, each user/role association in WF_LOCAL_USER_ROLES, and each user/role assignment in WF_ROLE_HIERARCHIES. After that date, an expired role is no longer included in the predefined WF_USERS and WF_ROLES view, an expired user/role association is no longer included in the predefined WF_USER_ROLES view, and an expired user/role assignment is no

longer included in the WF_USER_ROLE_ASSIGNMENTS_V view.

However, note that although the expired rows no longer appear in these views, they still exist in the Workflow local tables. You should periodically purge expired ad hoc users and roles using the *WF_PURGE.Directory()* API in order to improve performance. See: Directory, *Oracle Workflow API Reference*.

You can also create your own directory service by defining custom views with the required columns. However, note that only the predefined directory services provided by Oracle Workflow are supported by Oracle. See: Oracle Workflow Support Policy, *Oracle Workflow Developer's Guide*.

If you create your own custom view definitions:

*   Each individual user identified by WF_USERS must also appear in the WF_ROLES view as a role.

*   You should set the user flag in the underlying tables to `Y` for all the roles that also represent individual users, and set the user flag to `N` for all other roles.

*   You should include the WF_LOCAL_ROLES and WF_LOCAL_USER_ROLES tables in the view definitions. You can optionally also include the WF_LOCAL_ROLES_TL table for multiple language support.

*   You should avoid selecting from `DUAL` to incorporate additional users and roles into the directory service views as this allows you to violate the unique constraint on certain columns of the views and reduces performance with unnecessary joins between the `'select from DUAL'` statements.

*   To take advantage of unique indexes when querying users, make sure you initially enter the usernames in your database in uppercase only. Forcing the usernames to uppercase in your view definition results in poor performance when accessing these views.

*   You should run the script `wfdirchk.sql` to validate your directory service data model. This script is located in the `$FND_TOP/sql` directory. See: Wfdirchk.sql, page 9-8.

    Note: Avoid making a join to a view that contains a union, as this results in poor database performance. The Oracle Database is unable to preserve the indexes in that view when you make such a join. The workflow directory service views you create will most likely contain unions; therefore you should not join to them directly. If you need to retrieve data from any of the three directory services views, use the appropriate directory services API. See: Workflow Directory Service

APIs, *Oracle Workflow API Reference*.

WF_USERS

The WF_USERS view references information about the individuals in your organization who may utilize Oracle Workflow functionality or receive workflow notifications.

> **Note:** In WF_LOCAL_ROLES, a role that is an individual user has the user flag set to `Y`.

> **Note:** This view includes only Oracle Applications users originating from the FND_USER table, TCA person parties, TCA contacts, and ad hoc users, although an Oracle Applications user record may also include information from Oracle Human Resources if the user is linked to an Oracle Human Resources person.

The WF_USERS view must contain the following required columns:

- Name - The internal name of the user as referenced by the Workflow Engine and Notification System. For example, an internal name for a user can be `MBEECH` or `009`, where `009` represents the user's employee ID.

  > **Important:** If you define custom views, the Name column must be sourced from a column that is no longer than 320 characters, and it is recommended that the internal name be all uppercase. If your source table does not have a column that meets these criteria, DO NOT use string functions to force these restrictions. Instead, define the Name column to be `<orig_system>:<orig_system_id>` so that Oracle Workflow can reference the original base table where users are stored and a unique user in that table. For example, `PER_PEOPLE:009` could represent a user whose employee ID is `009` and whose record is stored in a personnel table called `PER_PEOPLE`.

- Display_Name - The display name of the user. An example of a display name can be `'Beech, Matthew'`.

- Description - An optional description of the user.

- Notification_Preference - The method by which this user prefers to receive notifications. A value of `MAILTEXT`, `MAILHTML`, `MAILHTM2`, or `MAILATTH` allows users to receive and respond to notifications by plain text e-mail, HTML-formatted e-mail with attachments, HTML-formatted e-mail without attachments, or by plain text e-mail with HTML attachments, respectively. A value of `QUERY` allows users to

query notifications from the Worklist Web page. A value of DISABLED indicates that the e-mail address on record for the user is invalid, so no e-mail notifications will be sent; meanwhile the user can only query notifications from the Worklist Web page. Finally, a value of SUMMARY or SUMHTML allows users to receive periodic e-mail summaries of their open notifications. However, to respond to the individual notifications, they must view the notification in the Worklist Web pages. See: Overview of Notification Handling, *Oracle Workflow User's Guide* and Notification Preferences, page 2-58.

> **Note:** A notification preference of MAILTEXT, MAILHTML, MAILHTM2, or MAILATTH also allows users to view their notifications in the Worklist Web pages.

> **Note:** If no notification preference is provided by the originating system, the notification preference is set to MAILHTML by default.

> **Note:** If you define custom views, you can map the Notification_Preference column over the Oracle Workflow preferences table using the statement below. The benefit of doing so is that you can then globally set the default notification preference for all users in your enterprise using the Workflow Configuration page, and let individual users override that default value by changing their notification preference in the Preferences page in Oracle Applications. See: Setting Global User Preferences, page 2-9 and get_pref, *Oracle Workflow Developer's Guide*.
>
> ```
> NVL(wf_pref.get_pref(USR.USER_NAME,'MAILTYPE'),
> 'MAILHTML')
> ```

- Language - The value of the database NLS_LANGUAGE initialization parameter that specifies the default language-dependent behavior of the user's notification session. The Language column is required and cannot be left null. Refer to your Oracle Database user's guide or installation manual for the list of supported language conventions.

> **Note:** For roles that are Oracle Applications users marked with an originating system of FND_USR or PER, Oracle Workflow uses the *GetRoleInfo()* procedure to find the language for a user, rather than querying this column in the directory service views. *GetRoleInfo()* by default retrieves the language value from the ICX: Language profile option for that Oracle Applications user.

> However, if the WF_PREFERENCE resource token is defined and set to FND, then the *GetRoleInfo()* procedure obtains the language value

from the Oracle Workflow preferences table instead.

> **Note:** If you define custom views, and the WF_PREFERENCE
> resource token is set to FND, then you can map the Language
> column over the Oracle Workflow preferences table using the
> statement below. The benefit of doing so is that you can then
> globally set the default language for all users in your enterprise and
> let individual users override that default value by changing their
> language preference. See: Setting Global User Preferences, page 2-9
> and get_pref, *Oracle Workflow Developer's Guide*.
>
> ```
> NVL(wf_pref.get_pref(USR.USER_NAME,'LANGUAGE'),
> <default_language>)
> ```

> **Important:** Ensure that the e-mail templates used by notification
> mailers to send notifications have been translated by Oracle to the
> language you wish to set. The standard e-mail templates are
> delivered in a file called wfmail.wft under the subdirectory
> $FND_TOP/import/<lang>. You can check the appropriate
> language subdirectory to verify if the templates have been
> translated to the language you wish to set. See: Modifying Your
> Message Templates, page 2-84.

- Territory - The value of the database NLS_TERRITORY initialization parameter that
  specifies the default territory-dependent formatting used in the user's notification
  session. The Territory column is required and cannot be left null. Refer to your
  Oracle Database user's guide or installation manual for the list of supported
  territory conventions.

> **Note:** For roles that are Oracle Applications users marked with an
> originating system of FND_USR or PER, Oracle Workflow uses the
> *GetRoleInfo()* procedure to find the territory for a user, rather than
> querying this column in the directory service views. *GetRoleInfo()*
> by default retrieves the territory value from the ICX: Territory
> profile option for that Oracle Applications user.
>
> However, if the WF_PREFERENCE resource token is defined and set
> to FND, then the *GetRoleInfo()* procedure obtains the territory value
> from the Oracle Workflow preferences table instead.

> **Note:** If you define custom views, and the WF_PREFERENCE
> resource token is set to FND, then you can map the Territory

column over the Oracle Workflow preferences table using the statement below. The benefit of doing so is that you can then globally set the default territory for all users in your enterprise and let individual users override that default value by changing their territory preference. See: Setting Global User Preferences, page 2-9 and get_pref, *Oracle Workflow Developer's Guide*.

```
NVL(wf_pref.get_pref(USR.USER_NAME,'TERRITORY'),
<default_territory>)
```

- Email_Address - A valid electronic mail address for this user or a mail distribution list defined by your electronic mail system.

- Fax - A fax number for the user.

- Orig_System - A code that you assign to originating system, which is the directory repository that this view is ultimately based on. For example, if this view is based on the personnel data stored in a Human Resource Management System, Orig_System can be defined as `PER`.

- Orig_System_ID - The primary key that identifies the user in this repository system. For example, Orig_System_ID can be defined as the value stored in a column called `PERSON_ID` in a Human Resources database table called `PER_PEOPLE`.

- Parent_Orig_System - An optional code for the originating system of an entity that you want to mark as being related to this user. For example, a supplier could be marked as the parent of a supplier site.

- Parent_Orig_System_ID - The primary key that identifies the parent entity in the parent originating system.

  > **Note:** If no parent information is provided, the Parent_Orig_System and Parent_Orig_System_ID default to the user's own Orig_System and Orig_System_ID, respectively.

- Start_Date - The date at which the user becomes valid in the directory service.

- Status - The availability of the user to participate in a workflow process. The possible statuses are: active (`ACTIVE`), unavailable for an extended period (`EXTLEAVE`), permanently unavailable (`INACTIVE`), and temporarily unavailable (`TMPLEAVE`). These statuses are also stored in the lookup type called `WFSTD_AVAILABILITY_STATUS`.

- Expiration_Date - The date at which the user is no longer valid in the directory service. After this date, the user will no longer appear in the seeded WF_USERS

view.

- Owner_Tag - A code to identify the program or application that owns the information for this user.

WF_ROLES

The WF_ROLES view references information about all the roles in your organization who may utilize Oracle Workflow functionality or receive workflow notifications. This view must contain the following required columns pertaining to the roles in your repository. Those columns that are preceded by an asterisk (*) are similar to the corresponding columns described for the WF_USERS view.

> **Important:** Each user identified by WF_USERS must also appear in the WF_ROLES view as a role. This is a requirement for Oracle Workflow.

> **Note:** If a user is a member of a role and the user information such as language and notification preference is different from the role information, the Expand Roles option for a notification addressed to the role determines whether the user information or the role information takes precedence. If the Expand Roles option is not checked and the Notification System delivers the notification to the role, the role information will override the user information. If Expand Roles is checked, however, then each user in the role will receive a separate copy of the notification, and the user information will override the role information.

> If a user has a notification preference of SUMMARY or SUMHTML, and the user is also a member of a multi-user role with a different notification preference such as MAILHTML, the Notification System will use the Expand Roles setting to determine whether to deliver the notification according to the role or user notification preference. However, even if Expand Roles is not checked and the notification preference of the role takes precedence, the notification will still appear in the user's summary message because the notification is part of the user's worklist.

- Name - The internal name of the role as referenced by the Workflow Engine and Notification System.

> **Important:** If you define custom views, the Name column must be sourced from a column that is no longer than 320 characters, and it is recommended that the internal name be all uppercase. If your source table does not have a column that meets these criteria, DO NOT use string functions to force these restrictions. Instead, define the Name column to be *<orig_system>*:*<orig_system_id>* so

that Oracle Workflow can reference the original base table where roles are stored and a unique role in that table. For example, "PER_POSITION:009" could represent a position whose ID is 009 and whose record is stored in the personnel table called PER_POSITION.

- *Display_Name

- *Description

- *Notification_Preference

- *Language

- *Territory

- Email_Address - If the e-mail address is null for a given role, notification mailers send an individual e-mail to each user within the role.

- *Fax

- *Orig_System

- *Orig_System_ID

- *Parent_Orig_System

- *Parent_Orig_System_ID

- *Start_Date

- *Status

- Expiration_Date - The date at which the role is no longer valid in the directory service. After this date, the role will no longer appear in the seeded WF_ROLES view.

- *Owner_Tag

WF_USER_ROLES

The WF_USER_ROLES view is an intersection of the users and roles in WF_USERS and WF_ROLES, showing which users are members of which roles.

> **Note:** A role can contain only individual users as its members. It cannot contain another role. However, roles can be related to each other in a

hierarchy so that users assigned to one role automatically inherit membership in its superior roles as well.

The WF_USER_ROLES view must contain the following required columns:

- User_Name - The internal name of the user as listed in the view WF_USERS.

- Role_Name - The internal name of the role as listed in the view WF_ROLES.

- User_Orig_System - A code that you assign to the user directory repository as listed in the view WF_USERS.

- User_Orig_System_ID - The primary key that identifies the user in the user directory repository as listed in the view WF_USERS.

- Role_Orig_System - A code that you assign to the role directory repository as listed in the view WF_ROLES.

- Role_Orig_System_ID - The primary key that identifies the role in the role directory repository as listed in the view WF_ROLES.

- Start_Date - The date at which the association of this user with this role becomes valid in the directory service.

- Expiration_Date - The date at which the association of this user with this role is no longer valid in the directory service. After this date, the user and role association will no longer appear in the seeded WF_USER_ROLES view.

- Assignment_Type - A code indicating how the user was assigned to membership in this role.
    - D - The user was directly assigned to this role.

    - I - The user inherited this role through membership in another role.

    - B - The user has both direct and inherited assignments to this role.

- Parent_Orig_System - An optional code for the originating system of an entity that you want to mark as being related to this user/role association.

- Parent_Orig_System_ID - The primary key that identifies the parent entity in the parent originating system.

WF_USER_ROLE_ASSIGNMENTS_V

The WF_USER_ROLE_ASSIGNMENTS_V view is an intersection of the users and roles in WF_USERS and WF_ROLES, that tracks how assignments of users to roles are made directly or inherited through role hierarchy relationships. The view shows only

currently active assignments.

The WF_USER_ROLE_ASSIGNMENTS_V view contains the following columns:

- User_Name - The internal name of the user as listed in the view WF_USERS.

- Role_Name - The internal name of the role as listed in the view WF_ROLES.

- Assigning_Role - The role from which the user is inheriting assignment to this role.

- Start_Date - The date at which the assignment of this user to this role becomes valid in the directory service.

- End_Date - The date at which the assignment of this user to this role is no longer valid in the directory service.

- Assignment_Type - The way in which the user was assigned to membership in this role, either `DIRECT` or `INHERITED`.

WF_ALL_ROLES_VL

The WF_ALL_ROLES_VL view contains role information similar to the WF_ROLES view. However, WF_ALL_ROLES_VL includes all roles, whether not yet valid, currently valid, or expired.

The WF_ALL_ROLES_VL view contains the following columns:

- Name - The internal name of the role.

- Display_Name - The display name of the role.

- Description - An optional description of the role.

- Notification_Preference - The method by which this role prefers to receive notifications.

- Language - The value of the database `NLS_LANGUAGE` initialization parameter that specifies the default language-dependent behavior of the role's notification session.

- Territory - The value of the database `NLS_TERRITORY` initialization parameter that specifies the default territory-dependent formatting used in the role's notification session.

- Email_Address - A valid electronic mail address for this role.

- Fax - A fax number for the role.

- Orig_System - A code that you assign to originating system on which this view is ultimately based.

- Orig_System_ID - The primary key that identifies the role in the originating system.

- Start_Date - The date at which the role becomes valid in the directory service.

- Status - The availability of the role to participate in a workflow process. The possible statuses are: active (`ACTIVE`), unavailable for an extended period (`EXTLEAVE`), permanently unavailable (`INACTIVE`), and temporarily unavailable (`TMPLEAVE`). These statuses are also stored in the lookup type called `WFSTD_AVAILABILITY_STATUS`.

- Expiration_Date - The date at which the role is no longer valid in the directory service.

- Owner_Tag - A code to identify the program or application that owns the information for this role.

- Created_By - Standard Who column.

- Creation_Date - Standard Who column.

- Last_Updated_By - Standard Who column.

- Last_Update_Date - Standard Who column.

- Last_Update_Login - Standard Who column.

WF_ALL_USER_ROLES

The WF_ALL_USER_ROLES view contains user/role association information similar to the WF_USER_ROLES view. However, WF_ALL_USER_ROLES includes all user/role associations, whether not yet valid, currently valid, or expired.

The WF_ALL_USER_ROLES view contains the following columns:

- User_Name - The internal name of the user.

- Role_Name - The internal name of the role .

- User_Orig_System - A code that you assign to the user directory repository.

- User_Orig_System_ID - The primary key that identifies the user in the user originating system.

- Role_Orig_System - A code that you assign to the role directory repository.

- Role_Orig_System_ID - The primary key that identifies the role in the role originating system.

- Parent_Orig_System - An optional code for the originating system of an entity that

you want to mark as being related to this user/role association.

- Parent_Orig_System_ID - The primary key that identifies the parent entity in the parent originating system.

- Assignment_Type - A code indicating how the user was assigned to membership in this role.

    - D - The user was directly assigned to this role.

    - I - The user inherited this role through membership in another role.

    - B - The user has both direct and inherited assignments to this role.

- Start_Date - The date at which the association of this user with this role becomes valid in the directory service.

- Expiration_Date - The date at which the association of this user with this role is no longer valid in the directory service.

- Owner_Tag - A code to identify the program or application that owns the information for the association of this user with this role.

- Created_By - Standard Who column.

- Creation_Date - Standard Who column.

- Last_Updated_By - Standard Who column.

- Last_Update_Date - Standard Who column.

- Last_Update_Login - Standard Who column.

WF_ALL_USER_ROLE_ASSIGNMENTS

The WF_ALL_USER_ROLE_ASSIGNMENTS view contains information about how assignments of users to roles are made directly or inherited through role hierarchy relationships, similar to the WF_USER_ROLE_ASSIGNMENTS_V view. However, WF_ALL_USER_ROLE_ASSIGNMENTS includes all user/role assignments, whether not yet valid, currently valid, or expired.

The WF_ALL_USER_ROLE_ASSIGNMENTS view contains the following columns:

- User_Name - The internal name of the user.

- Role_Name - The internal name of the role .

- Assigning_Role - The role from which the user is inheriting assignment to this role.

- Start_Date - The date at which the assignment of this user to this role becomes valid in the directory service.

- End_Date - The date at which the assignment of this user to this role is no longer valid in the directory service.

- Assignment_Type - The way in which the user was assigned to membership in this role, either `DIRECT` or `INHERITED`.

- Created_By - Standard Who column.

- Creation_Date - Standard Who column.

- Last_Updated_By - Standard Who column.

- Last_Update_Date - Standard Who column.

- Last_Update_Login - Standard Who column.

Step 4: Setting Up Additional Languages

The Oracle Workflow Web pages, your workflow definitions, and workflow notifications can be translated to the languages defined in your Oracle installation.

> **Note:** You can only display languages that require a multibyte character set if your database uses a character set that supports these languages, such as UTF8. For more information, see: Choosing a Character Set, *Oracle Database Globalization Support Guide*.

> **Note:** The path of the language-dependent Oracle Workflow resource file in Oracle Applications is `$FND_TOP/$APPLRSC/wf<language>.res`.

WF_LANGUAGES View:

To support additional languages, Oracle Workflow uses a view called WF_LANGUAGES that identifies the languages defined in your Oracle installation. This view is automatically created during installation. Oracle Workflow uses the WF_LANGUAGES view to create, in its translatable tables, a row for each language that maps to a row found in the corresponding non-translated base table.

The WF_LANGUAGES view includes the following columns:

- Code - The language code.

- Display_Name - The display name of the language.

- NLS_Language - The value of the Oracle NLS_LANGUAGE initialization parameter that specifies the default language-dependent behavior of a session.

- NLS_Territory - The value of the Oracle NLS_TERRITORY initialization parameter that specifies the default territory-dependant date and numeric formatting of a session.

- NLS_Codeset - The character set for the language.

- Installed_Flag - Flag to indicate if the language is installed and available for use.

See: *Oracle Database Globalization Support Guide*.

To Display Oracle Workflow Web Pages in Other Languages:

You select and install additional languages as part of the Oracle Applications installation. Users can set their language preference to an installed language through the Personal Homepage in order to view Oracle Applications Web pages, including Oracle Workflow pages, in that language. See: Selecting NLS Settings, *Installing Oracle Applications* and Set Preferences, *Oracle Applications User's Guide*.

To Create and View Workflow Definitions in Other Languages using Oracle Workflow Builder:

1. Set the NLS_LANG environment variable for the new language, territory, and encoded character set that you want to use for the workflow definition. For example, for Windows 2000, run the `regedit` command and locate the `NLS_LANG` setting under the `HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE` hierarchy. Double click on `NLS_LANG`. Then set the variable to the new value and save your work. Specify the value for NLS_LANG in the following format:

   `LANGUAGE_TERRITORY.CHARSET`

   For more information about setting NLS_LANG, see: Globalization Support, *Oracle Database Installation Guide*.

2. Start Oracle Workflow Builder. Create a translated version of your workflow definition and save it as a flat file (`.wft`), or open and view a workflow definition that is already translated.

   Note: Although you can enter and view property values for your workflow definitions in other languages, the Oracle Workflow Builder user interface is still displayed in English.

To Load Workflow Definitions in Other Languages to a Database:

1. Ensure that the WF_LANGUAGES view has been created in your workflow server. This view is automatically created during installation.

2. Ensure that the language you want is set up in the database. You select and install additional languages as part of the Oracle Applications installation. See: Selecting NLS Settings, *Installing Oracle Applications*.

3. Load the translated workflow definition to your workflow database using either the Workflow Definitions Loader or the Workflow Builder.

   - Before running the Workflow Definitions Loader program, you must set the NLS_LANG environment variable to the appropriate territory and character set for the workflow definition you want to load. The character set must match the character set encoding used to create the workflow definition file, which is determined by the NLS_LANG value that was set on the client PC before the `.wft` file was created in the Workflow Builder. For example, if the `.wft` file was created in the Japanese native character set encoding `JA16SJIS`, then you must specify `JA16SJIS` in the character set portion of NLS_LANG before loading the file, and you cannot specify a different character set such as `UTF8`.

     To set NLS_LANG before running the Workflow Definitions Loader, use the following format:

     `_TERRITORY.CHARSET`

     Note that it is important to include the underscore (_) before the territory name and the period (.) between the territory name and the character set name in the NLS_LANG value. For example, if the `.wft` file was created in the Japanese native character set encoding `JA16SJIS`, set NLS_LANG to the following value:

     `_JAPAN.JA16SJIS.`

     You do not need to include the language in this NLS_LANG value because the Workflow Definitions Loader uses the language specified within the `.wft` file to determine the language to load. See: Using the Workflow Definitions Loader, page 8-8.

     > **Note:** If you create all your translated workflow definition files in Unicode encoding, you can simply set NLS_LANG to `.UTF8` before loading these files. In this case you will not need to reset NLS_LANG for translated files in different languages, because the `.UTF8` character set applies to all the files in Unicode encoding.

   - Before using the Workflow Builder to save a translated workflow definition to

the database, you must set the NLS_LANG environment variable to the appropriate language, territory, and character set. If you are saving several workflow definitions in different languages, you must reset NLS_LANG for each language. See: Opening and Saving Item Types, *Oracle Workflow Developer's Guide*.

> **Note:** The translated versions of Oracle Workflow's standard and demonstration workflow definitions are provided in native character set encoding, not in UTF8.

To Send E-mail Notifications in Other Languages:

1. Determine whether Oracle has translated the e-mail notification templates to the language you wish to set by checking for the file containing the templates in the appropriate language subdirectory, `$FND_TOP/import/<lang>`. The standard templates are delivered in a file called `wfmail.wft`. See: Modifying Your Message Templates, page 2-84.

2. If the e-mail templates are available for the desired language, Oracle Workflow uses the language preference for the notification recipient to determine the language for an e-mail notification.

   Oracle Applications users can set their language preference in the Preferences page. This preference is also stored in the ICX: Language profile option. See: Set Preferences, *Oracle Applications User's Guide*.

   > **Note:** Users can select a session-level language in the Oracle Applications login window, which overrides their user-level language preference for that session. However, Oracle Workflow still uses the user-level language preference to determine the language in which e-mail notifications are sent.

Step 5: Setting Up Background Workflow Engines

When the Workflow Engine initiates and performs a process, it completes all necessary activities before continuing to the next eligible activity. In some cases, an activity can require a large amount of processing resource or time to complete. Oracle Workflow lets you manage the load on the Workflow Engine by setting up supplemental engines to run these costly activities as background tasks. In these cases, the costly activity is *deferred* by the Workflow Engine and run later by a background engine. The main Workflow Engine can then continue to the next available activity, which may occur on some other parallel branch of the process. A workflow process can also include a Wait activity, which defers the continuation of a process until a later time. This type of deferred activity is also completed by a background engine.

A background engine must also be set up to handle timed out notification activities. When the Workflow Engine comes across a notification activity that requires a response, it calls the Notification System to send the notification to the appropriate performer, and then sets the notification activity to a status of 'NOTIFIED' until the performer completes the notification activity. Meanwhile, a background engine set up to handle timed out activities periodically checks for 'NOTIFIED' activities and whether these activities have time out values specified. If a 'NOTIFIED' activity does have a time out value, and the current date and time exceeds that time out value, the background engine marks that activity as timed out and calls the Workflow Engine. The Workflow Engine then resumes by trying to execute a `<Timeout>` transition activity.

Additionally, a background engine must be set up to handle stuck processes. A process is identified as stuck when it has a status of `ACTIVE`, but cannot progress any further. For example, a process could become stuck in the following situations:

- A thread within a process leads to an activity that is not defined as an End activity but has no other activity modeled after it, and no other activity is active.

- A process with only one thread loops back, but the pivot activity of the loop has the On Revisit property set to `Ignore`.

- An activity returns a result for which no eligible transition exists. For instance, if the function for a function activity returns an unexpected result value, and no default transition is modeled after that activity, the process cannot continue.

The background engine sets the status of a stuck process to `ERROR:#STUCK` and executes the error process defined for it.

The following table lists the standard queues used in background engine processing.

Background Engine Queues

Queue Table	Queue Name	Payload Type	Retention Time	Description
WF_DEFERRED _QUEUE_M	WF_DEFERRED _QUEUE_M	SYSTEM. WF_PAYLOAD_ T	0 days	Standard background deferred queue
WF_OUTBOUN D_QUEUE	WF_OUTBOUN D_QUEUE	SYSTEM. WF_PAYLOAD_ T	0 days	Standard background outbound queue
WF_INBOUND_ QUEUE	WF_INBOUND_ QUEUE	SYSTEM. WF_PAYLOAD_ T	0 days	Standard background inbound queue

See: Workflow Queue APIs, *Oracle Workflow API Reference*.

You can define and start up as many background engines as you like to check for deferred and timed out activities.

Background engines can be restricted to handle activities associated with specific item types, and within specific cost ranges. A background engine runs until it completes all eligible activities at the time it was initiated.

Generally, you should set the background engine up to run periodically by scheduling the Background Process concurrent program to resubmit periodically.

Ensure that you have at least one background engine that can check for timed out activities, one that can process deferred activities, and one that can handle stuck processes. At a minimum, you need to set up one background engine that can handle both timed out and deferred activities as well as stuck processes. However, for performance reasons we recommend that you run three separate background engines at different intervals.

- Run a background engine to handle only deferred activities every 5 to 60 minutes.

- Run a background engine to handle only timed out activities every 1 to 24 hours as needed.

- Run a background engine to handle only stuck processes once a week to once a month, when the load on the system is low.

To Schedule Background Engines:

You can submit the background engine procedure as a concurrent program, which lets you schedule different background engines to run at different times. Use the Submit Requests window in Oracle Applications to submit the Workflow Background Process. See: Overview of Concurrent Programs and Requests, *Oracle Applications System Administrator's Guide*.

Additionally, you can use the Oracle Workflow Manager component of Oracle Applications Manager to submit and manage the Workflow Background Process concurrent program. See: Background Engines, page 7-78.

> **Note:** Make sure you have a least one background engine that can check for timed out activities, one that can process deferred activities, and one that can handle stuck processes. At a minimum, you need to set up one background engine that can handle both timed out and deferred activities as well as stuck processes.

> **Note:** If you require a larger rollback segment for the Workflow Background Process than the default, you can use the Concurrent Programs window in the System Administrator responsibility to specify the rollback segment that you want. This rollback segment will

be used instead of the default and will be used up until the first commit.

Query the Workflow Background Process concurrent program (FNDWFBG) in the Concurrent Programs window, and choose the Session Control button. Then in the Session Control window, enter the rollback segment you want in the Rollback Segment field, and save your work. See: Concurrent Programs Window, *Oracle Applications System Administrator's Guide*.

To Run a Workflow Background Process as a Concurrent Program

1. Navigate to the Submit Requests form.

2. Submit the Workflow Background Process concurrent program as a request. See: Running Reports and Programs, *Oracle Applications User's Guide*.

3. In the Parameters window, enter values for the following parameters:

Item Type	Specify an item type to restrict this engine to activities associated with that item type. If you do not specify an item type, the engine processes any deferred activity regardless of its item type.
Minimum Threshold	Specify the minimum cost that an activity must have for this background engine to execute it, in hundredths of a second.
Maximum Threshold	Specify the maximum cost that an activity can have for this background engine to execute it, in hundredths of a second.
	By using Minimum Threshold and Maximum Threshold you can create multiple background engines to handle very specific types of activities. The default values for these arguments are null so that the background engine runs activities regardless of cost.
Process Deferred	Specify whether this background engine checks for deferred activities. Setting this parameter to `'Yes'` allows the engine to check for deferred activities.
Process Timeout	Specify whether this background engine checks for activities that have timed out. Setting this parameter to `'Yes'` allows the engine to check for timed out activities.

Process Stuck	Specify whether this background engine checks for stuck processes. Setting this parameter to `'Yes'` allows the engine to check for stuck processes.

> **Note:** Make sure you have a least one background engine that can check for timed out activities, one that can process deferred activities, and one that can handle stuck processes.

4. Choose OK to close the Parameters window.

5. When you finish modifying the run options to define the schedule for the background engine, choose Submit to submit the request.

To Set Engine Thresholds:

To set the thresholds of background engines, specify the minimum threshold and maximum threshold arguments when starting the engine. The background engine then only processes activities with costs within your specifications.

The Workflow Engine threshold is set to 50 as a default. Activities with a cost higher than 50 are deferred for background engines to process.

In some cases, you may want to force the engine to defer an activity although the activity's cost is less than fifty. You can do this by altering the Workflow Engine threshold in the PL/SQL stored procedure for a function activity.

The engine threshold is set in an externalized constant called `THRESHOLD`. Include the following line in your PL/SQL procedure to set the Workflow Engine threshold to a different value:

```
WF_ENGINE.THRESHOLD := n;
```

You should reset the threshold value afterwards in SQL*Plus or in the next function activity so that other activities are processed as expected.

Related Topics

Activity Cost, *Oracle Workflow Developer's Guide*

Timeout Transitions, *Oracle Workflow Developer's Guide*

Deferring Activities, page C-6

Wait Activity, *Oracle Workflow Developer's Guide*

Background, *Oracle Workflow API Reference*

Step 6: Implementing Notification Mailers

A notification mailer is a Java program that performs e-mail send and response

processing for the Oracle Workflow Notification System, using the JavaMail API. You need to implement one or more notification mailers only if you want to have your workflow users receive their notifications by e-mail, as well as from the Worklist Web pages.

See: Reviewing Notifications via Electronic Mail, *Oracle Workflow User's Guide*.

Managing Notification Mailers:

The notification mailer program is defined as a service component type in the Generic Service Component Framework. This framework helps to simplify and automate the management of background Java services. For more information about managing service components, see: Service Components, page 7-7.

Oracle Workflow provides one seeded notification mailer service component, called Workflow Notification Mailer. Most of the configuration parameters for this mailer are set to default values. You can enter several of the remaining required parameters using AutoConfig. After installation, you then only need to enter the e-mail inbox password to complete the configuration of the Workflow Notification Mailer. Alternatively, if you only want to send outbound messages and do not need to receive inbound messages, you only need to disable inbound processing to complete the configuration of the Workflow Notification Mailer.

If the mail servers and Business Event System components used by notification mailers are set up, and the service component container to which the Workflow Notification Mailer belongs is started, the seeded notification mailer automatically starts running once its configuration is complete.

You cannot delete the seeded Workflow Notification Mailer or edit its name, assigned agents, correlation ID value, or container. However, if necessary you can optionally update other configuration parameters, schedule control events, or manually choose control commands to start, stop, suspend, resume, or refresh this notification mailer.

> **Note:** Oracle Alert also uses the Workflow Notification Mailer to send and receive alert e-mail messages. If you use Oracle Alert, ensure that the configuration of the Workflow Notification Mailer meets your alert requirements. See: Setup Steps, *Oracle Alert User's Guide*.

You can also optionally create additional notification mailer service components. For example, you can create a notification mailer that processes only messages that belong to a particular workflow item type, or create additional mailers that process the same types of message to increase throughput.

You can also configure any notification mailer service component to process only inbound messages, or only outbound messages. You associate inbound and outbound mailers with each other by assigning them the same mailer node name. The mailer node name indicates which inbound mailer can process incoming responses to outbound messages sent by a particular outbound mailer.

You can optionally assign the same node name to multiple mailers for load balancing

purposes. However, each mailer that performs inbound processing for a node must have its own inbox.

- If you enable both outbound and inbound processing for the same mailer, that mailer will automatically use the same node name for both types of processing, enabling it to process incoming responses to the outbound messages it sent. You can optionally also create other notification mailers that share the same node name.

- If you create an outbound-only mailer, but you still want to perform response processing for e-mail responses to the outbound messages it sends, you should create at least one other mailer with the same node name that does perform inbound message processing. Otherwise, there will be no inbound mailer that can process incoming responses to outbound messages sent by this outbound mailer.

- If you only want to implement outbound message processing, without inbound e-mail response processing, then you can configure an outbound-only mailer without creating a corresponding inbound mailer. In this case you should configure the mailer to use message templates for response-required notifications that do not request a response by e-mail, but instead direct recipients to respond from the Notification Details Web page. For example, you can configure the mailer to send response-required notifications using the Workflow View From UI message template, which is an alternative template provided by Oracle Workflow in the System: Mailer item type, or create your own custom message templates. The outbound-only mailer can still use the standard message templates to send outbound summary notifications or For Your Information (FYI) notifications that do not require a response.

- Create an inbound-only mailer only if you have also created at least one mailer with the same node name that performs outbound message processing. If no outbound mailer shares the same node name, no incoming response messages will be marked with that node name, and the inbound-only mailer will have no messages to process.

  > **Note:** The node name for each node must be unique. However, multiple mailers can share the same node. The maximum length for a node name is eight characters, and the node name cannot include any spaces or any of the following characters: left bracket (`[`), right bracket (`]`), slash (`/`), or at sign (`@`).

Service components must be hosted by a service component container. If you create custom notification mailer service components, you can assign them to the seeded container for notification mailers.

A service component container is implemented as a Generic Service Management (GSM) service. The seeded container for notification mailers is named `Workflow Mailer Service`.

Based on the volume to be handled by the seeded container, you can also choose to create your own custom containers as GSM services in Oracle Applications Manager. If you create a custom GSM service in OAM, you can copy the service parameters from the seeded Workflow Mailer Service to your new service in order to specify how to run the new service.

Setting Up Notification Mailers:

Currently, Oracle Workflow supports the Simple Mail Transfer Protocol (SMTP) for outbound messages and the Internet Message Access Protocol (IMAP) for inbound messages. You must have an SMTP server set up in order to send Oracle Workflow notification e-mail messages, and an IMAP server set up if you want to receive e-mail notification responses. Users can receive e-mail notifications using various e-mail clients, although notifications may be displayed differently in different clients, depending on the features each client supports.

> **Note:** Oracle Workflow supports IMAP version 4 (IMAP4) compliant mail servers. Ensure that your mail server uses this IMAP version. For more information, see the *JavaMail API Design Specification*: http://java.sun.com/products/javamail/JavaMail-1.2.pdf

> **Note:** If you have certain types of software installed, you may already have the necessary mail server functionality available. For example, products such as Oracle Email, Microsoft Exchange, or Lotus Notes include IMAP services. You can use a UNIX server as an SMTP server by configuring the Sendmail program.
>
> Additionally, you can choose to use IMAP server software that is available for download from some sources. For example, the University of Washington offers the UW IMAP Server as a public service, and Carnegie Mellon University offers the Cyrus IMAP Server. You might choose this option if your enterprise uses UNIX Sendmail e-mail accounts, for instance. For more information, see: http://www.washington.edu/imap/, http://asg.web.cmu.edu/cyrus/, and http://www.imap.org/.

> **Note:** Third party software products are mentioned as examples only. Oracle makes no recommendation or endorsement of these third party software products.

Use the Oracle Workflow Manager component of Oracle Applications Manager (OAM) to configure and run notification mailers. For more information, see: Notification Mailers, page 7-19.

To set up a notification mailer, you must perform the following steps.

To Set Up a Notification Mailer

1. Set up an SMTP mail server to send outbound messages.

2. Set up an IMAP4 compliant mail server with an e-mail account for the notification mailer if you want to receive inbound messages.

   The notification mailer requires three folders in this e-mail account: the inbox, a folder to store processed messages, and a folder to store discarded messages.

   If the e-mail account does not already include folders named PROCESS and DISCARD, Oracle Workflow automatically creates these two folders when you complete the basic notification mailer configuration. You can optionally specify other folders for the notification mailer using the advanced configuration wizard.

   > **Note:** Use your e-mail client to create folders manually for the notification mailer to use. A notification mailer may not be able to access folders that were created using command line tools outside the e-mail client.

3. You can enter the following configuration parameters for the seeded Workflow Notification Mailer service component during installation using AutoConfig. For more information about running AutoConfig, see *Using AutoConfig to Manage System Configurations with Oracle Applications Release 12*, Oracle*MetaLink* note 387859.1 and AutoConfig, *Oracle Applications Concepts*.

   - SMTP server

   - IMAP server (if you want to receive inbound messages)

   - Inbox username (if you want to receive inbound messages)

   - Reply to e-mail address (if you want to receive inbound messages)

   - HTML agent name (defaults to the value you enter for the Applications Servlet Agent parameter)

     > **Note:** When you enter the SMTP Server and IMAP Server parameters, specify the actual host name for each server. Do not use localhost as the setting for these parameters. You can optionally specify the port number to use on each server. If you do not specify a port number, the notification mailer uses port 143 on the IMAP server and port 25 on the SMTP server by default. Specify each server in the following format:
     >
     > *<server_name>*[:*<port_number>*]

4. Ensure that the Business Event System status is set to `Enabled` in the global workflow preferences, and that the `JOB_QUEUE_PROCESSES` database initialization parameter, which is required for the Business Event System, is set to an appropriate value. The Business Event System status is set to `Enabled` by default, and usually you do not need to change this status. If notification processing is not being completed, however, you should check this preference value.

5. **(Recommended)** You can optionally set the WF: Workflow Mailer Framework Web Agent profile option to the host and port of the Web server that notification mailers should use to generate the content for Oracle Application Framework regions that are embedded in notifications. If this profile option is not set, notification mailers will use the same Web agent specified in the Application Framework Agent profile option. However, if necessary for load balancing purposes, you can optionally specify a different Web agent for notification mailers to use. The WF: Workflow Mailer Framework Web Agent profile option should be set at site level. See: Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide*.

6. Before a service component can run, the container which manages it must first be started. The seeded Workflow Notification Mailer service component belongs to a container named `Workflow Mailer Service`. The seeded agent listener service components that are also required for notification mailer processing belong to a container named `Workflow Agent Listener Service`. You should ensure that these two containers are running, using Oracle Applications Manager. If you create your own custom containers in OAM for custom service components, ensure that those containers are running as well.

   > **Note:** You can run a diagnostic test to verify the GSM services for Oracle Workflow. See: Oracle Workflow Diagnostic Tests, page E-1.

7. When the Workflow Agent Listener Service or WFALSNRSVC container is running, it automatically starts seeded agent listener service components named Workflow Deferred Notification Agent Listener, Workflow Error Agent Listener, and Workflow Inbound Notifications Agent Listener, which are required for notification mailer processing. Ensure that these agent listeners are running.

8. Use the notification mailer configuration wizard to configure your notification mailer service component.

   > **Note:** The Basic Configuration page lets you configure a notification mailer quickly by entering only the minimum required parameters, while the advanced configuration wizard lets you specify additional parameters to control how the notification mailer processes messages. See: Notification Mailer Configuration Wizard,

page 7-30.

If you entered configuration parameters for the seeded Workflow Notification Mailer in AutoConfig, you only need to enter the password for the e-mail inbox in order to complete the configuration for that mailer and begin running it.

If you did not enter parameters for the seeded mailer during installation, then in order to complete the configuration for that mailer you need to enter only the SMTP server, IMAP server, e-mail inbox username, e-mail inbox password, and reply-to e-mail address. All other configuration parameters for the seeded Workflow Notification Mailer are initially set to default values and do not need to be changed, although you can optionally do so if you choose.

> **Note:** The IMAP server, e-mail inbox username, e-mail inbox password, and reply-to e-mail address are required only if you want to receive inbound messages. Alternatively, if you only want to send outbound messages and do not need to receive inbound messages, you only need to disable inbound processing in order to complete the configuration of the Workflow Notification Mailer. To do so, deselect the Inbound Processing parameter in the Basic Configuration page.

9. **(Optional)** By default, the seeded Workflow Notification Mailer has a Launch Summary Notifications event scheduled to send summary notifications once a day. You can optionally use the notification mailer configuration wizard to modify the start time and interval for this event's schedule, or to schedule the Launch Summary Notifications event at the interval you choose for any notification mailer service component. When this event is processed, a summary notification is sent to each role with a notification preference of SUMMARY or SUMHTML, listing all the notifications that are currently open for that role.

10. **(Optional)** You can optionally configure a notification mailer to connect to the SMTP server and IMAP server through Secure Sockets Layer (SSL) to encrypt the data exchanged. See: Connecting to Mail Servers Through SSL, page 2-56.

11. **(Optional)** You can optionally set the internal mailer parameter named HTML_DELIMITER to specify which characters the notification mailer uses to delimit response values in response templates for HTML-formatted e-mail notifications. Valid values for the HTML_DELIMITER parameter are:

   • DEFAULT - The notification mailer uses the default delimiters, currently set as the single quote (') for both the opening and the closing delimiter. The notification mailer also uses the default delimiters if the HTML_DELIMITER parameter value is left null.

- `APOS` - The notification mailer uses the single quote, or apostrophe (`'`), as both the opening and the closing delimiter. This setting is currently the same as the default.

- `QUOTE` - The notification mailer uses the double quote (`"`) as both the opening and the closing delimiter.

- `BRACKET` - The notification mailer uses the left bracket (`[`) as the opening delimiter and the right bracket (`]`) as the closing delimiter.

Using single quotes as the delimiters accommodates e-mail applications that cannot process double quotes in the <A HREF="mailto:"> tag for the response template link, but can accept single quotes. However, if you want users to be able to use apostrophes or single quotes in their response values without entering an escape character, you can use double quotes or brackets as the delimiters, depending on what your e-mail application supports. See: To Respond to an HTML E-mail Notification, *Oracle Workflow User's Guide*.

> **Note:** If the `HTML_DELIMITER` parameter is set to an invalid value, the notification mailer throws an exception at startup. Any notifications created during this time are rendered with the default delimiters instead.

By default, the `HTML_DELIMITER` parameter is set to the value `DEFAULT`. Use the `afsvcpup.sql` script to change the parameter value to specify the delimiters you want to use. See: To Set Internal Mailer Parameters, page 2-50.

If a particular notification message has the special `#WFM_HTML_DELIMITER` message attribute defined, however, the notification mailer will use the `#WFM_HTML_DELIMITER` attribute value to determine which delimiters to use for that notification, instead of using the `HTML_DELIMITER` parameter value.

> **Note:** The `HTML_DELIMITER` parameter only controls the response templates for HTML-formatted notifications. This parameter does not apply to plain text notifications.

12. **(Optional)** The seeded Workflow Notification Mailer uses the Automatic startup mode by default and will be started automatically when you complete its configuration. If you select the Manual startup mode for a notification mailer service component, use the Service Components page in Oracle Workflow Manager to start that notification mailer. You can also use this page to manage any notification mailer service component.

To Set Internal Mailer Parameters
Use the `afsvcpup.sql` script to set internal mailer parameters that do not appear in

the notification mailer configuration wizard. This script is located in the
`$FND_TOP/sql` directory.

1. Use the following command to run the `afsvcpup.sql` script:

   ```
   sqlplus <user/pwd> @afsvcpup
   ```

2. At the prompts, enter the component ID for your notification mailer service component, the parameter ID for the parameter to set, and the value to assign to that parameter. You can find the IDs to enter in the lists displayed by the script, which show first the service components defined in your installation of Oracle Workflow and then the parameters defined for the specified service component. You can also find the component ID for a notification mailer in the Define page of the configuration wizard.

Outbound Notification Mailer Processing:

When the Workflow Engine determines that a notification message must be sent, it raises an event in the Business Event System called `oracle.apps.wf.notification.send`. Oracle Workflow provides a seeded subscription to this event, which is defined to be deferred immediately so that the workflow process that owns the notification can continue. The event is placed on the standard WF_DEFERRED agent. Oracle Workflow provides a seeded agent listener named Workflow Deferred Notification Agent Listener that runs on this agent to continue notification processing. This agent listener is dedicated solely to processing deferred notification events.

When the event is dequeued from WF_DEFERRED and the subscription is processed, the subscription requires the event data for the event, causing the generate function for the event to be executed. The generate function for this event performs the following actions:

- Resolves the notification recipient role to a single e-mail address, which itself can be a mail list.

- Checks the notification preference of the recipient to determine whether an e-mail notification is required, and in what type of format.

- Switches its database session to the recipient role's preferred language and territory as defined in the directory service.

- Generates an XML representation of the notification message and any optional attachments using the appropriate message template.

Finally, the subscription places the event message on the standard WF_NOTIFICATION_OUT agent.

A notification mailer service component polls the WF_NOTIFICATION_OUT agent for messages that must be sent by e-mail. When the notification mailer dequeues a message from this agent, it uses a Java-based notification formatter to convert the XML

representation of the notification into a MIME (Multi-purpose Internet Mail Extensions) encoded message and sends the message by the Simple Mail Transfer Protocol (SMTP).

Outbound Notification Mailer Processing
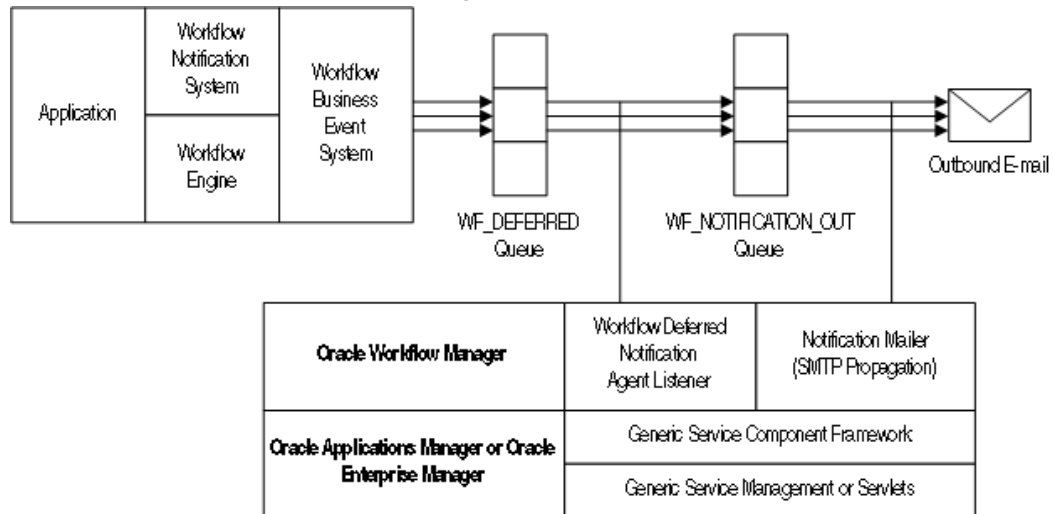
The e-mail notifications are based on message templates defined in Oracle Workflow Builder. Oracle Workflow provides a set of standard templates in the System: Mailer item type, which are used by default. It is not recommended to modify the standard templates. However, you can customize the message templates used to send your e-mail notifications by creating your own custom message templates in a custom item type using the Workflow Builder. Then assign these templates to a particular notification in a workflow process by defining special message attributes. In this case the templates assigned to the notification override any other templates. See: Modifying Your Message Templates, page 2-84 and Notification Mailer Message Template Attributes, *Oracle Workflow Developer's Guide*.

You can also create your own custom message templates in the System: Mailer item type using the Workflow Builder, and assign these templates to a particular notification mailer service component in the mailer configuration parameters. The templates assigned to a mailer override the default System: Mailer templates. However, if any notifications have templates specifically assigned to them through message attributes, the notification-level templates still override the templates assigned to the mailer. See: Modifying Your Message Templates, page 2-84.

If the notification mailer cannot deliver an e-mail notification because the recipient's e-mail address is invalid, it performs the following actions:

• Sets the mail status of the notification to FAILED. This mail status indicates that an exception prevented this e-mail notification from being delivered but does not prevent the mailer from processing other notifications.

• Adds the e-mail address to its invalid e-mail address list. To avoid unnecessary

processing, each notification mailer stores a list of e-mail addresses to which it could not deliver messages, and does not attempt to send any further messages to those addresses. Instead, for any subsequent notifications to the listed addresses, the notification mailer simply sets the mail status directly to FAILED.

> **Note:** Each notification mailer can store up to 100 e-mail addresses in its invalid e-mail address list. If the notification mailer encounters additional invalid addresses when the list is already full, the notification mailer removes the oldest addresses from the list and adds the new addresses in their place. Also, the notification mailer clears the list by removing all addresses whenever you stop and restart the mailer.

- Changes the notification preference of the recipient to DISABLED. To further help avoid unnecessary processing, if a recipient has a notification preference of DISABLED, Oracle Workflow does not generate a complete XML representation of any notifications to that recipient, and a notification mailer does not attempt to send e-mail notifications to that recipient. Instead, the notification mailer simply sets the mail status of the notifications directly to FAILED. The change in notification preference also indicates to the user that e-mail notifications cannot be delivered. The user must correct the invalid e-mail address and then reset the notification preference in order to receive e-mail notifications.

- Sends a notification to the SYSADMIN user with the information that an e-mail notification could not be sent to one or more recipients, that the notification preference for those recipients has been set to DISABLED, and that those recipients' original notification preferences, which are listed, should be reset after the issues that caused the failures are corrected. See: User Notification Preference Update Report Message, page 2-134.

After correcting invalid e-mail addresses and resetting DISABLED notification preferences, you can run the Resend Failed Workflow Notifications concurrent program to retry open notifications that previously could not be sent. See: Handling Mailer Errors, page 2-73.

Inbound Notification Mailer Processing:

Notification mailers can also process e-mail responses from users, using the Internet Message Access Protocol (IMAP). A notification mailer uses a Java-based e-mail parser to interpret the text of each message and create an XML representation of it.

A notification mailer uses three folders in your response mail account for response processing: one to receive incoming messages, one to store processed messages, and one to store discarded messages.

A notification mailer does the following to process response messages:

- Logs into its IMAP e-mail account.

- Checks the inbox folder for messages. If a message exists, the notification mailer reads the message, checking for the notification ID (NID) and node identifier in the NID line.

  - If the message is not a notification response, meaning it does not contain an NID line, the notification mailer moves the message to the discard folder and treats it as an unsolicited message. For the first unsolicited message from a particular e-mail address, the notification mailer also sends a warning message back to the sender of the message.

    However, to avoid sending unnecessary warnings due to bounced or auto-reply messages, each mailer node stores a list of e-mail addresses from which it has received unsolicited mail, and does not send any further warning messages to those addresses. Instead, if the node receives a second unsolicited message from a particular address, the notification mailer discards the message and raises the `oracle.apps.wf.mailer.unsolicited` event. You can optionally define a subscription to this event if you want to perform some other action in response to the second unsolicited message. For all subsequent unsolicited messages, the notification mailer simply discards the message.

    > **Note:** Each mailer node can store up to 100 e-mail addresses in its warned list. If the node receives unsolicited messages from additional addresses when the list is already full, the notification mailer removes the oldest addresses from the list and adds the new addresses in their place. Also, the notification mailer clears the list by removing all addresses when you start the mailer for the first time, and again whenever you stop and restart its container. In these cases, the mailer may send another warning message if it receives further unsolicited e-mail from an address that is no longer on the warned list.

    > **Note:** You can optionally use the Send Warning for Unsolicited E-mail mailer parameter to prevent notification mailers from sending any warning messages at all. See: Notification Mailer Configuration Wizard, page 7-30.

  - If the message is a notification response, but for a different node, the notification mailer leaves the message in the inbox and adds the e-mail's Unique Message ID (UID) to its ignore list.

  - If the message is a notification response for the current node, meaning it contains an NID line including the node identifier of the current node, the

notification mailer processes the message.

The notification mailer performs the following steps for messages that belong to its node.

- Retrieves the notification ID.

- Checks to see if the message bounced by referring to the tags specified in the configuration parameters, if any. If the message bounced, the notification mailer updates the notification's status and stops any further processing, based on the specifications of the tag list.

- Checks the Oracle Workflow database for this notification based on the NID line.

  - If the notification does not exist, meaning the notification ID or the access key in the NID line is invalid, the notification mailer moves the message to the discard folder. If the NID line is incorrectly formatted, the notification mailer moves the message to the discard folder and and treats it as an unsolicited message.

  - If the notification exists, but is closed or canceled, the notification mailer moves the message to the processed folder and sends a Workflow Closed Mail or Workflow Canceled Mail message to the recipient role, respectively.

    > **Note:** You can optionally use the Send E-mails for Canceled Notifications mailer parameter to prevent notification mailers from sending any notification cancellation messages. See: Notification Mailer Configuration Wizard, page 7-30.

  - If the notification exists and is open, the notification mailer generates an XML representation of the message and places it on the standard WF_NOTIFICATION_IN agent as an event called `oracle.apps.wf.notification.receive.message`. The notification mailer then moves the e-mail message to the processed folder.

    > **Note:** If the character encoding of the response message is not compatible with the database codeset, the notification mailer may not be able to parse the response and recognize the response values. Ensure that the character encoding of messages in your mail client is compatible with the codeset of your database.

Finally, if there are no more unprocessed messages in the inbox, the notification mailer logs out of the e-mail account.

Oracle Workflow provides a seeded agent listener named Workflow Inbound Notifications Agent Listener that runs on the WF_NOTIFICATION_IN agent to continue notification processing for the valid response messages placed on that agent.

When an event message is dequeued from WF_NOTIFICATION_IN, Oracle Workflow executes a seeded subscription that calls the appropriate notification response function. This function verifies the response values with the definition of the notification message's response attributes in the database. If a response value is invalid, or if no response value is included, the notification response function causes the notification mailer to send a Workflow Invalid Mail message to the recipient role. If the responses are valid, the notification response function records the response and completes the notification.

Inbound Notification Mailer Processing
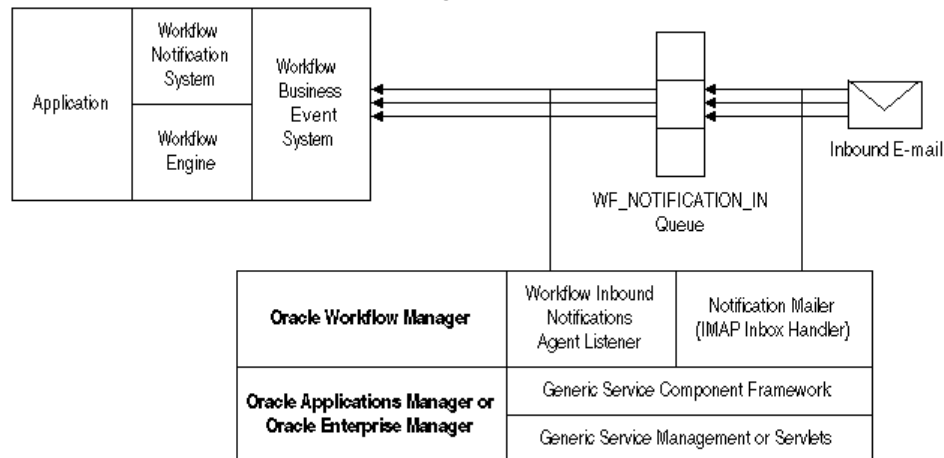
See: Workflow Warning Mail Message, page 2-118, Workflow Closed Mail Message, page 2-115, Workflow Canceled Mail Message, page 2-107, Workflow Invalid Mail Message, page 2-109, and Workflow Invalid Open Mail (More Information Request) Message, page 2-131.

Full MIME Support:

Oracle Workflow fully supports MIME (Multi-purpose Internet Mail Extensions) encoded messages. This means that notification mailers can exchange messages with workflow users containing languages with different character sets and multi-media encoded content.

Connecting to Mail Servers Through SSL:

You can optionally configure a notification mailer to connect to the SMTP server and IMAP server through the Secure Sockets Layer (SSL) protocol. SSL provides encrypted connections for sending data between the notification mailer and the mail servers, for enhanced security.

To use SSL, you must have an X.509 certificate and private key issued by a certificate authority. You can use the same certificate for both the SMTP server and the IMAP server.

> **Note:** If you will use this certificate solely for testing purposes or to secure communications within your own enterprise, you can set up your own certificate authority to issue the certificate. However, to secure communications with third parties, you must obtain a certificate from a publicly recognized certificate authority.

Additionally, to connect to the SMTP server through SSL, you must have Stunnel installed on the SMTP server. Stunnel is a program that lets you encrypt connections inside SSL. For more information, see: http://www.stunnel.org/.

To Set Up for SSL Connections to the SMTP Server

1. Obtain an X.509 certificate and private key from a certificate authority, and load the file containing the certificate and key onto the file system of the SMTP server. For detailed instructions, please refer to the documentation for your SMTP server.

2. Start an Stunnel process on the SMTP server, specifying the location of the certificate file in the arguments.

   For example, if the certificate file is named `imapd.pem` and is located in the `/usr/share/ssl/certs/` directory on the SMTP server, use the following Stunnel command to redirect SSL connections from port 465 to port 25 on the SMTP server.

   ```
   stunnel -d 465 -r 25 -o /tmp/stunnel.log -p
   /usr/share/ssl/certs/imapd.pem
   ```

   For more information, see the Stunnel documentation.

3. In the notification mailer configuration wizard, select the Outbound SSL Enabled parameter to enable the notification mailer to use SSL for connections to the SMTP server. See: Notification Mailer Configuration Wizard, page 7-30.

   > **Note:** When you enable SSL, the notification mailer connects to the SMTP server through port 465 by default. You can optionally specify a different port number along with the SMTP server name in the Outbound E-mail Account: Server Name parameter in the notification mailer configuration wizard.

4. **(Conditionally Required)** If you set up your own certificate authority to issue the certificate, then you must also load the certificate into a local trust store.

   - If possible, load the certificate into the default local trust store on the concurrent manager host where the notification mailer resides.

   - Otherwise, load the certificate to an alternate trust store, and specify the location of that trust store in the internal mailer parameter named `MAILER_SSL_TRUSTSTORE`.

By default, the `MAILER_SSL_TRUSTSTORE` parameter is set to `NONE`, which causes the notification mailer to use the default Java security configuration. Use the `afsvcpup.sql` script to change the parameter value to the location of the alternate trust store. See: To Set Internal Mailer Parameters, page 2-50.

To Set Up for SSL Connections to the IMAP Server

5. If the IMAP server is located on a different host than the SMTP server, load the file containing the X.509 certificate and private key onto the file system of the IMAP server. For detailed instructions, please refer to the documentation for your IMAP server.

6. In the notification mailer configuration wizard, select the Inbound SSL Enabled parameter to enable the notification mailer to use SSL for connections to the IMAP server. See: Notification Mailer Configuration Wizard, page 7-30.

> **Note:** When you enable SSL, the notification mailer connects to the IMAP server through port 993 by default. You can optionally specify a different port number along with the IMAP server name in the Inbound E-mail Account: Server Name parameter in the notification mailer configuration wizard.

To Begin Using SSL Connections

7. After completing the SSL setup, stop and restart the service component container named `Workflow Mailer Service` in Oracle Applications Manager for the changes to take effect. See: Service Components, page 7-7.

Notification Preferences:

Oracle Workflow lets users determine how they view notifications by setting a notification preference. As a workflow administrator, you can set the default notification preference for all users in your enterprise using the Workflow Configuration page. Users can override the default by modifying their individual notification preference setting the Preferences page in Oracle Applications. See: Setting Global User Preferences, page 2-9 and Set Preferences, *Oracle Applications User's Guide*.

Often, the functionality of a user's mail reader determines what the user's notification preference should be. Some mail readers can only display plain text, others can display HTML formatting, while still others can only display HTML formatting in an attachment. The following notification preferences are available:

- Plain text mail (`MAILTEXT`) - The notification message appears as plain text, with no attachments. See: Plain Text E-mail, page 2-60.

- HTML mail with attachments (`MAILHTML`) - The notification message appears as HTML-formatted text, with at least one standard attachment that is a link to the

notification in the Notification Details Web page. If the notification message has 'Content-Attached' message attributes, these attributes appear as additional custom attachments to the message. See: HTML-Formatted E-mail with Attachments, page 2-61.

- HTML mail (`MAILHTM2`) - The notification message appears as HTML-formatted text, but does not include any standard attachments. If the notification message has 'Content-Attached' message attributes, however, these attributes appear as custom attachments to the message. See: HTML-Formatted E-mail, page 2-63.

    > **Important:** If users wish to view notifications with HTML formatting, but their mail reader is not able to interpret HTML formatting in the mail message body, they can change their notification preference to 'Plain text mail with HTML attachments' (`MAILATTH`). The `MAILATTH` preference delivers an HTML-formatted version of the notification as an attachment to the plain text notification.

- Plain text mail with HTML attachments (`MAILATTH`) - The notification message appears as plain text, with at least two standard attachments. One attachment is an HTML-formatted version of the message, and the other is a link to the notification in the Notification Details Web page. If the notification message has 'Content-Attached' message attributes, these attributes appear as additional custom attachments to the message. See: Plain Text E-mail with an HTML Attachment, page 2-65.

- Plain text summary mail (`SUMMARY`) - The message is a plain text summary of all open notifications. To respond to the individual notifications in the summary, the user must access the notifications from the Worklist Web page.

- HTML summary mail (`SUMHTML`) - The message is an HTML-formatted summary of all open notifications, with a link to the Worklist Web page as well as links to each notification in the Notification Details Web page. To respond to the individual notifications in the summary, the user must access the notifications from the Web pages.

- Do not send me mail (`QUERY`) - The notification mailers do not send the user e-mail notifications. Instead the user must query and respond to notifications from the Worklist Web page.

- Disabled (`DISABLED`) - Oracle Workflow sets this preference for a user if a notification mailer attempted to send e-mail to the user, but the e-mail address on record for the user was invalid. Notification mailers do not send any further e-mail notifications to a user with the `DISABLED` notification preference. The user must correct his or her e-mail address in the Oracle Workflow directory service and then

reset the notification preference in order to receive e-mail notifications.

When a user's notification preference is set to `DISABLED`, Oracle Workflow also sends a notification about the failed e-mail and the reset notification preference to the `SYSADMIN` user. After the failed recipient user's e-mail address and notification preference are corrected, you can run the Resend Failed Workflow Notifications concurrent program to retry open notifications that could not be sent to the user previously. See: Outbound Notification Mailer Processing, page 2-51 and Handling Mailer Errors, page 2-73.

> **Important:** Users can always query and respond to their notifications from the Worklist Web page, even if they set their notification preference to receive e-mail or their notification preference is set to `DISABLED`.

You can run a diagnostic test to check that all users with a notification preference to receive e-mail have an e-mail address defined. See: Oracle Workflow Diagnostic Tests, page E-1.

See: Reviewing Notifications via Electronic Mail, *Oracle Workflow User's Guide*, Viewing Notifications from a Web Browser, *Oracle Workflow User's Guide*, and Reviewing a Summary of Your Notifications via Electronic Mail, *Oracle Workflow User's Guide*.

Plain Text E-mail

If the performer of a notification has a notification preference of plain text mail (`MAILTEXT`), when a notification mailer processes the notification, it generates a plain text e-mail message and sends it to the performer role. The notification mailer uses the Text Body defined for the message in the Oracle Workflow Builder message property page to generate the plain text e-mail. It token replaces all attribute values referenced in the message body with plain text values. For example:

- PL/SQL and PL/SQL CLOB document attributes are token replaced with a plain text version of a PL/SQL document.

- URL attributes are token replaced with the display name of the URL attribute, followed by a colon and the URL:

  <URL_Attribute_Display_Name>:*<URL>*

- A document attribute that represents an Oracle Application Framework region is token replaced with a plain text version of the region. Note that non-text content such as images, links, or special HTML formatting do not appear in the text version of the region. A notification with an embedded Oracle Application Framework region can contain multiple regions. However, it cannot contain any tokens for content other than regions.

  > **Important:** Message attributes that have Attach Content checked in

their Attributes property page, are attached as plain text to their parent notification. Note that this may render some attachments unreadable if the attachment includes special formatting or your plain text e-mail reader does not recognize attachments. To view these attachments, you should display your notifications in the Worklist Web page. See: Viewing Notifications from a Web Browser, *Oracle Workflow User's Guide*.

A recipient of a plain text e-mail notification responds by manually replying to the notification and entering response values following the instructions provided in the notification. See: To Respond to a Plain Text E-mail Notification Using Templated Response, *Oracle Workflow User's Guide* and To Respond to a Plain Text E-mail Notification Using Direct Response, *Oracle Workflow User's Guide*.

HTML-Formatted E-mail with Attachments

If the performer of a notification has a notification preference of HTML mail with attachments (`MAILHTML`), when a notification mailer processes the notification, it generates an HTML-formatted e-mail notification and sends it to the performer role. The recipient should use an e-mail reader that can interpret and display HTML content within a message body.

> **Note:** If your e-mail reader cannot interpret HTML formatting in a message body, you should set your notification preference to plain text mail with HTML Attachments (`MAILATTH`).

The notification mailer uses the HTML Body defined for the message in the Message Body property page to generate the HTML e-mail message. If no HTML Body is defined, it uses the Text Body to generate the HTML message. The notification mailer token replaces all message attributes referenced in the message body with HTML-formatted values. For example:

- PL/SQL and PL/SQL CLOB document attributes are token replaced with HTML text or plain text between `<pre>...</pre>` HTML tags.

- URL attributes that point to an image file are token replaced with `<IMG>...</IMG>` HTML tags to display the image inline within the message body. Notification mailers treat URL attributes as inline images if the URL points to a file with an extension of `gif`, `jpg`, `png`, `tif`, `bmp`, or `jpeg`, and the attribute value begins with the prefix `IMG:` or no prefix.

- URL attributes other than images, as well as image URL attributes that begin with the `LNK:` prefix, are token replaced with HTML links. When you select such a link, your e-mail reader takes you to the target URL page.

- A document attribute that represents an Oracle Application Framework region is

token replaced with the HTML version of the region. A notification with an embedded Oracle Application Framework region can contain multiple regions. However, it cannot contain any tokens for content other than regions.

> **Note:** For notifications that do not include embedded Oracle Application Framework regions, message attributes that have Attach Content checked in their Attributes property page, are appended as attachments to their parent message. For example:
>
> • If the message attribute is a URL attribute, an attachment called Notification References is appended to the message. This attachment includes a link to each URL attribute for the message that has Attach Content checked. You can navigate to a URL by choosing its link. Note that the Notification References attachment does not display images inline. If Attach Content is checked, a URL attribute always appears as a link in the Notification References attachment, even if the URL points to an image file. Notification mailers also do not have any special handling of video or audio URL content.
>
> • If the message attribute is a PL/SQL, PL/SQL CLOB, or PL/SQL BLOB document attribute, the fully generated PL/SQL document is fetched and attached to the message.
>
> However, if a notification includes an embedded Oracle Application Framework region, then Oracle Workflow includes the Related Applications region in the e-mail message with links to the attached URLs or PL/SQL documents, instead of appending them as separate attachments.

You can respond to your HTML-formatted notification by clicking on a link that represents the response in the HTML message body. The response link generates a plain text e-mail response that includes a response template modified with the predefined response value that you select. See: To Respond to an HTML E-mail Notification, *Oracle Workflow User's Guide*.

If your notification preference is `MAILHTML`, each HTML-formatted notification always includes at least one standard attachment. The attachment is called Notification Detail Link. When you select this attachment, your e-mail reader opens a browser window that displays your notification in the Notification Details Web page. You can alternatively respond directly to your notification from this Web page, bypassing the need to process your response through a notification mailer.

> **Note:** Depending on your configuration, if you are not already logged in, you may be prompted to log in when you select the Notification Detail Link before you can access the Notification Details page. See:

Responses through the Notification Detail Link Attachment, page 2-69.

> **Note:** You can use the Inline Attachment configuration parameter to set the Content-Disposition MIME header to either `inline` or `attachment` for all attachments to notification messages, including the Notification Detail Link, Notification References containing attached URLs, and attached PL/SQL, PL/SQL CLOB, or PL/SQL BLOB documents. Note, however, that some e-mail clients may not support the Content-Disposition header, or may support it in varying ways. Consequently, the Inline Attachment setting may not always have the desired effect, depending on the e-mail clients with which users read their e-mail messages.

> **Note:** The file name of the Notification Detail Link attachment is determined by the text value for the `WF_URL_NOTIFICATION` resource token, or by the token name if no text value is defined. Similarly, the file name of the Notification References attachment is determined by the text value for the `WF_URLLIST_ATTACHMENT` resource token, or by the token name if no text value is defined. The default file names are "`Notification Detail Link.html`" and "`Notification References.html`", respectively. If you want to specify different file names for these attachments, you must first create a `.msg` source file specifying the new file names as the text values for the `WF_URL_NOTIFICATION` and `WF_URLLIST_ATTACHMENT` resource tokens. Then use the Workflow Resource Generator program to upload the new seed data from the source file to the database table WF_RESOURCES. See: To Run the Workflow Resource Generator, *Oracle Workflow API Reference*.

HTML-Formatted E-mail

If the performer of a notification has a notification preference of HTML mail (`MAILHTM2`), without standard attachments, when a notification mailer processes the notification, it generates an HTML-formatted e-mail notification and sends it to the performer role. The recipient should use an e-mail reader that can interpret and display HTML content within a message body.

> **Note:** If your e-mail reader cannot interpret HTML formatting in a message body, you should set your notification preference to plain text mail with HTML Attachments (`MAILATTH`).

The notification mailer uses the HTML Body defined for the message in the Message Body property page to generate the HTML e-mail message. If no HTML Body is

defined, it uses the Text Body to generate the HTML message. The notification mailer token replaces all message attributes referenced in the message body with HTML-formatted values. For example:

- PL/SQL and PL/SQL CLOB document attributes are token replaced with HTML text or plain text between `<pre>...</pre>` HTML tags.

- URL attributes that point to an image file are token replaced with `<IMG>...</IMG>` HTML tags to display the image inline within the message body. Notification mailers treat URL attributes as inline images if the URL points to a file with an extension of `gif`, `jpg`, `png`, `tif`, `bmp`, or `jpeg`, and the attribute value begins with the prefix `IMG:` or no prefix.

- URL attributes other than images, as well as image URL attributes that begin with the `LNK:` prefix, are token replaced with HTML links. When you select such a link, your e-mail reader takes you to the target URL page.

- A document attribute that represents an Oracle Application Framework region is token replaced with the HTML version of the region. A notification with an embedded Oracle Application Framework region can contain multiple regions. However, it cannot contain any tokens for content other than regions.

  > **Note:** For notifications that do not include embedded Oracle Application Framework regions, message attributes that have Attach Content checked in their Attributes property page, are usually appended as attachments to their parent message. For example:
  >
  > - If the message attribute is a URL attribute, an attachment called Notification References is appended to the message. This attachment includes a link to each URL attribute for the message that has Attach Content checked. You can navigate to a URL by choosing its link. Note that the Notification References attachment does not display images inline. If Attach Content is checked, a URL attribute always appears as a link in the Notification References attachment, even if the URL points to an image file. Notification mailers also do not have any special handling of video or audio URL content.
  >
  > - If the message attribute is a PL/SQL, PL/SQL CLOB, or PL/SQL BLOB document attribute, the fully generated PL/SQL document is fetched and attached to the message.
  >
  > However, if a notification includes an embedded Oracle Application Framework region, then Oracle Workflow includes the Related Applications region in the e-mail message with links to the attached URLs or PL/SQL documents, instead of appending them as separate

attachments.

Note that although such message-specific attachments may be included, no standard attachments are included with the notification message if your notification preference is MAILHTM2.

You can respond to your HTML-formatted notification by clicking on a link that represents the response in the HTML message body. The response link generates a plain text e-mail response that includes a response template modified with the predefined response value that you select. See: To Respond to an HTML E-mail Notification, *Oracle Workflow User's Guide*.

> **Note:** You can use the Inline Attachment configuration parameter to set the Content-Disposition MIME header to either inline or attachment for all attachments to notification messages, including Notification References containing attached URLs and attached PL/SQL, PL/SQL CLOB, or PL/SQL BLOB documents. Note, however, that some e-mail clients may not support the Content-Disposition header, or may support it in varying ways. Consequently, the Inline Attachment setting may not always have the desired effect, depending on the e-mail clients with which users read their e-mail messages.

> **Note:** The file name of the Notification References attachment is determined by the text value for the WF_URLLIST_ATTACHMENT resource token, or by the token name if no text value is defined. The default file name is "Notification References.html". If you want to specify a different file name for this attachment, you must first create a .msg source file specifying the new file name as the text value for the WF_URLLIST_ATTACHMENT resource token. Then use the Workflow Resource Generator program to upload the new seed data from the source file to the database table WF_RESOURCES. See: To Run the Workflow Resource Generator, *Oracle Workflow API Reference*.

Plain Text E-mail with an HTML Attachment

If the performer of a notification has a notification preference of plain text mail with HTML attachments (MAILATTH), when a notification mailer processes the notification, it generates a plain text e-mail notification with HTML attachments and sends it to the performer role. The recipient should use an e-mail reader that supports HTML attachments.

The notification mailer uses the Text Body defined for the message in the Message Body property page to generate the plain text body of the e-mail. It also generates an HTML version of the notification message and sends it as an attachment called HTML Message Body to the plain text e-mail. The notification mailer generates the content of the HTML

attachment from the HTML Body defined for the message. If no HTML Body is defined, it uses the Text Body to generate the HTML mail. The notification mailer token replaces all message attributes referenced in the plain text message body with plain text values and token replaces all message attributes referenced in the attached HTML message with HTML-formatted values. See: Plain Text E-mail, page 2-60 and HTML-Formatted E-mail, page 2-63.

If your e-mail reader supports HTML formatting in the message body, you can optionally select the Inline Attachment configuration parameter to set the Content-Disposition MIME header to `inline` for attachments. Then the HTML attachment will also appear inline in the message body. Note, however, that some e-mail clients may not support the Content-Disposition header, or may support it in varying ways. Consequently, the Inline Attachment setting may not always have the desired effect, depending on the e-mail clients with which you read your e-mail messages.

> **Note:** For notifications that do not include Oracle Application Framework regions, message attributes that have Attach Content checked in their Attributes property page, are usually appended as attachments. For example:
>
> • If the message attribute is a URL attribute, an attachment called Notification References is appended to the message. This attachment includes a link to each URL attribute for the message that has Attach Content checked. You can navigate to a URL by choosing its link.
>
> • If the message attribute is a PL/SQL, PL/SQL CLOB, or PL/SQL BLOB document attribute, the fully generated PL/SQL document is fetched and attached to the message.
>
> However, if a notification includes an embedded Oracle Application Framework region, then Oracle Workflow includes the Related Applications region in the e-mail message with links to the attached URLs or PL/SQL documents, instead of appending them as separate attachments.

The notifications received by a user whose notification preference is `'Plain text with HTML attachments'` always contain at least two standard attachments. The first attachment is HTML Message Body and the other is Notification Detail Link. When you select Notification Detail Link, your e-mail reader opens a browser window that displays your notification in the Notification Details Web page. You can respond directly to your notification from this Web page, bypassing the need to process your response through a notification mailer. See: To Respond to a Plain Text E-mail Notification with an HTML Attachment, *Oracle Workflow User's Guide*.

> **Note:** Depending on your configuration, if you are not already logged in, you may be prompted to log in when you select the Notification Detail Link before you can access the Notification Details page. See: Responses through the Notification Detail Link Attachment, page 2-69.

Alternatively, a recipient of this type of notification can respond in one of two other ways:

- Manually reply to the notification and enter response values following the instructions provided in the notification. See: To Respond to a Plain Text E-mail Notification Using Templated Response, *Oracle Workflow User's Guide* and To Respond to a Plain Text E-mail Notification Using Direct Response, *Oracle Workflow User's Guide*.

- Select the HTML Message Body attachment to display the HTML-formatted version of the e-mail message, and click on the HTML link that represents the response. The response link generates a plain text e-mail response that includes a response template updated with the predefined response value you select.

  > **Note:** You can use the Inline Attachment configuration parameter to set the Content-Disposition MIME header to either `inline` or `attachment` for all attachments to notification messages, including the Notification Detail Link, HTML Message Body, Notification References containing attached URLs, and attached PL/SQL, PL/SQL CLOB, or PL/SQL BLOB documents. Note, however, that some e-mail clients may not support the Content-Disposition header, or may support it in varying ways. Consequently, the Inline Attachment setting may not always have the desired effect, depending on the e-mail clients with which users read their e-mail messages.

  > **Note:** The file name of the HTML Message Body attachment is determined by the text value for the `WF_HTML_MESSAGE` resource token, or by the token name if no text value is defined. Similarly, the file name of the Notification Detail Link attachment is determined by the text value for the `WF_URL_NOTIFICATION` resource token, or by the token name if no text value is defined; and the file name of the Notification References attachment is determined by the text value for the `WF_URLLIST_ATTACHMENT` resource token, or by the token name if no text value is defined. The default file names are "`HTML Message Body.html`", "`Notification Detail Link.html`", and "`Notification References.html`", respectively. If you want to specify different file names for these attachments, you must first create a `.msg` source file specifying the new file names as the text values for the `WF_HTML_MESSAGE`, `WF_URL_NOTIFICATION`, and

`WF_URLLIST_ATTACHMENT` resource tokens. Then use the Workflow Resource Generator program to upload the new seed data from the source file to the database table WF_RESOURCES. See: To Run the Workflow Resource Generator, *Oracle Workflow API Reference*.

E-mail Notification Security:

Each individual e-mail notification message sent by a notification mailer includes a line containing a notification ID (NID), access key, and node identifier, which are used to authenticate responses to the notification.

- The NID identifies the notification in the database.

- The notification access key is a distinct random key generated by the Notification System for each NID. The access key must be included in a response to the notification in order for a notification mailer to accept the response, thereby serving as a password that allows only users who actually received the notification containing the key to respond to that notification.

- The node identifier specifies the notification mailer node to which the message belongs.

The format of the NID line is as follows:

`NID[`*`NID/access_key@node_identifier`*`]`

Responses by E-mail

When a user responds to a notification by e-mail, the response message must include the NID line from the original notification message. A notification mailer accepts the response only if the correct NID and access key combination is included in the response. Users can ensure that the response message contains the NID and access key either by including the entire original message when replying or by using a response template that includes the NID line.

> **Note:** Some mail clients, notably early releases of Microsoft Outlook Express, may not copy the NID line properly in a reply message. When responding to a notification, users should verify that the NID line is included in full and contains the prefix `NID` and all the details between the square brackets.

A user who receives an e-mail notification message may forward the message to another user through the e-mail application. When you configure a notification mailer, you can choose whether to allow a user to respond by e-mail to an e-mail notification that has been forwarded from another role.

- If you deselect the Allow Forwarded Response configuration parameter, the notification mailer will check if the "From:" e-mail address of the notification

response exactly matches the e-mail address of the recorded recipient role, or the e-mail address of a user in that role. If the two e-mail addresses match exactly, meaning the notification was not forwarded or was forwarded according to a valid vacation routing rule, the notification mailer treats the response as a valid response. If the two e-mail addresses do not match exactly, meaning the notification was simply forwarded using the e-mail Forward command, the notification mailer does not process the response and treats it as unsolicited mail.

- If you select the Allow Forwarded Response configuration parameter, the notification mailer that receives the notification never checks the "From:" e-mail address of the notification response and always allows the response to be processed. In this case, users can delegate notifications to other users simply by forwarding the notification message through the e-mail application, and the new recipient of a forwarded notification automatically receives the authority to respond to it.

> **Important:** Note that there are limitations when you deselect the Allow Forwarded Response parameter. For example, suppose a notification is sent to a distribution list mail alias that does not have a user/role relationship in the Oracle Workflow directory service. If any user from the distribution list responds to the notification, the notification mailer will always treat their notification response as unsolicited mail, because the "From:" e-mail address, which is an individual user's e-mail address, will never match the distribution list mail alias.

Responses through the Notification Detail Link Attachment

HTML-formatted e-mail notifications with attachments and plain text e-mail notifications with HTML attachments include an attachment called Notification Detail Link. When this link is clicked, it displays the notification in the Notification Details Web page. A user who receives a notification with a Notification Detail Link attachment can use this Web page to respond directly to the notification, instead of sending an e-mail response message to be processed by a notification mailer.

You can choose whether to require users to log in before they can access the Notification Details Web page for a notification through the Notification Detail Link.

- By default, users must always log in before they can access the Notification Details page in Oracle Applications from the Notification Detail Link.

- You can optionally enable guest access to the Notification Details page. Guest access lets users access this page from e-mail notifications without logging in to Oracle Applications with an individual user name and password. This feature is not recommended due to security considerations. However, if you choose to allow guest access, you can perform the following steps to enable it:

  - Set the WF: GUEST Access to Notification profile option to `Enabled` at the site

level. See: Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide*.

- Create a grant assigning the "`Workflow Guest User permission set`" to the `GUEST` user. When defining the set for the grant, select the set type `Navigation Menu` and select the menu named "`Workflow Guest User permission set`" (internal code: `WF_GUEST_GRANTS`). After creating the grant, you must stop and restart Oracle HTTP Server for the change to take effect. See: Create Grant, *Oracle Applications System Administrator's Guide*.

- In Oracle Applications Manager, stop and restart the service component container named `Workflow Mailer Service`. See: Service Components, page 7-7.

With guest access, if a user navigates to the Notification Details page and is not already logged in to Oracle Applications, the user is logged in automatically as the `GUEST` user. The user can then respond to the notification, and can also reassign the notification or request more information if those actions are available for that notification. However, the user cannot access any other notification in the Notification Details page, nor any other Oracle Workflow Web pages.

In cases where Oracle Workflow records the identity of the logged in user who acted on a notification, the action history will show those actions as being performed by the `GUEST` user.

When a user views a notification through guest access, Oracle Workflow displays the notification according to the language and territory preferences of the recipient role for the notification and the date and number preferences of the `GUEST` user. To view notifications with their own preferences, users can log in with their own user names and passwords before accessing the notifications.

Oracle Workflow does not support guest access for notifications that require electronic signatures. If you want users to sign their notification responses with password-based signatures or certificate-based digital signatures, those users must log in with their own user names and passwords to enter their signatures.

> **Note:** If you enabled guest access but no longer want to allow it, you can disable it by setting the WF: GUEST Access to Notification profile option to `Disabled` and setting an end date for the grant you created. Then stop and restart Oracle HTTP Server and, in Oracle Applications Manager, stop and restart the service component container named `Workflow Mailer Service`. Users will then always be required to log in before they can access the Notification Details page from the Notification Detail Link.

E-mail Notification Summaries

Instead of individual e-mail notifications, users can also receive e-mail summaries

listing all their open notifications. Users can indicate that they want to receive e-mail summaries by choosing a notification preference of SUMMARY or SUMHTML.

- SUMMARY - Users receive plain text e-mail summary messages, which do not enable any direct response through e-mail to the notifications they list. Instead, to respond to the individual notifications in a summary, users must log in to Oracle Workflow and access the notifications through the Worklist Web page.

- SUMHTML - Users receive HTML-formatted e-mail summary messages. An HTML-formatted summary does not enable responses through e-mail. However, it includes a link to the Worklist Web page as well as links to each notification in the Notification Details Web page, where users can respond to the individual notifications. Users must log in to Oracle Applications before they can access the Worklist and Notification Details pages, unless you enable guest access to the Notification Details page. See: Responses Through the Notification Detail Link Attachment, page 2-69.

To send e-mail summaries, schedule a Launch Summary Notifications event for a notification mailer. For the seeded Workflow Notification Mailer, the Launch Summary Notifications event is scheduled to send e-mail summary notifications once a day by default.

Mail Server Connections Through SSL

You can configure a notification mailer to connect to the SMTP server and IMAP server through Secure Sockets Layer (SSL) to encrypt the data exchanged. See: Connecting to Mail Servers Through SSL, page 2-56.

Confirming Responses with Electronic Signatures

You can require that the response to a notification be signed with either a password-based signature or a certificate-based digital signature. In this case, users cannot respond to that notification through e-mail. Instead, they must respond to the notification from the Notification Details Web page and enter the appropriate type of signature.

Users can access the Notification Details page by these methods.

- Log into Oracle Applications separately.

- Use the Notification Detail Link, for users with the MAILHTML and MAILATTH notification preferences.

- Use a "Click here to respond" link included in the HTML-formatted version of the message, for users with the MAILHTML, MAILHTM2, and MAILATTHnotification preferences.

Users must log in to Oracle Applications before they can access the Notification Details page from the Notification Detail Link or "Click here to respond" link.

> **Note:** If you enable guest access to the Notification Details page, the "Click here to respond" link does not appear in the HTML-formatted message. See: Responses Through the Notification Detail Link Attachment, page 2-69.

Use the special message attribute `#WF_SIG_POLICY` to specify the signature policy for a notification. See: #WF_SIG_POLICY Attribute, *Oracle Workflow Developer's Guide*.

You can use the Signature Evidence Store to review details about the electronic signatures requested or submitted for notifications that require signatures. You can also view some signature details in the Notification Response Details page in the Status Monitor. See: Reviewing Electronic Signature Details, page 6-5 and Viewing Responses, page 5-12.

Excluding Notification Content From E-mail

If a particular notification contains sensitive information that you do not want to send in e-mail, you can choose to exclude the content of the notification from the e-mail version of the notification. In this case, users receive an e-mail message that only informs them that they must access the notification through the Notification Details Web page instead to view the content and respond. To access the Notification Details page, users can either log into Oracle Applications separately, or, if their notification preference includes HTML attachments, use the Notification Detail Link.

Use the special message attribute `#WF_SECURITY_POLICY` to specify the content security policy for a notification. See: #WF_SECURITY_POLICY Attribute, *Oracle Workflow Developer's Guide*.

CC and BCC E-mail Recipients

Oracle Workflow can send copies of a notification e-mail message to secondary (CC) and additional non-displayed (BCC) recipients in addition to the primary recipient. However, the CC and BCC recipients are not added to the recipient role of the notification in the Notification System.

Adding CC and BCC recipients is recommended only for FYI notifications. If you do add CC or BCC recipients to a response-required notification, consider whether the additional recipients should be able respond to the notification, and set up your security options accordingly.

- If the notification includes the Notification Detail Link attachment and you enable guest access to the Notification Details page, then CC and BCC recipients can use the Notification Detail Link attachment to access the Notification Details page, and view and respond to the notification there. If you do not want CC and BCC recipients to access the Notification Details page, then you should disable guest access, or require your users to set notification preferences that do not include the Notification Detail Link attachment, such as `MAILTEXT` or `MAILHTM2`. See: Responses Through the Notification Detail Link Attachment, page 2-69.

- If the notification allows responses by e-mail and you select the Allow Forwarded Response mailer configuration parameter, then CC and BCC recipients can respond to the notification by e-mail. If you do not want CC and BCC recipients to respond by e-mail, deselect the Allow Forwarded Response mailer configuration parameter, or choose notification options that require users to respond through the Notification Details page. For example, users must use the Notification Details page to respond to notifications that require electronic signatures and notifications marked as including secure content not sent by e-mail. See: Responses By E-mail, page 2-68, Confirming Responses with Electronic Signatures, page 2-71, and Excluding Notification Content From E-mail, page 2-72.

- If the notification requires an electronic signature, then only a primary recipient who provides a valid signature in the Notification Details page can respond. In this case CC and BCC recipients cannot respond to the notification. See: Confirming Responses with Electronic Signatures, page 2-71.

See: Additional E-mail Recipient Attributes, *Oracle Workflow Developer's Guide*.

Sending Outbound E-mail Notifications Only

If you do not want to allow responses by e-mail, you can choose to send only outbound e-mail notifications. To configure your notification mailers for outbound-only processing, set the inbound thread count to `0` (zero) in the configuration wizard for each notification mailer, or deselect the Inbound Processing parameter in the Basic Configuration page for each notification mailer.

When you set up an outbound-only mailer, you should configure the mailer to use message templates for response-required notifications that do not request a response by e-mail, but instead direct recipients to respond from the Notification Details Web page. For example, you can configure the mailer to send response-required notifications using the Workflow View From UI message template, which is an alternative template provided by Oracle Workflow in the System: Mailer item type, or create your own custom message templates. The outbound-only mailer can still use the standard message templates to send outbound summary notifications or For Your Information (FYI) notifications that do not require a response.

Disabling E-mail Notifications

Ultimately, the security of e-mail notifications depends on the security of your e-mail application. If you do not want to send any workflow information by e-mail, you can choose not to run any notification mailers at all. In this case users must always log on to Oracle Workflow and access the Worklist Web page to view and respond to their notifications.

See: Overview of Notification Handling, *Oracle Workflow User's Guide*.

Handling Mailer Errors:

To check the status of a particular notification or help investigate errors, you can run a script named `wfmlrdbg.sql` that displays debugging information. You can also obtain

this information by running a diagnostic test through Oracle Diagnostics. See: wfmlrdbg.sql, page 9-10 and Oracle Workflow Diagnostic Tests, page E-1.

Additionally, you can run diagnostic tests through Oracle Diagnostics to check that at least one notification mailer is configured, to validate the notification mailer configuration parameters, and to check that all users with a notification preference to receive e-mail have an e-mail address defined. See: Oracle Workflow Diagnostic Tests, page E-1.

> **Note:** You must particularly check the notification preference and e-mail address for the SYSADMIN user. This user is the default recipient for several types of notifications such as error notifications. By default, the SYSADMIN user has a notification preference to receive e-mail notifications. To enable Oracle Workflow to send e-mail to this user, navigate to the Users window in Oracle Applications and assign SYSADMIN an e-mail address that is fully qualified with a valid domain. However, if you want to access notifications only through the Oracle Workflow Worklist Web page, then you should change the notification preference for SYSADMIN to "Do not send me mail" in the Preferences page. In this case you do not need to define an e-mail address. See: System Administration Setup Tasks, *Oracle Applications System Administrator's Guide*.

You can also run command-line diagnostic tests for notification mailers through oracle.apps.fnd.wf.mailer.Mailer to check mail server and Web server connectivity and check the size or number of messages in an IMAP folder. See: Running Command-Line Notification Mailer Diagnostics, page 2-78.

The Generic Service Component Framework lets you control how errors are handled through the component-level Max Error Count parameter and the container-level SVC_COMP_MAX_ERROR_COUNT parameter.

- The Max Error Count (PROCESSOR_MAX_ERROR_COUNT) parameter for a service component determines how many consecutive errors the component can encounter before its container stops it and changes its status to Stopped with Error. If the component's startup mode is Automatic or On-Demand, the container will then restart the component. The default value for this parameter is 10.

- The SVC_COMP_MAX_ERROR_COUNT parameter for a container determines how many times a component within that container can be stopped with error. If this maximum count is reached, the container changes the status of the component to System Deactivated and will no longer automatically restart it. The default value for this parameter is 5.

The total number of errors before a mailer is permanently stopped consists of the Max Error Count value multiplied by the SVC_COMP_MAX_ERROR_COUNT value. For example, using the default values, a mailer can encounter 10 * 5 = 50 errors before it becomes System Deactivated.

If a mailer encounters multiple consecutive errors, it may be advantageous to let the container restart the mailer. Restarting causes the mailer to establish new connections and instantiate new objects, which may resolve the errors. Consequently, if you want to allow more errors before you must manually intervene to restart the mailer, it is usually better to increase the `SVC_COMP_MAX_ERROR_COUNT` value than the Max Error Count value.

For more information about configuring service component and container parameters, see: Service Components, page 7-7.

> **Note:** If the status of a notification mailer service component changes to `Stopped with Error` or `System Deactivated`, Oracle Workflow posts a system alert to the System Alerts and Metrics page in Oracle Applications Manager. See: System Alerts, Metrics, and Logs, *Oracle Applications System Administrator's Guide - Maintenance*.

In case of a large number of errored notifications, Oracle Workflow provides special scripts for mass mailer reprocessing. Do not run these scripts unless you are directed to do so by Oracle Support.

The following scripts are located in the `$FND_TOP/patch/115/sql` directory.

- `wfntfqup.sql` - This script rebuilds the WF_NOTIFICATION_OUT queue. It drops and recreates the WF_NOTIFICATION_OUT queue, removes pending notification messages from the WF_DEFERRED queue, and repopulates the WF_NOTIFICATION_OUT queue from the Oracle Workflow Notification System tables.

  You must stop the service component containers for notification mailers and agent listeners before you run this script, and restart the containers after the script completes. The container for notification mailers is named `Workflow Mailer Service`. The container for agent listeners is named `Workflow Agent Listener Service`.

  Use the script as follows:

  `sqlplus <user/pwd> @wfntfqup <APPSuser> <APPSpwd> <FND_schema>`

  Replace `<APPSuser>` and `<APPSpwd>` with the user name and password for the APPS user. The user name is usually `apps`. Replace `<FND_schema>` with the ORACLE username that connects to Oracle Application Object Library data in Oracle Applications, usually `applsys`.

- `wfnequ.sql` - This script moves all the `oracle.apps.wf.notification.send` event messages that are on the WF_ERROR queue back to the WF_DEFERRED queue for reprocessing.

  Use the script as follows:

  `sqlplus <user/pwd> @wfnequ <APPSuser> <APPSpwd> <FND_schema>`

  Replace `<APPSuser>` and `<APPSpwd>` with the user name and password for the

APPS user. The user name is usually `apps`. Replace *`<FND_schema>`* with the ORACLE username that connects to Oracle Application Object Library data in Oracle Applications, usually `applsys`.

The following scripts are located in the `$FND_TOP/patch/115/sql` directory.

- `wfntffix.sql` - This script updates the `MAIL_STATUS` to `NULL` for notifications that are ineligible to be sent by e-mail. Notifications become ineligible if the recipient role does not have a valid e-mail address defined, or if the recipient role has a notification preference set not to receive e-mail. Running this script removes ineligible notifications from the notification mailer throughput displayed in Oracle Workflow Manager, so you can review the outstanding eligible notifications only.

  Use the script as follows:

  `sqlplus <user/pwd> @wfntffix`

- `wfntfsnd.sql` - This script updates the `MAIL_STATUS` from null to `MAIL` for notifications of the specified item type that were sent on or after the specified date. The recipient role of each notification must have a valid e-mail address defined. Then the script enqueues those notifications on the WF_DEFERRED queue for reprocessing by the mailer.

  For example, if users change their notification preference from not receiving e-mail to receiving e-mail, run this script to send any existing open notifications to those users by e-mail.

  Use the script as follows:

  `sqlplus <user/pwd> @wfntfsnd <item_type> <begin_date_after>`

  Replace *`<item_type>`* with the internal name of the item type for the notifications to update. Replace *`<begin_date_after>`* with the earliest sent date for the notifications to update.

Oracle Workflow also provides concurrent programs that help enable mass reprocessing of notifications. See: Running Reports and Programs, *Oracle Applications User's Guide*.

- Resend Failed Workflow Notifications (`FNDWF_NTF_RESEND`) - This program lets you resend e-mail notifications with a mail status of `FAILED` after correcting the issues that prevented the e-mails from being sent. For example, Oracle Workflow sets a notification's mail status to `FAILED` if the e-mail address for the recipient is invalid and the e-mail cannot be delivered, if the e-mail address for the recipient is on a notification mailer's list of e-mail addresses previously found to be invalid, or if the recipient's notification preference is set to `DISABLED`. To correct these issues, correct the user's e-mail address in the Oracle Workflow directory service and then reset the user's notification preference.

  Use Standard Request Submission to run this program. Specify the recipient role whose notifications you want to resend, or leave this parameter blank to resend notifications to all recipient roles. The program then re-enqueues the notifications

on the notification mailer's outbound queue to be sent by e-mail.

> **Note:** When a user's notification preference is set to `DISABLED`, Oracle Workflow sends a notification about the failed e-mail and the reset notification preference to the `SYSADMIN` user. See: Outbound Notification Mailer Processing, page 2-51.

> **Note:** Run the Resend Failed Workflow Notifications program only after taking action to correct the issues that prevented the e-mails from being sent. Do not schedule the program to run repeatedly without investigating the e-mail issues; otherwise the notifications may fail again due to the same issues.

* Move Messages from Exception to Normal Queue of Workflow Agent (`FNDWF_MOVE_MSGS_EXCEP2NORMAL`) - This program lets you transfer messages from an Oracle Streams Advanced Queuing (AQ) exception queue back to the associated normal queue, for queues that are assigned to an inbound agent. AQ transfers messages to an exception queue if an agent listener repeatedly encounters exceptions when attempting to dequeue messages, or if messages expire before being dequeued. To aid in mass mailer reprocessing, you can run the concurrent program for the WF_DEFERRED agent or WF_NOTIFICATION_IN agent if necessary. You can then run the appropriate agent listener to reattempt processing for the messages that have been transferred back to the agent's normal queue. See: Exception Handling for Inbound Queues, page 2-153.

Resetting Notification Mailers After Cloning:

If you clone an Oracle Applications instance, the cloned instance by default includes notification mailer data from the source instance, including notification mailer configuration parameters and information for notifications that are eligible to be sent by e-mail. Consequently, the cloned instance may send e-mail notifications as if they were from the source instance. If you do not want the cloned instance to continue sending e-mail notifications, you can run a script named `wfmlpcln.sql` to reset the notification mailer configurations and notification mail statuses.

The`wfmlpcln.sql` script performs the following actions for all notification mailers in the Oracle Applications instance where you run the script:

* Resets required configuration parameters to their initial values as follows:

    * Outbound E-mail Account (SMTP): Server Name - Null

    * Inbound E-mail Account (IMAP): Server Name - Null

    * Inbound E-mail Account (IMAP): Username - Null

- Inbound E-mail Account (IMAP): Password - Null

- From - `Workflow Mailer`

- Reply-to Address - Null

- HTML Agent - Null

- Sets the override e-mail address to `NONE`.

- Sets the notification mailer service component status to `NOT_CONFIGURED`.

- Sets the status of the seeded subscription to the
  `oracle.apps.wf.notification.send.group` event group to `DISABLED`.

- Sets the mail status to null for all notifications that formerly had a mail status of
  `MAIL`, `ERROR`, or `FAILED`, indicating that they were eligible to be sent by e-mail.

The `wfmlpcln.sql` script is located in the `$FND_TOP/sql` directory. Use the script as
follows:

```
sqlplus <user/pwd> @wfmlpcln
```

If you later want to send e-mail notifications from the cloned instance, you can use
Oracle Workflow Manager to reconfigure the notification mailers as appropriate for that
instance. When you do so, Oracle Workflow automatically re-enables the seeded
subscription to the `oracle.apps.wf.notification.send.group` event group.

See: Notification Events, *Oracle Workflow Developer's Guide* and Cloning, *Oracle
Applications Concepts*.

Running Command-Line Notification Mailer Diagnostics:

Oracle Workflow provides command-line diagnostic tests for notification mailers
through `oracle.apps.fnd.wf.mailer.Mailer`. With these tests, you can:

- Test connectivity to the IMAP mail server.

- Test connectivity to the SMTP mail server.

- Test connectivity to the Web tier, which is required for the notification mailer to
  generate notifications that include Oracle Application Framework content.

- Check the number of messages in an IMAP folder or the total size of the messages
  in the folder in bytes, which indicates how much space you can regain by purging
  messages from the folder.

These tests use the following variables:

- `AFJVAPRG` - The location of the JDK or JRE executable for the concurrent processing

tier.

- `AF_CLASSPATH` - The classpath for the concurrent processing tier.

Use the following command to test connectivity to the IMAP mail server.

```
$AFJVAPRG -classpath $AF_CLASSPATH -Dprotocol=imap \
  ( -Ddbcfile=<dbcfileLocation> | -Ddbuser -Ddbpassword -Ddburl )\
  -Dserver=<server_name> [-Dport=<port>] \
  -Daccount=<account_name> -Dpassword=<password> \
  [ -Dfolder=<folder_name> ] \
  [ -Dssl=<Y|N> ] \
  [ -Dtruststore=<truststore> ]\
  [ -Dconnect_timeout=<seconds> ] \
  [ -Ddebug=<Y|N> ]\
  [ -Dlogfile=<log_file_name> ]\
  oracle.apps.fnd.wf.mailer.Mailer
```

Specify the following parameters.

- `-Dprotocol` - Specify `imap` to connect to the IMAP mail server.

- `-Ddbcfile` or `-Ddbuser`, `-Ddbpassword`, and `-Ddburl` - Either specify the location of the DBC file for the database, or specify the user name, password, and URL to use to connect to the database. The URL should include the host name, port number, and database system identifier (SID) in the following format:

  `jdbc:oracle:thin:<host_name>:<port_number>:<database_SID>`

- `-Dserver` - Specify the name of the IMAP mail server.

- `-Dport` - Optionally specify the port number to use on that server. If you do not specify a port number, the default is port 143.

- `-Daccount` and `-Dpassword` - Specify the user name and password of the mail account that the notification mailer uses to receive e-mail messages.

- `-Dfolder` - Optionally specify the name of the folder that the notification mailer uses as its inbox folder.

- `-Dssl` - Optionally specify `Y` to use SSL for connections to the IMAP server or `N` to use non-SSL connections. The default is `N`.

  > **Note:** If you specify the IMAP SSL port, 993, in the `-Dport` parameter, then the test automatically uses SSL, regardless of the value you specify for the `-Dssl` parameter.

- `-Dtruststore` - If you are using SSL, optionally specify the location of the trust store where the IMAP server's certificate is loaded.

- `-Dconnect_timeout` - Optionally specify the number of seconds to wait when

attempting to connect before timing out. The default is 5 seconds.

- -Ddebug - Optionally specify Y to run in debug mode and report more extensive debugging information in the program output or N to run in normal mode. The default is N.

- -Dlogfile - Optionally specify the name of the log file for the program output. The default is test.log.

Use the following command to test connectivity to the SMTP mail server.

```
$AFJVAPRG -classpath $AF_CLASSPATH -Dprotocol=smtp \
  ( -Ddbcfile=<dbcfileLocation> | -Ddbuser -Ddbpassword -Ddburl )\
  -Dserver=<server_name> [-Dport=<port>] \
  -Daccount=<account_name> \
  [ -Dssl=<Y|N> ] \
  [ -Dtruststore=<truststore> ]\
  [ -Dconnect_timeout=<seconds> ] \
  [ -Ddebug=<Y|N> ]\
  [ -Dlogfile=<log_file_name> ]\
  oracle.apps.fnd.wf.mailer.Mailer
```

Specify the following parameters.

- -Dprotocol - Specify smtp to connect to the SMTP mail server.

- -Ddbcfile or -Ddbuser, -Ddbpassword, and -Ddburl - Either specify the location of the DBC file for the database, or specify the user name, password, and URL to use to connect to the database. The URL should include the host name, port number, and database system identifier (SID) in the following format:

  jdbc:oracle:thin:<host_name>:<port_number>:<database_SID>

- -Dserver - Specify the name of the SMTP mail server.

- -Dport - Optionally specify the port number to use on that server. If you do not specify a port number, the default is port 25.

- -Daccount - Specify the user name of the mail account that the notification mailer uses.

- -Dssl - Optionally specify Y to use SSL for connections to the SMTP server or N to use non-SSL connections. The default is N.

    > **Note:** If you specify the SMTP SSL port, 465, in the -Dport parameter, then the test automatically uses SSL, regardless of the value you specify for the -Dssl parameter.

- -Dtruststore - If you are using SSL, optionally specify the location of the trust store where the SMTP server's certificate is loaded.

- `-Dconnect_timeout` - Optionally specify the number of seconds to wait when attempting to connect before timing out. The default is 5 seconds.

- `-Ddebug` - Optionally specify `Y` to run in debug mode and report more extensive debugging information in the program output or `N` to run in normal mode. The default is `N`.

- `-Dlogfile` - Optionally specify the name of the log file for the program output. The default is `test.log`.

Use the following command to test connectivity to the Web tier.

```
$AFJVAPRG -classpath $AF_CLASSPATH \
  ( -Ddbcfile=<dbcfileLocation> | -Ddbuser -Ddbpassword -Ddburl )\
  ( -Dnid=<notification_ID> | -Durl=<simpleURL> ) \
  [ -Dappuser=<ApplicationsUserID> ]\
  [ -Dappresp=<ApplicationsResponsibilityID>  ]\
  [ -Dappid=<ApplicationsAppID> ]\
  [ -Dhtp=<http | https> ]\
  [ -Durltimeout=<seconds> ]\

  [ -Dlogfile=<log_file_name> ]\
  oracle.apps.fnd.wf.mailer.Mailer
```

Specify the following parameters.

- `-Ddbcfile` or `-Ddbuser`, `-Ddbpassword`, and `-Ddburl` - Either specify the location of the DBC file for the database, or specify the user name, password, and URL to use to connect to the database. The URL should include the host name, port number, and database system identifier (SID) in the following format:

  `jdbc:oracle:thin:<host_name>:<port_number>:<database_SID>`

- `-Dnid` or `-Durl` - Either specify the notification ID for a notification that includes Oracle Application Framework content, or specify the URL for an Oracle Application Framework application page to which you want to connect.

- `-Dappuser` - Optionally specify the numerical user ID for the user through which the notification mailer accesses Oracle Application Framework content. The default is `0`, which is the user ID for the `SYSADMIN` user.

- `-Dappresp` - Optionally specify the numerical responsibility ID for the responsibility through which a notification mailer accesses Oracle Application Framework content. The default is `20420`, which is the responsibility ID for the System Administrator responsibility.

- `-Dappid` - Optionally specify the numerical application ID for the application through which a notification mailer accesses Oracle Application Framework content. The default is `1`, which is the application ID for the System Administration application.

- `-Dhtp` - Optionally specify `http` to connect using the HTTP protocol or `https` to

connect using the HTTPS (HTTP over Secure Sockets Layer) protocol.

> **Note:** If you specify `https`, ensure that the `LD_LIBRARY_PATH` environment variable is set to the value `$AF_LD_LIBRARY_PATH`, which is the library path for the concurrent processing tier. Otherwise, you may encounter the error `java.lang.UnsatisfiedLinkError: initSSLContextNative`.

- `-Durltimeout` - Optionally specify the number of seconds wait to access the URL for Oracle Application Framework content before timing out. The default is 30 seconds.

- `-Dlogfile` - Optionally specify the name of the log file for the program output. The default is `test.log`.

Use the following command to check the number of messages in an IMAP folder or the total size of the messages in the folder in bytes.

```
$AFJVAPRG -classpath $AF_CLASSPATH -Dprotocol=imap \
  ( -Ddbcfile=<dbcfileLocation> | -Ddbuser -Ddbpassword -Ddburl )\
  -Dserver=<server_name> [-Dport=<port>] \
  -Daccount=<account_name> -Dpassword=<password> \
  [ -Dssl=<Y|N> ] \
  [ -Dtruststore=<truststore> ]\
  [ -Dconnect_timeout=<seconds> ] \
  [ -Ddebug=<Y|N> ]\
  [ -Dlogfile=<log_file_name> ]\
  -Dfolder=<folder_name> \
  -Dfolder_usage=<count | size> \
  [ -Dcheck_age=<age_in_days> \
  [ -Dinclude_flag=<2 | 4 | 8 | 14> \
  oracle.apps.fnd.wf.mailer.Mailer
```

Specify the following parameters.

- `-Dprotocol` - Specify `imap` to connect to the IMAP mail server.

- `-Ddbcfile` or `-Ddbuser`, `-Ddbpassword`, and `-Ddburl` - Either specify the location of the DBC file for the database, or specify the user name, password, and URL to use to connect to the database. The URL should include the host name, port number, and database system identifier (SID) in the following format:

  `jdbc:oracle:thin:<host_name>:<port_number>:<database_SID>`

- `-Dserver` - Specify the name of the IMAP mail server.

- `-Dport` - Optionally specify the port number to use on that server. If you do not specify a port number, the default is port 143.

- `-Daccount` and `-Dpassword` - Specify the user name and password of the mail account that the notification mailer uses to receive e-mail messages.

- -Dssl - Optionally specify `Y` to use SSL for connections to the IMAP server or `N` to use non-SSL connections. The default is `N`.

  > **Note:** If you specify the IMAP SSL port, 993, in the -Dport parameter, then the test automatically uses SSL, regardless of the value you specify for the -Dssl parameter.

- -Dtruststore - If you are using SSL, optionally specify the location of the trust store where the IMAP server's certificate is loaded.

- -Dconnect_timeout - Optionally specify the number of seconds to wait when attempting to connect before timing out. The default is 5 seconds.

- -Ddebug - Optionally specify `Y` to run in debug mode and report more extensive debugging information in the program output or `N` to run in normal mode. The default is `N`.

- -Dlogfile - Optionally specify the name of the log file for the program output. The default is `test.log`.

- -Dfolder - Specify the name of the folder you want to check.

- -Dfolder_usage - Specify `count` to check the number of messages in the folder or `size` to check the total size of the messages in the folder in bytes.

- -Dcheck_age - If you are checking the folder size, optionally specify the minimum age in days of the messages to include in the check. With this option, you can check how many bytes are occupied by messages of a certain age or older.

- -Dinclude_flag - If you are checking the folder size, optionally specify one of the following values to indicate the status of the messages to include in the check.
  - `2` - Include only unseen messages.
  - `4` - Include only seen messages.
  - `8` - Include only deleted messages.
  - `14` - Include all messages.

  With this option, you can check how many bytes are occupied by messages in a particular status. The default is to include all messages.

  > **Note:** The -Dcheck_age and -Dinclude_flag parameters do not apply if you are checking the number of messages in the folder.

Step 7: Modifying Your Message Templates

Notification mailers use message templates defined in Oracle Workflow Builder to generate e-mail notifications. Oracle Workflow provides a set of standard templates which are used by default, as well as some alternative templates for certain types of messages. These message templates are defined in the System: Mailer item type.

Although message templates are defined as messages in Oracle Workflow Builder, they are not true messages. Rather, they serve as outlines for e-mail messages sent by notification mailers. Message templates determine the basic format of an e-mail notification, including what header information to include, and whether and where to include details such as the message due date and priority. Message templates for notifications that require a response should also describe the syntax the reply should follow and list the information needed to confirm the notification.

It is not recommended to modify the standard templates. However, you can optionally customize the message templates used to send your e-mail notifications by either using the alternative templates provided in the System: Mailer item type by Oracle Workflow, or creating your own custom message templates in the System: Mailer item type using the Workflow Builder. You can implement alternative standard or custom templates in the following ways:

- Assign the templates you want to a particular notification mailer service component in the mailer configuration parameters. The templates assigned to a mailer override the default System: Mailer templates. See: Notification Mailer Configuration Wizard, page 7-30.

- Assign the templates you want to a particular notification in a workflow process by defining special message attributes. In this case the templates assigned to the notification override both the templates assigned to a mailer and the default System: Mailer templates. See: Notification Mailer Message Template Attributes, *Oracle Workflow Developer's Guide*.

The templates in the System: Mailer item type have message attributes that represent every part of the notification message. Within the body of a template, the message attributes are token substituted to insert the specific information for a particular instance of a notification into the message outline.

> **Note:** Do not modify, add new attributes to, or delete existing attributes from the standard message templates in the System: Mailer item type.

If you create new custom templates, you must name the message attributes for these templates with the same names as the message attributes for the standard templates. A notification mailer can only token substitute the attributes in the message body if you use the standard attribute names.

You can optionally omit some of the standard tokens from your custom templates, if

you do not want to send the information they represent. However, you should not omit the tokens that represent the key information to be conveyed in the notification. For example, if you define a custom version of a template that includes the &BODY token, you must include the &BODY token in the custom template as well, in order to include the body text of the particular notification that is being sent into the template outline.

If you add a new token in a custom template, you must set up the necessary substitution yourself. By default, a notification mailer only performs token substitution for the standard tokens that are listed for the default templates.

Oracle Workflow provides the following message templates.

> **Note:** In addition to the message templates listed here, the System: Mailer item type also includes some other messages which are not currently used.

- Workflow Open Mail (Templated) Message, page 2-86

- Orig. Workflow Open Mail (Templated) Message, page 2-88

- Workflow Open Mail (Direct) Message, page 2-90

- Orig. Workflow Open Mail (Direct) Message, page 2-93

- Workflow Open Mail (Outlook Express) Message, page 2-95

- Orig. Workflow Open Mail (Outlook Express) Message, page 2-98

- Workflow Open FYI Mail Message, page 2-100

- Orig. Workflow Open FYI Mail Message, page 2-102

- Workflow View From UI Message, page 2-103

- Workflow View FYI From UI Message, page 2-104

- Workflow URL Attachment Message, page 2-106

- Workflow Canceled Mail Message, page 2-107

- Orig. Workflow Canceled Mail Message, page 2-108

- Workflow Invalid Mail Message, page 2-109

- Orig. Workflow Invalid Mail Message, page 2-112

- Workflow Closed Mail Message, page 2-115

- Orig. Workflow Closed Mail Message, page 2-116

- Workflow Summary Mail (HTML) Message, page 2-117

- Workflow Warning Mail Message, page 2-118

- Workflow Signature Required Mail Message, page 2-119

- Workflow Signature Warning Mail Message, page 2-122

- Workflow Secure Mail Content Message, page 2-123

- Workflow Open Mail (More Information Request) Message, page 2-124

- Orig. Workflow Open Mail (More Information Request) Message, page 2-126

- Workflow More Information Request (Outlook Express) Message, page 2-128

- Workflow Invalid Open Mail (More Information Request) Message, page 2-131

- User Notification Preference Update Report Message, page 2-134

- Notification Mailer Override Address Verification Message, page 2-135

See: Reviewing Notifications via Electronic Mail, *Oracle Workflow User's Guide*.

Workflow Open Mail (Templated) Message:

If you use the templated response method, the Notification System uses the Workflow Open Mail (Templated) message as a default template for e-mail notifications that require a response. The notification template includes generic instructions on how to respond to a notification. It also includes the date that a response is due and any history of actions on the notification.

> **Note:** The templated response method is the default response method for Oracle Workflow. Notification mailers use the templated response method unless you select the Direct Response parameter in the notification mailer configuration wizard. See: Notification Mailer Configuration Wizard, page 7-30.

The response instructions in the plain text message body describe how to reply manually using the templated response method. This message is used for notifications sent to performers with a notification preference of MAILTEXT or MAILATTH. The response instructions in the HTML-formatted message body describe how to reply using the automatically generated response template. This message is used for notifications sent to performers with a notification preference of MAILHTML or MAILHTM2, and is also attached to notifications sent to performers with a notification preference of MAILATTH.

The Workflow Open Mail (Templated) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
BODY	The text of the body defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
RESPONSE	The user response section as defined by the Respond message attributes in the actual notification message definition.
MAILTO	The content of the HTML tag that a recipient would click on to respond to a notification. This attribute is used only for HTML e-mail notifications.
CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is not currently used.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist. This attribute is not currently used.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
HISTORY	The history of actions on the notification.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.

TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow Open Mail (Templated) message, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE
_____Start of Response Template_____

Response Template for &NOTIFICATION

To submit your response, reply to this message, including this response
template with your reply.  Copy and paste from this message if
necessary to obtain an editable copy of the template. Insert your
response value between the quotes following each response prompt.


&RESPONSE

_____End of Response Template_____

Notification Details:
&HEADER
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>&HEADER
<P>&BODY
<P>Please click on one of the following choices to automatically
generate an E-mail response.  Before sending the E-mail response to
close this notification, ensure all response prompts include a
desired response value within quotes.
<P>&MAILTO
<P>&HISTORY
</BODY>
</HTML>
```

Orig. Workflow Open Mail (Templated) Message:

Oracle Workflow provides the Orig. Workflow Open Mail (Templated) message as an alternative template that you can optionally use as a template for e-mail notifications

that require a response if you use the templated response method. This template does not include the header attributes that are displayed in the Workflow Open Mail (Templated) message.

The Orig. Workflow Open Mail (Templated) notification template includes generic instructions on how to respond to a notification. It also includes the following information about a message: the name of the sender of the message, message priority, date that a response is due, and any comments from the sender or, if the notification is forwarded from another user, any comments from the forwarder.

> **Note:** The templated response method is the default response method for Oracle Workflow. Notification mailers use the templated response method unless you select the Direct Response parameter in the notification mailer configuration wizard. See: Notification Mailer Configuration Wizard, page 7-30 .

The response instructions in the plain text message body describe how to reply manually using the templated response method. This message is used for notifications sent to performers with a notification preference of `MAILTEXT` or `MAILATTH`. The response instructions in the HTML-formatted message body describe how to reply using the automatically generated response template. This message is used for notifications sent to performers with a notification preference of `MAILHTML` or `MAILHTM2`, and is also attached to notifications sent to performers with a notification preference of `MAILATTH`.

The Orig. Workflow Open Mail (Templated) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
BODY	The text of the body defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
RESPONSE	The user response section as defined by the Respond

message attributes in the actual notification message definition.

MAILTO   The content of the HTML tag that a recipient would click on to respond to a notification. This attribute is used only for HTML e-mail notifications.

CLICK_HERE_RESPONSE   The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is not currently used.

SENDER   The name of the sender of the message, as displayed in the From column in the Worklist.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
From: &SENDER
&COMMENT
_____Start of Response Template_____

Response Template for &NOTIFICATION

To submit your response, reply to this message, including this response
template with your reply.  Copy and paste from this message if
necessary to obtain an editable copy of the template. Insert your
response value between the quotes following each response prompt.


&RESPONSE

_____End of Response Template_____

Notification Details:
&BODY

Due Date: &DUE_DATE
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE> </HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P>
<P>From: <B>&SENDER</B>
<P><B><FONT SIZE=+1>&COMMENT</FONT> </B>
<P>&BODY
<P><B>Please click on one of the following choices to automatically
generate an E-mail response.  Before sending the E-mail response to
close this notification, ensure all response prompts include a
desired response value within quotes.</B>
<P>&MAILTO
</BODY>
</HTML>
```

Workflow Open Mail (Direct) Message:

If you select the direct response method, the Notification System uses the Workflow

Open Mail (Direct) message as a default template for e-mail notifications that require a response. The notification template includes generic instructions on how to respond to a notification. It also includes the date that a response is due and any history of actions on the notification.

> **Note:** To select the direct response method for a notification mailer, you must select the Direct Response parameter in the notification mailer configuration wizard. See: Notification Mailer Configuration Wizard, page 7-30.

The response instructions in the plain text message body describe how to reply using the direct response method. This message is used for notifications sent to performers with a notification preference of `MAILTEXT` or `MAILATTH`. The response instructions in the HTML-formatted message body describe how to reply using the automatically generated response template. This message is used for notifications sent to performers with a notification preference of `MAILHTML` or `MAILHTM2`, and is also attached to notifications sent to performers with a notification preference of `MAILATTH`.

> **Note:** Responses that are generated automatically from an HTML-formatted notification or attachment always use a response template, regardless of which response method you select.

The Workflow Open Mail (Direct) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
BODY	The text of the body defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
RESPONSE	The user response section as defined by the Respond message attributes in the actual notification message definition.

MAILTO	The content of the HTML tag that a recipient would click on to respond to a notification. This attribute is used only for HTML e-mail notifications.
CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is not currently used.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist. This attribute is not currently used.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
HISTORY	The history of actions on the notification.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow Open Mail (Direct) message, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE
_____

Response Instructions for &NOTIFICATION

To submit your response, reply to this message, including this note
with your reply.  The first lines of your reply must be your responses
to the notification questions.  Instructions below detail exactly what
should be placed on each line of your reply.

&RESPONSE


_____

Notification Details:
&HEADER
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML>
<HEAD>
<TITLE> Oracle Workflow Notification </TITLE>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>&HEADER
<P>&BODY
<P>Please click on one of the following choices to automatically
generate an E-mail response.  Before sending the E-mail response
to close this notification, ensure all response prompts include
a desired response value within quotes.
<P>&MAILTO
<P>&HISTORY
</BODY>
</HTML>
```

Orig. Workflow Open Mail (Direct) Message:

Oracle Workflow provides the Orig. Workflow Open Mail (Direct) message as an alternative template that you can optionally use as a template for e-mail notifications that require a response if you select the direct response method. This template does not include the header attributes that are displayed in the Workflow Open Mail (Direct) message.

The Orig. Workflow Open Mail (Direct) notification template includes generic instructions on how to respond to a notification. It also includes the following information about a message: the name of the sender of the message, message priority, date that a response is due, and any comments from the sender of the message or, if the notification is forwarded from another user, any comments from the forwarder.

> **Note:** To select the direct response method for a notification mailer, you must select the Direct Response parameter in the notification mailer configuration wizard. See: Notification Mailer Configuration Wizard, page 7-30.

The response instructions in the plain text message body describe how to reply using the direct response method. This message is used for notifications sent to performers with a notification preference of MAILTEXT or MAILATTH. The response instructions in the HTML-formatted message body describe how to reply using the automatically generated response template. This message is used for notifications sent to performers with a notification preference of MAILHTML or MAILHTM2, and is also attached to notifications sent to performers with a notification preference of MAILATTH.

> **Note:** Responses that are generated automatically from an

HTML-formatted notification or attachment always use a response template, regardless of which response method you select.

The Orig. Workflow Open Mail (Direct) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
BODY	The text of the body defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
RESPONSE	The user response section as defined by the Respond message attributes in the actual notification message definition.
MAILTO	The content of the HTML tag that a recipient would click on to respond to a notification. This attribute is used only for HTML e-mail notifications.
CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is not currently used.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
From: &SENDER
&COMMENT
```
_____

```
Response Instructions for &NOTIFICATION

To submit your response, reply to this message, including this note
with your reply.  The first lines of your reply must be your responses
to the notification questions.  Instructions below detail exactly what
should be placed on each line of your reply.

&RESPONSE
```

_____

```
Notification Details:
&BODY

Due Date: &DUE_DATE
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE> </HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P>
<P>From: <B>&SENDER</B>
<P><B><FONT SIZE=+1>&COMMENT</FONT> </B>
<P>&BODY
<P><B>Please click on one of the following choices to automatically
generate an E-mail response.  Before sending the E-mail response to
close this notification, ensure all response prompts include a
desired response value within quotes.</B>
<P>&MAILTO
</BODY>
</HTML>
```

Workflow Open Mail (Outlook Express) Message:

If you use an e-mail application such as Microsoft Outlook Express as your e-mail
client, you should select the standard Workflow Open Mail (Outlook Express) message
as a template for e-mail notifications that require a response, for users with a
notification preference of `MAILHTML`, `MAILHTM2`, or `MAILATTH`. This message includes
a link to the Notification Details Web page to let users respond to the notification there.
This template is provided to accommodate e-mail applications that cannot process the
response links included in the Workflow Open Mail (Templated) and Workflow Open
Mail (Direct) templates.

> **Note:** If you select the Workflow Open Mail (Outlook Express) message
> template for a notification mailer, then you should also select the
> Workflow More Information Request (Outlook Express) message
> template for that notification mailer. See: Workflow More Information
> Request (Outlook Express) Message, page 2-128.

The response instructions in the plain text message body describe how to reply

manually using the templated response method. This message is used for notifications sent to performers with a notification preference of MAILATTH. The HTML-formatted message body includes a link called "Click here to respond" which lets users access the notification in the Notification Details Web page to respond to the notification. This message is used for notifications sent to performers with a notification preference of MAILHTML or MAILHTM2, and is also attached to notifications sent to performers with a notification preference of MAILATTH.

> **Note:** When users choose the "Click here to respond" link, it automatically attempts to establish a Web session with the Web server. Users must be able to connect to the Web server to use this link to respond to a notification. Users must log in to Oracle Workflow to access the Notification Details page, unless you enable guest access in Oracle Applications. See: Reviewing Notifications via Electronic Mail, *Oracle Workflow User's Guide* and Responses Through the Notification Detail Link Attachment, page 2-69.

The Workflow Open Mail (Outlook Express) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
BODY	The text of the body defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
RESPONSE	The user response section as defined by the Respond message attributes in the actual notification message definition.
MAILTO	The content of the HTML tag that a recipient would click on to respond to a notification. This attribute is not currently used.

CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is used only for HTML e-mail notifications.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist. This attribute is not currently used.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
HISTORY	The history of actions on the notification.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow Open Mail (Outlook Express) message, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE

---------- Start of Response Template
-----------------------------------------

Response Template for &NOTIFICATION

To submit your response, reply to this message including this
response template in your reply.  Insert your response value
between the quotes following each response prompt.


&RESPONSE

---------- End of Response Template
-------------------------------------------

Notification Details:
&HEADER
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE> </HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>&HEADER
<P>&BODY
<P>&CLICK_HERE_RESPONSE
<P>&HISTORY
</BODY>
</HTML>
```

Orig. Workflow Open Mail (Outlook Express) Message:

Oracle Workflow provides the Orig. Workflow Open Mail (Outlook Express) message as an alternative template that you can optionally use as a template for e-mail notifications that require a response if you use an e-mail application such as Microsoft Outlook Express as your e-mail client. This template does not include the header attributes that are displayed in the Workflow Open Mail (Outlook Express) message.

The Orig. Workflow Open Mail (Outlook Express) message includes the name of the sender of the message, any comments from the sender or forwarder, and a link to the Notification Details Web page to let users respond to the notification there. This template can be used to accommodate e-mail applications that cannot process the response links included in the Orig. Workflow Open Mail (Templated) and Orig. Workflow Open Mail (Direct) templates.

> **Note:** If you select the Orig. Workflow Open Mail (Outlook Express) message template for a notification mailer, then you should also select the Workflow More Information Request (Outlook Express) message template for that notification mailer. See: Workflow More Information Request (Outlook Express) Message, page 2-128.

The response instructions in the plain text message body describe how to reply manually using the templated response method. This message is used for notifications sent to performers with a notification preference of MAILATTH. The HTML-formatted message body includes a link called "Click here to respond" which lets users access the notification in the Notification Details Web page to respond to the notification. This message is used for notifications sent to performers with a notification preference of MAILHTML or MAILHTM2, and is also attached to notifications sent to performers with a notification preference of MAILATTH.

> **Note:** When users choose the "Click here to respond" link, it automatically attempts to establish a Web session with the Web server. Users must be able to connect to the Web server to use this link to respond to a notification. Users must log in to Oracle Workflow to

access the Notification Details page, unless you enable guest access in Oracle Applications. See: Reviewing Notifications via Electronic Mail, *Oracle Workflow User's Guide* and Responses Through the Notification Detail Link Attachment, page 2-69.

The Orig. Workflow Open Mail (Outlook Express) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
BODY	The text of the body defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
RESPONSE	The user response section as defined by the Respond message attributes in the actual notification message definition.
MAILTO	The content of the HTML tag that a recipient would click on to respond to a notification. This attribute is not currently used.
CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is used only for HTML e-mail notifications.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
From: &SENDER
&COMMENT
---------- Start of Response Template
-----------------------------------------


Response Template for &NOTIFICATION

To submit your response, reply to this message including this response
template in your reply.  Insert your response value between the quotes
following each response prompt.


&RESPONSE


---------- End of Response Template
-------------------------------------------

Notification Details:
&BODY

Due Date: &DUE_DATE
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE> </HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P>
<P>From: <B>&SENDER</B>
<P><B><FONT SIZE=+1>&COMMENT</FONT> </B>
<P>&BODY
<P>&CLICK_HERE_RESPONSE
</BODY>
</HTML>
```

Workflow Open FYI Mail Message:

The Notification System uses the Workflow Open FYI Mail message as a default
template for all e-mail notifications that do not require a response. The template
indicates that the notification is for your information (FYI) and does not require a
response. In addition to the message, the template also includes any history of actions
on the notification.

The Workflow Open FYI Mail message has the following message attributes. The values
are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
BODY	The text of the body defined in the message.
COMMENT	Comments added by the sender or the forwarder.

PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist. This attribute is not currently used.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
HISTORY	The history of actions on the notification.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the text that appears in the body of the Workflow Open FYI Mail template, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent. The boilerplate text for the plain text message body is as follows:

```
Oracle Workflow Notification (FYI)

&TIMEZONE
_____

&HEADER
&BODY

&HISTORY
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<HTML><HEAD><TITLE>Oracle Workflow Notification (FYI)</TITLE><STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE></HEAD>
<BODY BGCOLOR="#FFFFFF">
<SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>&HEADER
<P>&BODY
<P>&HISTORY
</BODY>
</HTML>
```

Orig. Workflow Open FYI Mail Message:

Oracle Workflow provides the Orig. Workflow Open FYI Mail message as an alternative template that you can optionally use as a template for e-mail notifications that do not require a response. This template does not include the header attributes that are displayed in the Workflow Open FYI Mail message.

The Orig. Workflow Open FYI Mail template indicates that the notification is for your information (FYI) and does not require a response. In addition to the message, the template also includes the name of the sender of the message and any comments from the sender or forwarder.

The Orig. Workflow Open FYI Mail message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
BODY	The text of the body defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist.

The boilerplate text for the plain text message body is as follows:

```
Oracle Workflow Notification (FYI)
From: &SENDER
&COMMENT
_____

&BODY
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<HTML><HEAD></HEAD>
<BODY BGCOLOR="#FFFFFF"><b>Oracle Workflow Notification (FYI)</b>
<br>
From: <B>&SENDER</B>
<br>&COMMENT
<hr>
<P>&BODY
</BODY>
</HTML>
```

Workflow View From UI Message:

Oracle Workflow provides the Workflow View From UI message as an alternative template that you can optionally use for open response-required notifications whose content you do not want to send in e-mail. An e-mail message generated from this template will still include the header attributes for the notification, as well as any history of actions on the notification. However, the e-mail message will exclude the message body for the notification and will not enable responses through e-mail. Users can only view and respond to such notifications through the Notification Details Web page.

The notification template informs the recipient that the notification is best viewed from the Web page and directs the recipient to access the online version of the notification.

The Workflow View From UI message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist. This attribute is not currently used.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.

HISTORY	The history of actions on the notification.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow View From UI message, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE

Notification Details:
&HEADER

This notification is best viewed from the Notification Detail page.
Please access the online version of this notification.

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>&HEADER
<P>This notification is best viewed from the Notification Detail
page.<BR>
Please access the online version of this notification.
<P>&HISTORY
</BODY>
</HTML>
```

Workflow View FYI From UI Message:

Oracle Workflow provides the Workflow View FYI From UI message as an alternative template that you can optionally use for open FYI notifications whose content you do not want to send in e-mail. An e-mail message generated from this template will still include the header attributes for the notification, as well as any history of actions on the notification. However, the e-mail message will exclude the message body for the notification. Users can only view such notifications through the Notification Details Web page.

The notification template informs the recipient that the notification is best viewed from

the Web page and directs the recipient to access the online version of the notification.

The Workflow View FYI From UI message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

START_DATE	The date the message is sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line defined in the message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist. This attribute is not currently used.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
HISTORY	The history of actions on the notification.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow View FYI From UI message, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE

Notification Details:
&HEADER

This notification is best viewed from the Notification Detail page.
Please access the online version of this notification.

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>&HEADER
<P>This notification is best viewed from the Notification Detail
page.<BR>
Please access the online version of this notification.
<P>&HISTORY
</BODY>
</HTML>
```

Workflow URL Attachment Message:

The Notification System uses the Workflow URL Attachment message as a default template to create the Notification References attachment for HTML-formatted notification messages that include URL attributes with Attach Content checked. The template includes a list with links to each URL.

> **Note:** The Workflow URL Attachment message template is used only for notifications that do not include embedded Oracle Application Framework regions. For a notification that includes an embedded region, Oracle Workflow includes the Related Applications region in the e-mail message, instead of appending the Notification References attachment.

The Workflow URL Attachment message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

URLLIST	The list of URLs to be included in the attachment.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the text that appears in the body of the Workflow URL Attachment template, where an attribute preceded by an ampersand (&) is token substituted with a

runtime value when the notification is sent. The boilerplate text for the
HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification References </TITLE>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE> </HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><h2 class="OraHeader">Notification References</h2>

<HR WIDTH="100%">
<BR>
&URLLIST
<BR>
<HR WIDTH="100%">
<BR> 
</BODY>
</HTML>
```

Workflow Canceled Mail Message:

The default Workflow Canceled Mail message informs the recipient that a previously
sent notification is canceled. You can set the Send E-mails for Canceled Notifications
mailer parameter in the notification mailer configuration wizard to determine whether
or not notification mailers should send notification cancellation messages. See:
Notification Mailer Configuration Wizard, page 7-30.

The Workflow Canceled Mail message has the following message attributes, with
values that are drawn from the message definition associated with the canceled
notification activity:

START_DATE	The date the original message was sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line of the original message.
BODY	The text of the original message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message.

See: Header Attributes, *Oracle Workflow Developer's Guide*.

TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

The boilerplate text for the plain text message body is as follows:

```
&TIMEZONE

You earlier received the notification shown below.  That
notification is now canceled, and no longer requires your
response.  You may simply delete it along with this message.
_____

&HEADER
&BODY
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<html><Head><STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</Head>
<body>
<SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>You earlier received the notification shown below.  That
notification is now canceled, and no longer requires your
response.  You may simply delete it along with this message.
<hr>
&HEADER
<P>&BODY
</body></html>
```

Orig. Workflow Canceled Mail Message:

Oracle Workflow provides the Orig. Workflow Canceled Mail message as an alternative template that you can optionally use to inform the recipient that a previously sent notification is canceled. This template does not include the header attributes that are displayed in the Workflow Canceled Mail message.

You can set the Send E-mails for Canceled Notifications mailer parameter in the notification mailer configuration wizard to determine whether or not notification mailers should send notification cancellation messages. See: Notification Mailer Configuration Wizard, page 7-30.

The Orig. Workflow Canceled Mail message has the following message attributes, with values that are drawn from the message definition associated with the canceled notification activity:

START_DATE	The date the original message was sent.

TO	The role the notification is sent to; the performer.
SUBJECT	The subject line of the original message.
BODY	The text of the original message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.

The boilerplate text for the plain text message body is as follows:

```
You earlier received the notification shown below.  That
notification is now canceled, and no longer requires your
response.  You may simply delete it along with this message.
_____

&BODY
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<html><Head></Head><body>You earlier received the notification
shown below.  That notification is now canceled, and no longer
requires your response.  You may simply delete it along with
this message.
<hr>
&BODY
</body></html>
```

Workflow Invalid Mail Message:

The Workflow Invalid Mail message is sent to a user by default when a user responds incorrectly to a notification. For example, if a response message from a user contains a valid notification ID (NID) line matching it with a notification, but does not contain any response value or contains an invalid response value, the notification mailer sends a Workflow Invalid Mail message to the user. This message describes how to respond to the notification correctly. The message attributes are as follows:

START_DATE	The date the original message was sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line of the original message.
BODY	The text of the original message.
COMMENT	Comments added by the sender or the forwarder.

PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified by the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
RESPONSE	The user response section as defined by the Respond message attributes in the original message definition.
MAIL_ERROR_MESSAGE	An error message that the mail program generates if an error occurs upon processing the response.
MAIL_ERROR_STACK	An error stack of arguments that the mail program generates if an error occurs upon processing the response. This attribute is not currently used.
CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is not currently used.
MAIL_VALUE_FOUND	The invalid response value found in the user's response message.
MAIL_EXP_VALUES	Information about the expected valid response values.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
HISTORY	The history of actions on the notification.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

The boilerplate text for the plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE

Warning: Your previous response to this message was invalid
(see error message below).  Please resubmit your response.

Important: Some mail clients, notably early releases of Microsoft
Outlook Express, may not copy the 'NID' line properly in your
response. Please verify that the 'NID' line is included in full
and contains the prefix 'NID' and all the details between the
square brackets when responding.

Error Message: &MAIL_ERROR_MESSAGE

Value Found: &MAIL_VALUE_FOUND

Remarks: &MAIL_EXP_VALUES

_____

Response Instructions for &NOTIFICATION

To submit your response, reply to this message, including this
original with your reply.  This note contains a special 'NID'
string that is required to process the reponse.  The first lines
of your reply must be your responses to the notification questions.
You should enter one line for each response required by the
notfication; any additional lines will be ignored.  You may leave
a line blank to accept the default value for that specific
response.  You must supply a value or a blank line for each
question asked.  Instructions below detail exactly what should be
placed on each line of your reply.

&RESPONSE
_____

Notification Details:
&HEADER
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<html>
<Head>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</Head>
<body>
<SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<p>Warning: Your previous response to this message was
invalid (see error message below).  Please resubmit your
response.
<P>Error Message: &MAIL_ERROR_MESSAGE
<BR>
<BR>Value Found: &MAIL_VALUE_FOUND
<BR>
<BR>Remarks: &MAIL_EXP_VALUES
<HR>
<P>&HEADER
<P>&BODY
<P><B>Please click on one of the following choices to
automatically generate an E-mail response.  Before sending the
E-mail response to close this notification, ensure all response
prompts include a desired response value within quotes.</B>
<P><B>Important:</B> Some mail clients, notably early releases
of Microsoft Outlook Express, may not copy the 'NID' line
properly in your response. Please verify that the 'NID' line
is included in full and contains the prefix 'NID' and all the
details between the square brackets when responding.
<P>&MAILTO
<P>&HISTORY
</BODY>
</HTML>
```

Orig. Workflow Invalid Mail Message:

Oracle Workflow provides the Orig. Workflow Invalid Mail message as an alternative
template that you can optionally use when a user responds incorrectly to a notification.
This template does not include the header attributes that are displayed in the Workflow
Invalid Mail message.

The Orig. Workflow Invalid Mail message describes how to respond to the notification
correctly. The message attributes are as follows:

START_DATE	The date the original message was sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line of the original message.
BODY	The text of the original message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.

DUE_DATE	The date by which a response is required, specified by the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
RESPONSE	The user response section as defined by the Respond message attributes in the original message definition.
MAIL_ERROR_MESSAGE	An error message that the mail program generates if an error occurs upon processing the response.
MAIL_ERROR_STACK	An error stack of arguments that the mail program generates if an error occurs upon processing the response. This attribute is not currently used.
CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is not currently used.
MAIL_VALUE_FOUND	The invalid response value found in the user's response message.
MAIL_EXP_VALUES	Information about the expected valid response values.

The boilerplate text for the plain text message body is as follows:

```
Oracle Workflow Notification
&COMMENT

Warning: Your previous response to this message was invalid
(see error message below).  Please resubmit your response.

Important: Some mail clients, notably early releases of
Microsoft Outlook Express, may not copy the 'NID' line
properly in your response. Please verify that the 'NID'
line is inlcuded in full and contains the prefix 'NID' and
all the details between the square brackets when responding.

Error Message: &MAIL_ERROR_MESSAGE

Value Found: &MAIL_VALUE_FOUND

Remarks: &MAIL_EXP_VALUES

_____


Response Instructions for &NOTIFICATION

To submit your response, reply to this message, including
this original with your reply.  This note contains a special
'NID' string that is required to process the reponse.  The
first lines of your reply must be your responses to the
notification questions.  You should enter one line for each
response required by the notfication; any additional lines
will be ignored.  You may leave a line blank to accept the
default value for that specific response.  You must supply
a value or a blank line for each question asked.
Instructions below detail exactly what should be placed on
each line of your reply.

&RESPONSE
_____

Notification Details:
&BODY

Due Date: &DUE_DATE
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<html><Head></Head><body>Warning: Your previous response
to this message was invalid (see error message below).
Please resubmit your response.
<P>Error Message: &MAIL_ERROR_MESSAGE
<BR>
<BR>Value Found: &MAIL_VALUE_FOUND
<BR>
<BR>Remarks: &MAIL_EXP_VALUES
<HR><P><B><FONT SIZE=+1>&COMMENT</FONT> </B>
<P>&BODY
<P><B>Please click on one of the following choices to
automatically generate an E-mail response.  Before sending
the E-mail response to close this notification, ensure all
response prompts include a desired response value within
quotes.</B>
<P><B>Important:</B> Some mail clients, notably early
releases of Microsoft Outlook Express, may not copy the
'NID' line properly in your response. Please verify that
the 'NID' line is inlcuded in full and contains the prefix
'NID' and all the details between the square brackets when
responding.
<P>&MAILTO
</BODY> </HTML>
```

Workflow Closed Mail Message:

The default Workflow Closed Mail message informs the recipient that a previously sent notification is now closed. It has the following message attributes, with values that are drawn from the message definition associated with the closed notification activity:

START_DATE	The date the original message was sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line of the original message.
BODY	The text of the original message.
COMMENT	Comments added by the sender or the forwarder.
PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
TIMEZONE	The time zone of the dates and times displayed in the

notification, based on the time zone of the server from which the notification was sent.

TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

The boilerplate text for the plain text message body is as follows:

```
&TIMEZONE

You earlier received the notification shown below.  That
notification is now closed, and no longer requires your
response.  You may simply delete it along with this message.
_____

&HEADER
&BODY
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<html><Head><STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</Head>
<body><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>You earlier received the notification shown below.  That
notification is now closed, and no longer requires your
response.  You may simply delete it along with this message.
<hr>
&HEADER
<P>&BODY
</body></html>
```

Orig. Workflow Closed Mail Message:

Oracle Workflow provides the Orig. Workflow Closed Mail message as an alternative template that you can optionally use to inform the recipient that a previously sent notification is now closed. This template does not include the header attributes that are displayed in the Workflow Closed Mail message.

The Orig. Workflow Closed Mail message has the following message attributes, with values that are drawn from the message definition associated with the closed notification activity:

START_DATE	The date the original message was sent.
TO	The role the notification is sent to; the performer.
SUBJECT	The subject line of the original message.
BODY	The text of the original message.
COMMENT	Comments added by the sender or the forwarder.

PRIORITY	The priority of the notification message.
DUE_DATE	The date by which a response is required, specified in the notification activity.
NOTIFICATION	Required notification code used to identify the information in the notification.

The boilerplate text for the plain text message body is as follows:

```
You earlier received the notification shown below.  That
notification is now closed, and no longer requires your
response.  You may simply delete it along with this message.
_____

&BODY
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<html><Head></Head><body>You earlier received the
notification shown below.  That notification is now
closed, and no longer requires your response.  You
may simply delete it along with this message.
<hr>
&BODY
</body></html>
```

Workflow Summary Mail (HTML) Message:

The Notification System uses the Workflow Summary Mail (HTML) message by default as a template to send a summary of workflow notifications to users and roles that have their notification preference set to SUMMARY or SUMHTML in the Oracle Workflow directory service. The Workflow Summary Mail (HTML) message summarizes all currently open notifications for a given user/role. The HTML-formatted message body also provides a link to the Worklist Web page as well as links to each notification in the Notification Details Web page.

The Workflow Summary Mail (HTML) message has the following message attributes, with values that are drawn from the message definition associated with the open notification activity:

SUMMARY	Summary report.
USER_NAME	The user/role the notification summary is sent to; the performer.
SYSDATE	The current date.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the

HTML-formatted message body.

The plain text message body is used for notifications sent to performers with a notification preference of SUMMARY. The boilerplate text for the plain text message body is as follows:

```
&TIMEZONE
&SUMMARY
```

The HTML-formatted message body is used for notifications sent to performers with a notification preference of SUMHTML. The boilerplate text for the HTML-formatted message body is as follows:

```
<HTML><HEAD>
<STYLE>
&TEMPLATE_STYLE
</STYLE>
<TITLE>Summary Notification</TITLE>
</HEAD>
<BODY><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>&SUMMARY
</BODY>
</HTML>
```

Workflow Warning Mail Message:

The Notification System uses the Workflow Warning Mail message as a default template to send a message to a user the first time it receives unsolicited mail from that user. For example, if a message from a user does not contain a notification ID (NID) line matching it with a notification, or contains an incorrectly formatted NID line, the notification mailer sends a Workflow Warning Mail message to the user.

> **Note:** You can optionally use the Send Warning for Unsolicited E-mail mailer parameter to prevent notification mailers from sending any warning messages. See: Notification Mailer Configuration Wizard, page 7-30.

The Workflow Warning Mail message has the following message attributes, with values that are drawn from the unsolicited mail:

UBODY	The text of the unsolicited mail message body.
USUBJECT	The text of the unsolicited mail subject line.
UFROM	The address of the user that sent the unsolicited mail.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

The boilerplate text for the plain text message body is as follows:

```
Messages sent to this account are processed automatically
by the Oracle Workflow Notification Mailer.  The message
you sent did not appear to be in response to a notification.
If you are responding to a notification, please use the
response template that was included with your notification.
Take care to include the 'NID' line of the template in your
reply.  If you are not responding to a notification, please
do not send mail to this account.

Important: Some mail clients, notably early releases of
Microsoft Outlook Express, may not copy the 'NID' line
properly in your response. Please verify that the 'NID' line
is included in full and contains the prefix 'NID' and all
the details between the square brackets when responding.
```

_____

```
From: &UFROM
Subject: &USUBJECT

&UBODY
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<html><head>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</head><body>
<B>Messages sent to this account are processed automatically
by the Oracle Workflow Notification Mailer.  The message you
sent did not appear to be in response to a notification.  If
you are responding to a notification, please use the
auto-generated reply created when responding to the original
message. This contains the 'NID' line which is necessary for
identification.  If you are not responding to a notification,
please do not send mail to this account.
<P><B>Important:</B> Some mail clients, notably early
releases of Microsoft Outlook Express, may not copy the 'NID'
line properly in your response. Please verify that the 'NID'
line is included in full and contains the prefix 'NID' and
all the details between the square brackets when responding.
<hr>
<P>From: &UFROM
<BR>Subject: &USUBJECT

<P>&UBODY

</body></html>
```

Workflow Signature Required Mail Message:

The Notification System uses the Workflow Signature Required Mail message as a
default template for e-mail notifications that require an electronic signature in the user's
response. Users can only respond to such notifications through the Notification Details
Web page, where they can enter either a password-based signature or a
certificate-based digital signature, depending on the notification's requirements, to sign

the response. The notification template informs the recipient that a signature is required and that the response cannot be submitted through e-mail. Instead, the notification template directs the recipient to access the online version of the notification to submit a reponse.

The plain text message body is used for notifications sent to performers with a notification preference of `MAILTEXT` or `MAILATTH`. The HTML-formatted message body additionally includes a link called "Click here to respond" which lets users access the notification in the Notification Details Web page to respond. This message is used for notifications sent to performers with a notification preference of `MAILHTML` or `MAILHTM2`, and is also attached to notifications sent to performers with a notification preference of `MAILATTH`.

> **Note:** When users choose the "Click here to respond" link, it automatically attempts to establish a Web session with the Web server. Users must be able to connect to the Web server to use this link to respond to a notification. Users must also log in to Oracle Workflow to access the Notification Details page. See: Reviewing Notifications via Electronic Mail, *Oracle Workflow User's Guide*.
>
> However, if you enable guest access for links to the Notification Details page, the "Click here to respond" link does not appear in the HTML-formatted message body. When electronic signatures are required, users must log in with their own user names and passwords to enter their signatures with their notification responses. See: Responses Through the Notification Detail Link Attachment, page 2-69.

The Workflow Signature Required Mail message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

DUE_DATE	The date by which a response is required, specified in the notification activity.
BODY	The text of the body defined in the message.
NOTIFICATION	Required notification code used to identify the information in the notification. For notifications that require an electronic signature to be entered online, this notification code does not include the access key or node identifier information that would be needed for an e-mail response.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist. This attribute is not currently used.
SUBJECT	The subject line defined in the message.

HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
HISTORY	The history of actions on the notification.
COMMENT	Comments added by the sender or the forwarder.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a notification. This attribute is used only for HTML e-mail notifications.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow Signature Required Mail message, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE

This notification requires a signature in your response.
You cannot respond to this notification through e-mail. Please
access the online version of the notification to submit your
response.

Notification Details:
&HEADER
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE> <STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>&HEADER
<P>&BODY
<P><B>This notification requires a signature in your response.
You cannot respond to this notification through e-mail. Please
access the online version of the notification to submit your
response.</B>
<P>&CLICK_HERE_RESPONSE
<P>&HISTORY
</BODY>
</HTML>
```

See: #WF_SIG_POLICY Attribute, *Oracle Workflow Developer's Guide*.

Workflow Signature Warning Mail Message:

The Workflow Signature Warning Mail message is sent to a user by default if that user sends an e-mail response containing the notification ID (NID) line of a notification that requires an electronic signature. A valid response to such a notification can only be submitted through the Notification Details Web page. The notification template informs the recipient that a signature is required and that the response cannot be submitted through e-mail. Instead, the notification template directs the recipient to access the online version of the notification to submit a response.

The Workflow Signature Warning Mail message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

SUBJECT	The subject line of the original message.
COMMENT	Comments added by the sender or the forwarder.
BODY	The text of the original message.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow

Signature Warning Mail message, where attributes preceded by an ampersand (`&`) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE
&COMMENT

Warning: You earlier received the notification shown below.
This notification requires a signature in your response. You
cannot respond to this notification through e-mail. Please
access the online version of the notification to submit your
response.
_____

&HEADER
&BODY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<html><Head><STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</Head>
<body><P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>Warning: You earlier received the notification shown below.
This notification requires a signature in your response. You
cannot respond to this notification through e-mail. Please
access the online version of the notification to submit your
response.
<hr>
<P><B><FONT SIZE=+1>&COMMENT</FONT> </B>
<P>&HEADER
<P>&HISTORY
<P>&BODY
</body></html>
```

See: #WF_SIG_POLICY Attribute, *Oracle Workflow Developer's Guide*.

Workflow Secure Mail Content Message:

The Notification System uses the Workflow Secure Mail Content message as a default template for notifications that include sensitive content that cannot be sent in e-mail for security reasons. You can mark notifications as including secure content using the special `#WF_SECURITY_POLICY` message attribute. Users can only view and respond to such notifications through the Notification Details Web page. The notification template informs the recipient that the notification content cannot be sent in e-mail and directs the recipient to access the online version of the notification instead.

The Workflow Secure Mail Content message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

NOTIFICATION   Required notification code used to identify the information in the notification.

SUBJECT	The subject line defined in the message.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow Secure Mail Content message, where attributes preceded by an ampersand (`&`) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
_____

Notification &NOTIFICATION

This notification contains secure content which cannot be sent
through e-mail. Please access the online version of the
notification to see the details.
_____
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P>
<P>Notification: &NOTIFICATION <br><br>
<B>This notification contains secure content which cannot be
sent through e-mail. Please access the online version of the
notification to see the details.</B>
</BODY>
</HTML>
```

See: #WF_SECURITY_POLICY Attribute, *Oracle Workflow Developer's Guide*.

Workflow Open Mail (More Information Request) Message:

The Notification System uses the Workflow Open Mail (More Information Request) message as a default template to send a request for more information about a notification from one user to another user. The notification template includes generic instructions on how to respond with the requested information. It also includes the following information about a message: the name of the sender of the message, any history of actions on the notification, and the date that a response is due.

The response instructions in the plain text message body describe how to reply manually. This message is used for notifications sent to performers with a notification preference of MAILTEXT or MAILATTH. The response instructions in the HTML-formatted message body describe how to reply using the automatically generated response template. This message is used for notifications sent to performers with a notification preference of MAILHTML or MAILHTM2, and is also attached to

notifications sent to performers with a notification preference of `MAILATTH`.

The Workflow Open Mail (More Information Request) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

DUE_DATE	The date by which a response is required, specified in the notification activity.
BODY	The text of the body defined in the message.
SUBJECT	The subject line defined in the message.
COMMENT	Comments added by the sender or the forwarder.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist.
HISTORY	The history of actions on the notification.
QUESTION	Details about what information is being requested.
RESPONSE	The user response section.
NOTIFICATION	Required notification code used to identify the information in the notification.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow Open Mail (More Information Request) message, where attributes preceded by an ampersand (`&`) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE
_____Start of Response Template_____

More Information Template for &NOTIFICATION

User &SENDER has requested more information for the notification.
Please reply to this message, including this response template
with your reply.  Copy and paste from this message if necessary
to obtain an editable copy of the template. Insert your comments
between the quotes against the prompt.

Question: &QUESTION

&RESPONSE
_____End of Response Template_____

Notification Details:
&HEADER
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE><STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE> </HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>

<P>
<P>Question: <B>&QUESTION</B>
<P><B>Please click on the following link to automatically
generate an E-mail response for this question.  Before
sending the E-mail response, ensure desired comments within
quotes.</B>
<P>&MAILTO
<P><B>Notification Details:</B>
<P>&HEADER
<P>&BODY
<BR>
Question: <B>&QUESTION</B>
<P><B>Please click on the following link to automatically
generate an E-mail response for this question.  Before
sending the E-mail response, ensure desired comments within
quotes.</B>
<P>&MAILTO
<P>&HISTORY
</BODY>
</HTML>
```

Orig. Workflow Open Mail (More Information Request) Message:

Oracle Workflow provides the Orig. Workflow Open Mail (More Information Request)
message as an alternative template that you can optionally use as a template to send a

request for more information about a notification from one user to another user. This template does not include the header attributes that are displayed in the Workflow Open Mail (More Information Request) message.

The Orig. Workflow Open Mail (More Information Request) notification template includes generic instructions on how to respond with the requested information. It also includes the following information about a message: the name of the sender of the message, the date that a response is due, and any history of actions on the notification.

The response instructions in the plain text message body describe how to reply manually. This message is used for notifications sent to performers with a notification preference of MAILTEXT or MAILATTH. The response instructions in the HTML-formatted message body describe how to reply using the automatically generated response template. This message is used for notifications sent to performers with a notification preference of MAILHTML or MAILHTM2, and is also attached to notifications sent to performers with a notification preference of MAILATTH.

The Orig. Workflow Open Mail (More Information Request) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

DUE_DATE	The date by which a response is required, specified in the notification activity.
BODY	The text of the body defined in the message.
SUBJECT	The subject line defined in the message.
COMMENT	Comments added by the sender or the forwarder.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist.
HISTORY	The history of actions on the notification.
QUESTION	Details about what information is being requested.
RESPONSE	The user response section.
NOTIFICATION	Required notification code used to identify the information in the notification.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
_____Start of Response Template_____

More Information Template for &NOTIFICATION

User &SENDER has requested more information for the
notification. Please reply to this message, including this
response template with your reply.  Copy and paste from
this message if necessary to obtain an editable copy of the
template. Insert your comments between the quotes against
the prompt.

Question: &QUESTION

&RESPONSE
_____End of Response Template_____

Notification Details:
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD> <TITLE> Oracle Workflow Notification </TITLE> </HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P>
<P>User &SENDER has requested more information for the
following notification.
<P>&BODY
<P>Question: <B>&QUESTION</B>
<P><B>Please click on the following link to automatically
generate an E-mail response for this question.  Before
sending the E-mail response, ensure desired comments within
quotes.</B>
<P>&MAILTO
<P>&HISTORY
</BODY>
</HTML>
```

Workflow More Information Request (Outlook Express) Message:

If you use an e-mail application such as Microsoft Outlook Express as your e-mail
client, you should select the standard Workflow More Information Request (Outlook
Express) message as a template for requests for more information about a notification
from one user to another user, for users with a notification preference of MAILHTML,
MAILHTM2, or MAILATTH. This message includes a link to the Notification Details Web
page to let users respond to the request there. This template is provided to
accommodate e-mail applications that cannot process the response links included in the
Workflow Open Mail (More Information Request) template. In particular, if you select
the Workflow Open Mail (Outlook Express) message template for a notification mailer,
then you should also select the Workflow More Information Request (Outlook Express)
message template for that notification mailer. See: Workflow Open Mail (Outlook
Express Message), page 2-95.

> **Note:** You can use the Open Notification (More Information Request) parameter in the notification mailer configuration wizard to select the appropriate message template. See: Notification Mailer Configuration Wizard, page 7-30.

The response instructions in the plain text message body describe how to reply manually using the templated response method. This message is used for notifications sent to performers with a notification preference of MAILATTH. The HTML-formatted message body includes a link called "Click here to respond" which lets users access the notification in the Notification Details Web page to respond to the request for information. This message is used for notifications sent to performers with a notification preference of MAILHTML or MAILHTM2, and is also attached to notifications sent to performers with a notification preference of MAILATTH.

> **Note:** When users choose the "Click here to respond" link, it automatically attempts to establish a Web session with the Web server. Users must be able to connect to the Web server to use this link to respond to a notification. See: Reviewing Notifications via Electronic Mail, *Oracle Workflow User's Guide*.

The Workflow More Information Request (Outlook Express) message has the following message attributes. The values are drawn from the message definition associated with a notification activity.

DUE_DATE	The date by which a response is required, specified in the notification activity.
BODY	The text of the body defined in the message.
SUBJECT	The subject line defined in the message.
COMMENT	Comments added by the sender or the forwarder.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist.
HISTORY	The history of actions on the notification.
QUESTION	Details about what information is being requested.
RESPONSE	The user response section.
NOTIFICATION	Required notification code used to identify the information in the notification.
HEADER	Standard header attributes and any custom header

attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.

TIMEZONE	The time zone of the dates and times displayed in the notification, based on the time zone of the server from which the notification was sent.
CLICK_HERE_RESPONSE	The content of the HTML tag that a recipient would click on to access the Notification Details page to respond to a request for more information. This attribute is used only for HTML e-mail notifications.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow More Information Request (Outlook Express) message, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification
&TIMEZONE
_____Start of Response Template_____

More Information Template for &NOTIFICATION

User &SENDER has requested more information for the notification.
Please reply to this message, including this response template
with your reply.  Copy and paste from this message if necessary
to obtain an editable copy of the template. Insert your comments
between the quotes against the prompt.

Question: &QUESTION

&RESPONSE
_____End of Response Template_____

Notification Details:
&HEADER
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<HTML> <HEAD>
<TITLE> Oracle Workflow Notification </TITLE>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" >
<P><SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<P>From: &SENDER
<P>Question: <B>&QUESTION</B>
<P><B>Please click on the following link to respond to this
request for more information</B>
<P>&CLICK_HERE_RESPONSE
<P><B>Notification Detail:</B>
<P>&HEADER
<P>&BODY
<BR>
Question: <B>&QUESTION</B>
<P><B>Please click on the following link to respond to this
request for more information</B>
<P>&CLICK_HERE_RESPONSE
<P>&HISTORY
</BODY>
</HTML>
```

Workflow Invalid Open Mail (More Information Request) Message:

The Workflow Invalid Open Mail (More Information Request) message is sent to a user
by default when a user responds incorrectly to a request for more information. For
example, if an answering message from a user contains a valid notification ID (NID)
line matching it with a request for more information about a notification, but does not
contain any response value, the notification mailer sends a Workflow Invalid Open Mail
(More Information Request) message to the user. This message describes how to
respond to the request for information correctly.

The Workflow Invalid Open Mail (More Information Request) message has the
following message attributes. The values are drawn from the message definition
associated with a notification activity.

DUE_DATE	The date by which a response is required, specified in the notification activity.
BODY	The text of the body defined in the message.
SUBJECT	The subject line defined in the message.
COMMENT	Comments added by the sender or the forwarder.
SENDER	The name of the sender of the message, as displayed in the From column in the Worklist.
HISTORY	The history of actions on the notification.

QUESTION	Details about what information is being requested.
RESPONSE	The user response section.
NOTIFICATION	Required notification code used to identify the information in the notification.
HEADER	Standard header attributes and any custom header attributes defined to hold key information for the message. See: Header Attributes, *Oracle Workflow Developer's Guide*.
MAIL_EXP_VALUES	Information about the expected valid response values.
MAIL_VALUE_FOUND	The invalid response value found in the user's response message.
MAIL_ERROR_MESSAGE	An error message that the mail program generates if an error occurs upon processing the response.
TEMPLATE_STYLE	The stylesheet that determines the appearance of the HTML-formatted message body.

You can customize the boilerplate text that appears in the body of the Workflow Invalid Open Mail (More Information Request) message, where attributes preceded by an ampersand (&) are token substituted with runtime values when the notification is sent.

The boilerplate text for a plain text message body is as follows:

```
Oracle Workflow Notification

Warning: Your previous response to this message was invalid
(see error message below).  Please resubmit your response.

Important: Some mail clients, notably early releases of
Microsoft Outlook Express, may not copy the 'NID' line
properly in your response. Please verify that the 'NID'
line is included in full and contains the prefix 'NID' and
all the details between the square brackets when responding.

Error Message: &MAIL_ERROR_MESSAGE

Value Found: &MAIL_VALUE_FOUND

Remarks: &MAIL_EXP_VALUES
_____Start of Response Template_____

More Information Template for &NOTIFICATION

User &SENDER has requested more information for the
notification. Please reply to this message, including
this response template with your reply.  Copy and paste
from this message if necessary to obtain an editable
copy of the template. Insert your comments between the
quotes against the prompt.

Question: &QUESTION

&RESPONSE
_____End of Response Template_____

Notification Details:
&HEADER
&BODY

Due Date: &DUE_DATE

&HISTORY
```

The boilerplate text for an HTML-formatted message body is as follows:

```
<html>
<Head>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</Head>
<body>
<SPAN class="OraTipLabel">&TIMEZONE</SPAN>
<p>Warning: Your previous response to this message was invalid
(see error message below).  Please resubmit your response.
<P>Error Message: &MAIL_ERROR_MESSAGE
<BR>
<BR>Value Found: &MAIL_VALUE_FOUND
<BR>
<BR>Remarks: &MAIL_EXP_VALUES
<HR><P>Question: <B>&QUESTION</B>
<P><B>Please click on the following link to automatically
generate an E-mail response for this question.  Before
sending the E-mail response, ensure desired comments within
quotes.</B>
<P>&MAILTO
<P>&HEADER
<P>&BODY
<P><B>Please click on one of the following choices to
automatically generate an E-mail response.  Before sending
the E-mail response to close this notification, ensure all
response prompts include a desired response value within
quotes.</B>
<P><B>Important:</B> Some mail clients, notably early
releases of Microsoft Outlook Express, may not copy the 'NID'
line properly in your response. Please verify that the 'NID'
line is included in full and contains the prefix 'NID' and
all the details between the square brackets when responding.
<P>Question: <B>&QUESTION</B>
<P><B>Please click on the following link to automatically
generate an E-mail response for this question.  Before
sending the E-mail response, ensure desired comments within
quotes.</B>
<P>&MAILTO
<P>&HISTORY
</BODY>
</HTML>
```

User Notification Preference Update Report Message:

Oracle Workflow sends the User Notification Preference Update Report message to the
SYSADMIN user when a user's notification preference is set to DISABLED. The message
informs the recipient that an e-mail notification could not be sent to one or more
recipients, that the notification preference for those recipients has been set to DISABLED
, and that those recipients' original notification preferences, which are listed, should be
reset after the issues that caused the failures are corrected. See: Outbound Notification
Mailer Processing, page 2-51 and Handling Mailer Errors, page 2-73.

This message has the following message attributes:

NOTIFICATION_ID	The identifier of the original notification for which an

e-mail failure occurred.

ROLE   The role to which the original notification was sent.

UPDATED_USER_REPORT   A list of the recipients of the original notification to whom
                                     e-mail could not be delivered, and of those recipients'
                                     original notification preference settings before their
                                     notification preferences were set to DISABLED.

TEMPLATE_STYLE   The stylesheet that determines the appearance of the
                                     HTML-formatted message body.

The boilerplate text for the plain text message body is as follows:

```
The notification with the ID of &NOTIFICATION_ID experienced
problems when attempting to dispatch an email notification
to the role &ROLE. Subsequently the notification preference
for the following users has been set to DISABLED. Please
correct the issue and re-enable their notification preference.

------------------------------------------------------------------------
--------

&UPDATED_USER_REPORT
```

The boilerplate text for the HTML-formatted message body is as follows:

```
<html>
<head>
<STYLE>
<! -
&TEMPLATE_STYLE
 - >
</STYLE>
</head>
<body>
<p>The notification with the ID of <b>&NOTIFICATION_ID</b>
experienced problems when attempting to dispatch an email
notification to the role <b>&ROLE</b>. Subsequently the
notification preference for the following users has been set
to <b>DISABLED</b>. Please correct the issue and re-enable
their notification preference.
<p>
<hr>
<p>
<span style="font-family:monospace,courier">&UPDATED_USER_REPORT</span>
</body></html>
```

Notification Mailer Override Address Verification Message:

Oracle Workflow sends the Notification Mailer Override Address Verification message
when an administrator sets an override address where the notification mailer should
send all outgoing e-mail notifications for testing purposes. The message requests the
recipient to verify the override address to ensure that the address is accessible and that
its use is authorized. This message must not be customized. See: Reviewing Service
Component Details, page 7-13.

Step 8: Adding Worklist Functions to User Responsibilities

You can optionally give users access to the Advanced Worklist, Personal Worklist, and Notification Search Web pages from any responsibility you choose. To make a page available from a particular responsibility, add the appropriate function to the menu associated with that responsibility. Then you can assign that responsibility to your users. See: Overview of Function Security, *Oracle Applications System Administrator's Guide* and Overview of Menus and Function Security, *Oracle Applications Developer's Guide*.

The following table shows the functions that correspond to each page.

Worklist Functions

Function	User Function Name	Description
WF_WORKLIST	Advanced Workflow Worklist	Advanced Workflow worklist
WF_WORKLIST_CUSTOM	Personal Worklist	Advanced Workflow worklist with options to personalize search and display
WF_WORKLIST_SEARCH	Workflow Notification Search	Notification Search page that allows users to search for their own notifications and administrators to search for all users' notifications

The Advanced Worklist is seeded on the menu for the Workflow User Web Applications responsibility by default. You can also add this function to other responsibilities from which you want users to access notifications.

The Personal Worklist is an optional feature that is not seeded on any Oracle Workflow menu. If you want users to access this version of the Worklist, you must first add the Personal Worklist function to the menu for a responsibility assigned to those users.

If you add the Personal Worklist, you can use worklist flexfields to define specialized worklist views that display information specific to particular types of notifications. If you define a securing function for a view, add the securing function to the same menu as the Personal Worklist function. Your specialized worklist view will appear in the list of views only when users access the Personal Worklist from that responsibility. See: Defining Specialized Worklist Views with Worklist Flexfields, page 6-9.

> **Note:** If you have Oracle Application Framework set up in Oracle JDeveloper for custom development, you can also embed the Personal

Worklist as a region in an Oracle Application Framework page. In this way you can provide users access to the worklist from within your own application. See: Embedding the Personal Worklist in an Oracle Application Framework Page, page 6-24

The Notification Search page is seeded on the menu for the Workflow Administrator Web Applications responsibility by default. You can also add this function to other responsibilities from which you want users to be able to search for notifications. For example, if you want users with the Workflow User Web Applications responsibility to have access to the Notification Search page, you can add this function to the FND_WFUSER (Workflow User) menu with a prompt such as Notification Search.

A user must have workflow administrator privileges to access other users' notifications in the Notification Search page. If a user does not have administrator privileges, that user can only search for and access his or her own notifications. Workflow administrator privileges are assigned in the Workflow Configuration page. See: Setting Global User Preferences, page 2-9.

> **Note:** Ensure that you have assigned workflow administrator privileges only to appropriate users. If the Workflow System Administrator field in the Workflow Configuration page is set to *, granting administrator privileges to all users, then any user with a responsibility that includes the Notification Search page can search for and access other users' notifications.

Related Topics

To View Notifications from the Advanced Worklist, *Oracle Workflow User's Guide*

To View Notifications from the Personal Worklist, *Oracle Workflow User's Guide*

Searching for Users' Notifications, page 6-1

Step 9: Setting the WF: Notification Reassign Mode Profile Option

You can use the WF: Notification Reassign Mode profile option to control which reassign modes are available to users. Oracle Workflow provides the following reassign modes.

- Delegate - This mode lets users give another user authority to respond to a notification on their behalf, while still retaining ownership of the notification themselves. For example, a manager might delegate all vacation scheduling approvals to an assistant.

- Transfer - This mode lets users give another user complete ownership of and responsibility for a notification. For example, users might select this option if they

should not have received a certain notification and they want to send it to the correct recipient or to another recipient for resolution. A transfer may have the effect of changing the approval hierarchy for the notification. For example, a manager might transfer a notification about a certain project to another manager who now owns that project.

You can specify which reassign modes users can select by setting the WF: Notification Reassign Mode profile option to one of the following values.

- Reassign - This setting provides users access to both the Delegate and Transfer reassign modes. With this setting, the Advanced Worklist, the Personal Worklist, and the Response section of the Notification Details page display a Reassign button. Users can select this button to navigate to a Reassign page that lets them choose to either delegate or transfer the notification to another user. The `Reassign` setting is the default value for the WF: Notification Reassign Mode profile option.

- Delegate - This setting provides users access only to the Delegate reassign mode. With this setting, the Advanced Worklist, the Personal Worklist, and the Response section of the Notification Details page display a Delegate button in place of the Reassign button. Users can select the Delegate button to navigate to a Reassign page that only lets them delegate the notification to another user.

- Transfer - This setting provides users access only to the Transfer reassign mode. With this setting, the Advanced Worklist, the Personal Worklist, and the Response section of the Notification Details page display a Transfer button in place of the Reassign button. Users can select the Transfer button to navigate to a Reassign page that only lets them transfer the notification to another user.

You can set the WF: Notification Reassign Mode profile option in the System Profile Values window. This profile option can be set at site, application, responsibility, and user levels. The internal name for this profile option is `FND_NTF_REASSIGN_MODE`.

Related Topics

Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide*

To View Notifications from the Advanced Worklist, *Oracle Workflow User's Guide*

To View Notifications from the Personal Worklist, *Oracle Workflow User's Guide*

To View the Details of a Notification, *Oracle Workflow User's Guide*

To Reassign a Notification to Another User, *Oracle Workflow User's Guide*

Step 10: Setting Up Notification Handling Options

Vacation rules handle notifications automatically when users are not available to manage their notifications personally. These rules are defined according to the item type to which notifications belong. You can control what item types are available for

vacation rules using the WF: Vacation Rule Item Types lookup type and the WF: Vacation Rules - Allow All profile option.

Additionally, one user can grant another user access to his or her worklist, so that the second user can handle notifications on behalf of the first user. Users can choose to grant access only to notifications that belong to selected item types. You can use the WF: Vacation Rule Item Types lookup type to control what item types can be specified for worklist access.

Adding Item Types for Vacation Rules and Worklist Access:

By default, the list of item types a user can select when creating a vacation rule or when granting worklist access displays those item types for which the user has previously received at least one notification. You can also choose to add item types that you want to appear in the lists for all users. In this way you can allow users to create rules or grant worklist access to handle any notifications they may receive from those item types in the future.

To add an item type to the list, define the internal name of the item type as a lookup code for the WF: Vacation Rule Item Types lookup type.

1. Navigate to the Application Object Library Lookups window in the Application Developer responsibility.

2. Query the `WF_RR_ITEM_TYPES` lookup type with the meaning `WF: Vacation Rule Item Types` in the `Application Object Library` application.

3. Define the item type you want as a new lookup code for this lookup type. Ensure that you enter the item type internal name in the Code field exactly as the name is defined in your database. See: Application Utilities Lookups and Application Object Library Lookups, *Oracle Applications online help*.

Allowing Vacation Rules that Apply to All Item Types:

Use the WF: Vacation Rules - Allow All profile option to determine whether the list of item types for vacation rules includes the "All" option. The "All" option lets users create a generic rule that applies to notifications associated with any item type.

Set the profile option to Enabled if you want the "All" option to appear in the list of item types for vacation rules, or to Disabled if you do not want the "All" option to appear. If you choose Disabled, then users must always specify the item type to which a vacation rule applies. The WF: Vacation Rules - Allow All profile option must be set at site level. The default value is Enabled. See: Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide*.

After changing the value of this profile option, you must stop and restart Oracle HTTP Server for the change to take effect.

> **Note:** This profile option does not apply to worklist access options.

When granting worklist access, users always have the option to grant access to notifications belonging to all item types.

Related Topics

Defining Vacation Rules for Users, page 6-4

Vacation Rules, *Oracle Workflow User's Guide*

Worklist Access, *Oracle Workflow User's Guide*

To View Notifications from the Advanced Worklist, *Oracle Workflow User's Guide*

Step 11: Setting Up for Electronic Signatures

Notifications can require that a user's response be signed by a password-based signature or a certificate-based digital signature. Perform the following setup steps to enable users to provide these signatures.

For more information, see: #WF_SIG_POLICY Attribute, *Oracle Workflow Developer's Guide* and Reviewing Electronic Signature Details, page 6-5.

Implementing Password-based Signatures with Single Sign-On:

Oracle Workflow supports password-based signatures for notifications based on Oracle Application Object Library (FND) passwords. If you maintain your directory service based on Oracle Application Object Library users and passwords, no additional setup is required. However, if you have implemented single sign-on functionality for your site through Oracle Internet Directory, and you want to use password-based signatures, you must perform the following steps.

1. Set the Applications SSO Login Types profile option to either Local or Both at user level for all users who need to enter password-based signatures.

2. Ensure that these users have valid passwords defined in Oracle Application Object Library. See: Managing Oracle Applications Security, *Oracle Applications System Administrator's Guide*.

For more information, see: Oracle Single Sign-On Integration, *Oracle Applications System Administrator's Guide - Security*.

Loading Certificates for Digital Signatures:

If a notification requires a certificate-based digital signature, the user must sign the response with a valid X.509 certificate issued by a certificate authority. Before users can sign responses with their certificates, you must load these certificates into your Oracle Applications database using the Workflow Certificate Loader.

When you load a certificate, you must also specify the Oracle Applications user to

whom that certificate is assigned. Oracle Workflow uses this information to validate that the user attempting to sign with a certain certificate is the same user to whom that certificate is assigned.

A user can have more than one certificate assigned to him or her. However, each certificate can only be assigned to one user. Additionally, after you have loaded a certificate for a user, you cannot delete it from the database or assign it to a different user. If a certificate is incorrectly assigned, the user to whom it belongs must revoke it and obtain a new certificate instead.

You can load several certificates at once by listing the information for all the certificates in a data file for the loader. You can also load a single certificate by specifying the certificate information in the command line for the loader.

> **Note:** If your users access Oracle Applications with Microsoft Internet Explorer, ensure that you also set the Browser Signing DLL Location global preference in the Workflow Configuration page. See: To Set Global Preferences for Oracle Workflow, page 2-9.

1. For each certificate, obtain the following information:

    • The Oracle Applications user name of the user to whom the certificate belongs.

    • The personal certificate itself, in the DER encoded binary X.509 format. The certificate should be provided as a file with an extension of `.cer`.

    • The root certificate of the certificate authority that issued the personal certificate, as well as any intermediate certificates required for this type of personal certificate.

    • A URL for each root and intermediate certificate, specifying the location from which the corresponding Certificate Revocation List (CRL) can be downloaded.

        > **Note:** You only need to load the root certificate for a particular certificate authority, and the intermediate certificates for a particular type of certificate, once. If you already loaded the root and intermediate certificates required for a new personal certificate, you can simply load the personal certificate without reloading the others.

2. If you want to load several certificates at once, create a data file for the Workflow Certificate Loader that specifies the location of the certificates to be loaded and the users to whom they belong. The data file should be a text file containing one entry for each root, intermediate, or personal certificate to be loaded.

    All certificate entries in the file must appear in the order of the certification path, beginning with the root certificate for the certificate authority, followed by any

intermediate certificates and then by the personal certificate. However, if the root or intermediate certificates required for a particular personal certificate were loaded previously, you do not need to reload them.

Each certificate entry must be a single line. For a root or intermediate certificate, use the following format:

```
user=CA; domain=CA; filename=<certificate_file>; crl_url=<URL>
```

where *<certificate_file>* is the full path and file name specifying the location of the certificate file, and *<URL>* is the location from which the corresponding Certificate Revocation List (CRL) can be downloaded.

For a personal certificate, use the following format:

```
user=<user_name>; domain=U; filename=<certificate_file>
```

where *<user_name>* is the Oracle Applications user name of the user to whom the certificate belongs, and *<certificate_file>* is the full path and file name specifying the location of the certificate file.

You can also include comments in the data file. Start each comment line with a number sign (#).

The following example shows a sample data file. Note that although the lines may appear to wrap in this document, each certificate entry is a single line in the data file.

```
#Root certificate for certificate authority myCA
user=CA; domain=CA; filename=/certs/myCA.cer;
crl_url=http://myCA.com/myCA.crl
#
#Personal certificate for user BLEWIS
user=BLEWIS; domain=U; filename=/certs/blewis.cer
```

3. To load several certificates at once using a data file, run the Workflow Certificate Loader with the following command:

```
java oracle.apps.fnd.wf.DigitalSignature.loader.CertificateLoader
[-v] <user_name> <password> <connect_string> <data_file>
```

You can optionally specify the −v option to run the Workflow Certificate Loader in verbose mode, displaying additional diagnostic information in the output.

Replace the variables with your parameters as follows:

- *<user_name>* - The user name of your Oracle Applications database account.

- *<password>* - The password for your Oracle Applications database account.

- *<connect_string>* - The connect string for the database, including the host name, TNS port number, and database system identifier (SID) in the following format:

  <host_name>:<port_number>:<database_SID>

- `<data_file>` - The full path and file name specifying the location of the data file you created in the previous step.

For example:

```
java oracle.apps.fnd.wf.DigitalSignature.loader.CertificateLoader
-v apps apps myserv:4105:mySID myCertData.txt
```

4. To load a single certificate without using a data file, run the Workflow Certificate Loader specifying the certificate information in the command line. For a root or intermediate certificate, use the following command:

```
java oracle.apps.fnd.wf.DigitalSignature.loader.CertificateLoader
[-v] -s <user_name> <password> <connect_string> user=CA
domain=CA filename=<certificate_file> crl_url=<URL>
```

For a personal certificate, use the following command:

```
java oracle.apps.fnd.wf.DigitalSignature.loader.CertificateLoader
[-v] -s <user_name> <password> <connect_string> user=<user_name>
domain=U filename=<certificate_file>
```

You can optionally specify the -v option to run the Workflow Certificate Loader in verbose mode, displaying additional diagnostic information in the output.

Replace the variables with your parameters as follows:

- `<user_name>` - The user name of your Oracle Applications database account.

- `<password>` - The password for your Oracle Applications database account.

- `<connect_string>` - The connect string for the database, including the host name, TNS port number, and database system identifier (SID) in the following format:

  `<host_name>:<port_number>:<database_SID>`

- `<user_name>` - The Oracle Applications user name of the user to whom the personal certificate belongs.

- `<certificate_file>` - The full path and file name specifying the location of the certificate file.

- `<URL>` - The location from which the corresponding Certificate Revocation List (CRL) for the root or intermediate certificate can be downloaded.

For example:

```
java oracle.apps.fnd.wf.DigitalSignature.loader.CertificateLoader
-s apps apps myserv:4105:mySID user=BLEWIS domain=U
filename=/certs/blewis.cer
```

> **Note:** You can display a help message describing the usage of the Workflow Certificate Loader by specifying the -h option with the

following command:

```
java
oracle.apps.fnd.wf.DigitalSignature.loader.CertificateL
oader -h
```

Troubleshooting the Workflow Certificate Loader

The following list shows Workflow Certificate Loader error messages and suggested steps to resolve them.

- **No parent certificate found for certificate** - The loader could not locate the root or intermediate certificate that should precede the current certificate in the certificate path. Ensure that either the parent certificate is already loaded to the database, or a valid entry for the parent certificate appears before the entry for the current certificate in the data file.

- **Unable to create certificate object from file** - The data in the certificate file may be corrupted. Check that the certificate is valid by double-clicking the certificate file and viewing its status. Also, ensure that the certificate is stored in the DER encoded binary X.509 format.

- **FND USER does not exist** - The user name specified in a certificate entry in the data file is not defined as an Oracle Applications user. Ensure that the user name is specified as either CA for a certificate authority or a valid Oracle Applications user name for an individual user.

- **Certificate already associated with another user** - The certificate has already been loaded to the database and assigned to a different user. If a certificate is incorrectly assigned, the user to whom it belongs must revoke it and obtain a new certificate instead.

- **Certdatafile not in proper format** - The data file for the loader does not follow the required format. Ensure that the data file contains only certificate entries and comments, each certificate entry is a single line containing the appropriate arguments, and each comment line begins with a number sign (#).

- **The Network Adapter could not establish the connection** - The loader was unable to connect to the database using the specified parameters. Ensure that you specify the correct database user name, password, and connect string when you run the loader.

- **Illegal Argument Exception** - The loader could not process the parameters provided in the run command. Ensure that you specify the loader parameters in the required format.

Step 12: Customizing the Logo on Oracle Workflow's Web Pages

After Oracle HTTP Server is installed and configured for your Oracle Workflow instance, you can customize the company logo that appears in the upper right corner of Oracle Workflow's Web pages.

To Customize Oracle Workflow's Web Pages:

1. Copy or rename your company logo file (in `.gif` format) to `FNDLOGOS.gif`.

2. Move the file to the physical directory that your Web server's `/OA_MEDIA/` virtual directory points to.

> **Note:** The mapping of `/OA_MEDIA/` is completed as part of the Oracle Applications installation and setup steps. For more information, see the installation documentation for your release.

Step 13: Adding Custom Icons to Oracle Workflow

Oracle Workflow Builder looks for icons in the `ICON` subdirectory of the Oracle Workflow area on your PC. The `ICON` subdirectory is defined in the registry of Oracle Workflow Builder. The Oracle Workflow area is typically the `wf` subdirectory within your Oracle home.

Oracle Workflow provides a variety of icons that you can use to represent your activities and processes. You can add custom icon files to this area as long as they are Windows icon files with the `.ico` suffix.

If you want the custom icons that you include in your Oracle Workflow Builder process definition to appear in the Workflow Monitor when you view the process diagram, perform the following steps.

1. Convert the custom icon files (`.ico`) to *gif* format (`.gif`).

2. Copy the `.gif` files to the physical directory that your Web server's `/OA_MEDIA/` virtual directory points to, so that the Workflow Monitor can access them.

> **Note:** The mapping of `/OA_MEDIA/` is completed as part of the Oracle Applications installation and setup steps. For more information, see the installation documentation for your release.

Step 14: Setting Up the Business Event System

The Business Event System is an application service delivered with Oracle Workflow

that uses Oracle Advanced Queuing (AQ) to communicate business events between systems. You need to perform this step to use event processing. See: Overview of the Oracle Workflow Business Event System, *Oracle Workflow API Reference* and Managing Business Events, *Oracle Workflow Developer's Guide*.

To set up the Business Event System and enable message propagation, perform the following steps:

1. If you want to communicate business events between the local system and external systems, create database links to those external systems, page 2-147.

2. If you want to use custom queues for propagating events, set up your queues, page 2-147.

3. Check the Business Event System setup parameters, page 2-151.

4. Schedule listeners for local inbound agents, page 2-151.

5. Schedule propagation for local outbound agents, page 2-154.

6. Synchronize event and subscription license statuses with product license statuses, page 2-154.

7. Ensure that the WF_CONTROL queue is periodically cleaned up to remove inactive subscribers, page 2-155.

8. You can optionally change the maximum cache size used in Business Event System subscription processing, page 2-157.

9. You can optionally enable static function calls for custom PL/SQL functions, page 2-157.

10. If you want to invoke business process execution language (BPEL) processes through event subscriptions, you can optionally specify the BPEL server, page 2-159.

You should recheck your setup whenever you make changes to your agents that affect the physical implementation required for propagation. See: Agents, *Oracle Workflow Developer's Guide*.

> **Note:** Oracle Workflow sets the status of the local system to Enabled by default. After you finish setting up the Business Event System, you can update your global workflow preferences to set the system status that you want for event processing. See: Setting Global User Preferences, page 2-9.

Oracle Workflow provides scripts to help diagnose Business Event System processing and retry failed events. See: Handling Business Event System Errors, page 2-160.

Creating Database Links:

To propagate event messages between systems, you must create database links from your local system to the remote systems. You should fully qualify the database link name with the domain name.

You can either create database links manually, or use Oracle Enterprise Manager to perform this step. Oracle Enterprise Manager allows workflow administrators to quickly and easily create and administer database links, queue tables, queues, and queue propagation without requiring knowledge of the SQL DDL commands. For more information, see the Oracle Enterprise Manager online help.

You can use the following syntax to create a database link manually:

```
CREATE DATABASE LINK <database link name> CONNECT TO
   <user> IDENTIFIED BY <password> USING '<connect string>';
```

For example:

```
CREATE DATABASE LINK wf10g.mycompany.com CONNECT TO
   wfuser IDENTIFIED BY welcome
   USING 'wf10g';
```

If you have multiple installations of Oracle Workflow on both the local database and the remote database, and you want to use the same username and password to access both systems, you can omit the `<user> IDENTIFIED BY <password>` clause. In this case, the database link uses the username and password of the user who is connected to the database.

```
CREATE DATABASE LINK <database link name> CONNECT TO
   USING '<connect string>';
```

If you want to create a public database link available to all users, specify the parameter `PUBLIC`.

```
CREATE PUBLIC DATABASE LINK <database link name> CONNECT TO
   <user> IDENTIFIED BY <password>
   USING '<connect string>';
```

To verify the names of your database links, use the following syntax:

```
SELECT db_link FROM all_db_links;
```

See: CREATE DATABASE LINK, *Oracle SQL Reference.*

Setting Up Queues:

The Business Event System uses Oracle Advanced Queuing (AQ) to communicate event messages between systems. You must associate a queue with each agent on a Workflow-enabled system that you define in the Event Manager.

When you install Oracle Workflow, several standard queues are created automatically for the standard Workflow agents. These queues all use either the standard WF_EVENT_T structure or JMS Text messages as their payload type. See: Standard Agents, *Oracle Workflow Developer's Guide*, Event Message Structure, *Oracle Workflow API Reference*, and Mapping Between WF_EVENT_T and

SYS.AQ$_JMS_TEXT_MESSAGE, *Oracle Workflow API Reference*.

The following table lists the standard queues.

Business Event System Queues

Queue Table	Queue Name	Payload Type	Retention Time	Description
WF_CONTROL	WF_CONTROL	SYS.AQ$_JMS_TEXT_MESSAGE	1 day	Oracle Workflow internal queue, not for customer use
WF_DEFERRED	WF_DEFERRED	WF_EVENT_T	1 day	Standard queue for deferred subscription processing in the database
WF_ERROR	WF_ERROR	WF_EVENT_T	0 days	Standard queue for error handling in the database
WF_IN	WF_IN	WF_EVENT_T	7 days	Default inbound queue
WF_JAVA_DEFERRED	WF_JAVA_DEFERRED	SYS.AQ$_JMS_TEXT_MESSAGE	1 day	Standard queue for deferred subscription processing in the middle tier
WF_JAVA_ERROR	WF_JAVA_ERROR	SYS.AQ$_JMS_TEXT_MESSAGE	0 days	Standard queue for error handling in the middle tier
WF_JMS_IN	WF_JMS_IN	SYS.AQ$_JMS_TEXT_MESSAGE	7 days	Default inbound queue for JMS Text messages
WF_JMS_OUT	WF_JMS_OUT	SYS.AQ$_JMS_TEXT_MESSAGE	7 days	Default outbound queue for JMS Text messages

Queue Table	Queue Name	Payload Type	Retention Time	Description
WF_NOTIFICATION_IN	WF_NOTIFICATION_IN	SYS.AQ$_JMS_TEXT_MESSAGE	1 day	Standard inbound queue for e-mail notification responses
WF_NOTIFICATION_OUT	WF_NOTIFICATION_OUT	SYS.AQ$_JMS_TEXT_MESSAGE	1 day	Standard outbound queue for e-mail notifications
WF_OUT	WF_OUT	WF_EVENT_T	7 days	Default outbound queue
WF_WS_JMS_IN	WF_WS_JMS_IN	SYS.AQ$_JMS_TEXT_MESSAGE	7 days	Default inbound queue for Web service messages
WF_WS_JMS_OUT	WF_WS_JMS_OUT	SYS.AQ$_JMS_TEXT_MESSAGE	7 days	Default outbound queue for Web service messages

> **Note:** Oracle Workflow also includes three queues named
> WF_REPLAY_IN, WF_REPLAY_OUT, and WF_SMTP_O_1_QUEUE,
> which are not currently used. Oracle XML Gateway provides additional
> standard queues.
>
> Oracle Workflow includes three queues for background engine
> processing named WF_DEFERRED_QUEUE_M,
> WF_OUTBOUND_QUEUE, and WF_INBOUND_QUEUE. These
> queues are separate from the Business Event System queues. See:
> Setting Up Background Workflow Engines, page 2-39.

If necessary, you can change the default retention time set for consumed messages on
the standard Workflow queues, using the PL/SQL procedure
DBMS_AQADM.Alter_Queue. You must not change any other part of the setup of these
queues.

You can also set up your own queues for event message propagation. You can either set
up queues manually, or use Oracle Enterprise Manager to perform this step. Oracle
Enterprise Manager allows workflow administrators to quickly and easily create and
administer database links, queue tables, queues, and queue propagation without

requiring knowledge of the SQL DDL commands. For more information, see the Oracle Enterprise Manager online help.

To set up a queue manually, you must create the queue table, create the queue, and start the queue. You should perform these tasks using a schema that is appropriate for the application for which you will use the queue. You must specify the schema that owns the queue as part of the queue name when you assign the queue to a Business Event System agent.

- To create a queue table, use the PL/SQL procedure DBMS_AQADM.Create_Queue_Table. Use the following syntax:

```
DBMS_AQADM.Create_Queue_Table (
   queue_table => '<queue table name>',
   queue_payload_type => '<queue payload type>',
   sort_list => 'PRIORITY,ENQ_TIME',
   multiple_consumers => TRUE
   compatible => '8.1');
```

For queues that should use the standard Workflow format, specify the queue payload type as WF_EVENT_T. These queues can use the standard queue handler provided with Oracle Workflow, WF_EVENT_QH. For queues that should use the JMS Text message format, specify the queue payload as $AQ_JMS_TEXT_MESSAGE. These queues can use the standard JMS queue handler provided with Oracle Workflow, WF_EVENT_OJMSTEXT_QH. If you define a queue with a different payload type, you must create a queue handler to translate between the standard Workflow format and the format required by the queue. See: Standard APIs for a Queue Handler, *Oracle Workflow Developer's Guide*.

You can also use the storage_clause parameter to specify the tablespace where you want to create the queue table. You may want to specify a tablespace if you expect a queue to be very large.

- To create a queue, use the PL/SQL procedure DBMS_AQADM.Create_Queue. Use the following syntax:

```
DBMS_AQADM.Create_Queue (
   queue_name => '<queue name>',
   queue_table => '<queue table name>');
```

> **Note:** If you want other database users to enqueue messages onto or dequeue messages from your queue, you must grant those users the appropriate privileges using the PL/SQL procedure DBMS_AQADM.Grant_Queue_Privilege.

- To start a queue, use the PL/SQL procedure DBMS_AQADM.Start_Queue. Use the following syntax:

```
DBMS_AQADM.Start_Queue (
   queue_name => '<queue name>');
```

Oracle Workflow provides a sample script called wfevquc2.sql which you can

modify to set up your queues, as well as a sample script called `wfevqued.sql` which you can modify to drop queues. These scripts are located on your server in the `$FND_TOP/sql` directory.

You can verify that your queues are set up properly using the Oracle Workflow Manager component of Oracle Applications Manager. See: Agents, page 7-100.

You can also run a diagnostic test to verify your queues. See: Oracle Workflow Diagnostic Tests, page E-1.

See: Oracle Streams AQ Administrative Interface, *Oracle Streams Advanced Queuing User's Guide and Reference* and DBMS_AQADM, *Oracle Database PL/SQL Packages and Types Reference*

Checking the Business Event System Setup:

Use the Oracle Workflow Manager component of Oracle Applications Manager to verify that the required parameters and components have been set up to enable message propagation for the Business Event System, including the required `JOB_QUEUE_PROCESSES` database initialization parameter. See: Oracle Workflow Manager Overview, page 7-1.

You can either modify the parameter in the `init.ora` file and restart the database, or you can use the `ALTER SYSTEM` statement to dynamically modify the parameter for the duration of the instance.

The `JOB_QUEUE_PROCESSES` parameter defines the number of SNP job queue processes for your instance. Oracle Workflow requires job queue processes to handle propagation of Business Event System event messages by AQ queues. You must start at least one job queue process to enable message propagation. The minimum recommended number of processes for Oracle Workflow is ten.

> **Note:** If you want to review more detailed information about AQ processing, you can optionally use another initialization parameter, `EVENT`, for detailed database level tracing of issues related to AQ. Add the following line to your `init.ora` file:
>
> ```
> event = "24040 trace name context forever, level 10"
> ```
>
> Then restart your database to make this change effective. Be aware that using this parameter may generate large trace files.

See: *Oracle Streams Advanced Queuing User's Guide and Reference*.

Scheduling Listeners for Local Inbound Agents:

To communicate events between different agents, you must schedule listeners for the inbound agents on your local system. The Business Event System requires listeners to be scheduled to receive inbound event messages. Run PL/SQL agent listeners to process event subscriptions with a PL/SQL rule function in the database, and run Java agent listeners to process event subscriptions in the middle tier.

When you schedule a listener for an agent, it monitors the agent's queue, dequeuing any inbound event messages. When an event message is received, the Event Manager searches for and executes any enabled subscriptions by the local system to that event with a source type of External, and also any enabled subscriptions by the local system to the Any event with a source type of External. The listener exits after all event messages on the agent's queue have been dequeued.

The PL/SQL and Java agent listener programs are defined as service component types in the Generic Service Component Framework. This framework helps to simplify and automate the management of background Java services.

You can use Oracle Workflow Manager to submit and manage agent listener service components. You can also view the distribution of event messages on different agents, drill down to view details about individual event messages, and review queue details for the agents. Oracle Workflow Manager is available as a component of Oracle Applications Manager. See: Oracle Workflow Manager Overview, page 7-1.

Oracle Workflow also provides an administrative script named `wfagtlst.sql` that you can use to run a PL/SQL agent listener. See Wfagtlst.sql, page 9-5.

Oracle Workflow provides seeded PL/SQL agent listener service components for the standard WF_DEFERRED, WF_ERROR, and WF_NOTIFICATION_IN agents. These agent listeners are named Workflow Deferred Agent Listener, Workflow Deferred Notification Agent Listener, Workflow Error Agent Listener, and Workflow Inbound Notifications Agent Listener, and they support deferred subscription processing in the database, dedicated deferred subscription processing for notification messages, error handling for the Business Event System in the database, and inbound e-mail processing for notification mailers, respectively.

Oracle Workflow also provides seeded Java agent listener service components for the standard WF_JAVA_DEFERRED, WF_JAVA_ERROR, and WF_WS_JMS_IN agents. These agent listeners are named Workflow Java Deferred Agent Listener, Workflow Java Error Agent Listener, and Web Services IN Agent, and they support deferred subscription processing in the middle tier, error handling for the Business Event System in the middle tier, and inbound Web service message processing, respectively.

Additionally, Oracle XML Gateway provides two seeded PL/SQL agent listener service components for the standard ECX_INBOUND and ECX_TRANSACTION agents. These agent listeners are named ECX Inbound Agent Listener and ECX Transaction Agent Listener, respectively. For more information, see Monitor Workflow Processes, *Oracle XML Gateway User's Guide*.

You can also optionally create additional agent listener service components. For example, you can configure agent listeners for other inbound agents that you want to use for event message propagation, such as the standard WF_IN and WF_JMS_IN agents, or any custom agents. You can also configure an agent listener service component that only processes messages on a particular agent that are instances of a specific event.

Service components must be hosted by a service component container. If you create custom agent listener service components, you can assign them to the seeded container

for agent listeners.

A service component container is implemented as a Generic Service Management (GSM) service. The seeded container for agent listeners is named `Workflow Agent Listener Service`.

Based on the volume to be handled by the seeded container, you can also choose to create your own custom containers as GSM services in Oracle Applications Manager. If you create a custom GSM service in OAM, you can copy the service parameters from the seeded `Workflow Agent Listener Service` to your new service in order to specify how to run the new service.

Before agent listener service components can run, the container which manages them must first be started. In order to run the seeded agent listeners, you should ensure that the `Workflow Agent Listener Service` container is running using Oracle Applications Manager. If you create your own custom containers in OAM for custom service components, ensure that those containers are running as well.

> **Note:** You can run a diagnostic test to verify the GSM services for Oracle Workflow. See: Oracle Workflow Diagnostic Tests, page E-1.
>
> Additionally, if the status of an agent listener service component changes to `Stopped with Error` or `System Deactivated`, Oracle Workflow posts a system alert to the System Alerts and Metrics page in Oracle Applications Manager. See: System Alerts, Metrics, and Logs, *Oracle Applications System Administrator's Guide - Maintenance*.

See: Agents, *Oracle Workflow Developer's Guide* and Listen, *Oracle Workflow API Reference*.

Exception Handling for Inbound Queues

Oracle Streams Advanced Queuing (AQ) transfers event messages from an agent's normal queue to the associated exception queue in the following cases:

- A message expires before being dequeued.

- An agent listener repeatedly attempts to process a message but rolls back the transaction due to an error, and the number of attempts exceeds the queue's retry limit.

You can run the Move Messages from Exception to Normal Queue of Workflow Agent concurrent program (`FNDWF_MOVE_MSGS_EXCEP2NORMAL`) to transfer such messages back to the agent's normal queue. This program helps enable mass reprocessing in case of a large number of errors. Use Standard Request Submission to run the program, specifying the inbound agent for which you want to transfer messages back to the normal queue. After the program completes, run the appropriate agent listener to reattempt normal processing for these event messages. See: Running Reports and Programs, *Oracle Applications User's Guide* and Exception Handling, *Oracle Streams Advanced Queuing User's Guide and Reference*.

Scheduling Propagation for Local Outbound Agents:

To communicate events between different agents, you must schedule propagation for the outbound agents on your local system. The Business Event System requires propagation to be scheduled to send outbound event messages.

When you send an event message to an agent, the Event Manager places the message on the queue associated with the outbound agent. The message is then asynchronously delivered to the inbound agent by propagation.

You can schedule AQ propagation for agents that use the SQLNET protocol by the following methods:

• Use the Distributed Database Management feature to manage AQ through Oracle Enterprise Manager. See: Oracle Enterprise Manager Support, *Oracle Streams Advanced Queuing User's Guide and Reference* and the Oracle Enterprise Manager online help.

• Run the DBMS_AQADM.Schedule_Propagation API in SQL*Plus. See: DBMS_AQADM, *Oracle Database PL/SQL Packages and Types Reference*.

If you want to use the standard WF_OUT and WF_JMS_OUT agents or custom agents for event message propagation, ensure that you schedule propagation for those agents. You do not need to schedule propagation for the WF_CONTROL, WF_NOTIFICATION_OUT, or WF_WS_JMS_OUT agents, however. The middle tier processes that use WF_CONTROL dequeue messages directly from its queue, and notification mailers send messages placed on the WF_NOTIFICATION_OUT queue. For WF_WS_JMS_OUT, you can optionally start a Web services outbound component named Web Services OUT Agent, provided by Oracle Workflow.

For agents that use protocols other than the SQLNET protocol, you must provide external propagation logic. See: Agents, *Oracle Workflow Developer's Guide*.

You can use Oracle Workflow Manager to review the propagation schedules for your local outbound agents. You can also view the distribution of event messages on different agents, drill down to view details about individual event messages, and review queue details for the agents. Oracle Workflow Manager is available as a component of Oracle Applications Manager. See: Oracle Workflow Manager Overview, page 7-1.

See: Agents, *Oracle Workflow Developer's Guide*.

Synchronizing License Statuses:

Some Oracle Applications products provide seeded events and subscriptions. In these cases, Oracle Workflow executes subscriptions only if the triggering event and the subscription are both owned by products that you have licensed with a status of Installed or Shared.

You can use the License Manager utility to review which products you currently have licensed. See: 'License Manager, *Oracle Applications System Administrators Guide -*

Maintenance.

To ensure that the license status of the seeded events and subscriptions in the Business Event System is updated according to the status of the products you currently have licensed, you can run the Synchronize Product License and Workflow BES License concurrent program. Use the Submit Requests form in Oracle Applications to submit this concurrent program.

If you upgrade from an Oracle Applications release earlier than Release 11.5.9, you should run the Synchronize Product License and Workflow BES License concurrent program once after the upgrade to update the license status of the existing events and subscriptions in your Event Manager. Otherwise, subscriptions may not be correctly processed after the upgrade. Subsequently, when you license a product, Oracle Workflow automatically updates the license status for all the events and subscriptions owned by that product.

> **Note:** Any events and subscriptions that you define with a customization level of User are always treated as being licensed.

See: Events, *Oracle Workflow Developer's Guide* and Event Subscriptions, *Oracle Workflow Developer's Guide*.

To submit the Synchronize Product License and Workflow BES License concurrent program

1. Navigate to the Submit Requests form in Oracle Applications to submit the Synchronize Product License and Workflow BES License concurrent program. When you install and set up Oracle Applications and Oracle Workflow, your system administrator needs to add this concurrent program to a request security group for the responsibility that you want to run this program from. The executable name for this concurrent program is "Synchronize Product License and Workflow BES License" and its short name is FNDWFLIC. See: Overview of Concurrent Programs and Requests, *Oracle Applications System Administrator's Guide*.

2. Select the Synchronize Product License and Workflow BES License concurrent program as the request to run. This program does not require any parameters. See: Running Reports and Programs, *Oracle Applications User's Guide*.

3. When you finish modifying the print and run options to define the schedule for this request, choose Submit to submit the request.

Cleaning Up the Workflow Control Queue:

Oracle Workflow contains a standard Business Event System agent named WF_CONTROL, which is associated with a standard queue that is also named WF_CONTROL. This queue has a payload type of JMS Text message. The WF_CONTROL agent is used for internal processing only, and is not meant for customer use. You should not place custom event messages on this queue.

The Generic Service Component Framework uses WF_CONTROL to handle control events for containers and service components, such as notification mailer or agent listener service components. WF_CONTROL is also used for other Oracle Applications internal processing.

You do not need to schedule propagation for the WF_CONTROL agent, because the middle tier processes that use WF_CONTROL dequeue messages directly from its queue.

However, the subscribers to the WF_CONTROL queue need to be cleaned up periodically. A concurrent program named Workflow Control Queue Cleanup is automatically scheduled to perform this cleanup for you

When a middle tier process for Oracle Applications starts up, it creates a JMS subscriber to the queue. Then, when an event message is placed on the queue, a copy of the event message is created for each subscriber to the queue. If a middle tier process dies, however, the corresponding subscriber remains in the database. For more efficient processing, you should ensure that WF_CONTROL is periodically cleaned up by removing the subscribers for any middle tier processes that are no longer active.

The *WF_BES_CLEANUP.Cleanup_Subscribers()* procedure sends an event named `oracle.apps.wf.bes.control.ping` to check the status of each subscriber to the WF_CONTROL queue. If the corresponding middle tier process is still alive, it sends back a response. The next time the cleanup procedure runs, it checks whether responses have been received for each ping event sent during the previous run. If no response was received from a particular subscriber, that subscriber is removed. See: Cleanup_Subscribers, *Oracle Workflow API Reference*.

The recommended frequency for performing cleanup is every twelve hours. In order to allow enough time for subscribers to respond to the ping event, the minimum wait time between two cleanup runs is thirty minutes. If you run the procedure again less than thirty minutes after the previous run, it will not perform any processing.

Control Queue Cleanup

Oracle Workflow provides a concurrent program named Workflow Control Queue Cleanup, which uses the *WF_BES_CLEANUP.Cleanup_Subscribers()* API to perform the necessary cleanup. This concurrent program is scheduled to run every twelve hours by default, which is the recommended frequency for performing cleanup. You can optionally run this program with a different schedule if you want to perform cleanup at a different frequency.

To submit the Workflow Control Queue Cleanup concurrent program

1. Navigate to the Submit Requests form in Oracle Applications to submit the Workflow Control Queue Cleanup concurrent program. When you install and set up Oracle Applications and Oracle Workflow, your system administrator needs to add this concurrent program to a request security group for the responsibility that you want to run this program from. The executable name for this concurrent program is "Workflow Control Queue Cleanup" and its short name is

. See: Overview of Concurrent Programs and Requests, *Oracle Applications System Administrator's Guide*.

2. Select the Workflow Control Queue Cleanup concurrent program as the request to run. This program does not require any parameters. See: Running Reports and Programs, *Oracle Applications User's Guide*.

3. When you finish modifying the print and run options to define the schedule for this request, choose Submit to submit the request.

You can run a diagnostic test to check that the Workflow control queue is properly accessible. See: Oracle Workflow Diagnostic Tests, page E-1.

See: Business Event System Control Events, *Oracle Workflow Developer's Guide*, Standard Agents, *Oracle Workflow Developer's Guide*, and Business Event System Cleanup API, *Oracle Workflow API Reference*.

Changing the Maximum Cache Size for the Business Event System:

The Business Event System caches event, subscription, and agent definitions to enhance performance during subscription processing. The default maximum size of the cache is 50 records. You can optionally increase the maximum cache size to reduce the database queries performed by the Business Event System, or decrease the maximum cache size to reduce the amount of memory used by the cache.

The maximum cache size is determined by the value for the `WFBES_MAX_CACHE_SIZE` resource token. To change this value, you must first create a `.msg` source file specifying the new size as the value for the `WFBES_MAX_CACHE_SIZE` resource token. Then use the Workflow Resource Generator program to upload the new seed data from the source file to the database table WF_RESOURCES. See: To Run the Workflow Resource Generator, *Oracle Workflow API Reference*.

Enabling Static Function Calls for Custom PL/SQL Functions:

If you use custom PL/SQL functions within the Business Event System, including event data generate functions, event subscription rule functions, and queue handler enqueue and dequeue APIs, Oracle Workflow calls those functions using dynamic SQL by default. However, you can enable Oracle Workflow to call your custom functions statically to enhance performance.

Oracle Workflow provides two PL/SQL packages with procedures that contain lists of static function calls. The Business Event System internally calls these procedures during subscription processing to check whether static function calls are available for the procedures being executed.

*   WF_BES_DYN_FUNCS package - Contains static function calls for generate functions and rule functions.

*   WF_AGT_DYN_FUNCS package - Contains static function calls for queue handler

enqueue and dequeue APIs.

The initial seeded versions of these packages include static function calls only for seeded Oracle Workflow functions, such as the rule function WF_RULE.Default_Rule and the queue handler APIs WF_EVENT_QH.Enqueue and WF_EVENT_QH.Dequeue. You can use the `wfbesfngen.sql` script to add functions from other Oracle Applications products or your own custom functions to these packages.

1. Run the `wfbesfngen.sql` script as follows:

   ```
   sqlplus <user/pwd> @wfbesfngen <type> <name>
   ```

   Replace the variables with your parameters as follows:

   - `<type>` - The type of functions you want to add.

     - `EVENTS` - Add generate functions and rule functions to the WF_BES_DYN_FUNCS package body.

     - `AGENTS` - Add queue handler enqueue and dequeue APIs to the WF_AGT_DYN_FUNCS package body.

   - `<name>` - Specify the names of the events or agents with which the functions you want to add are associated. You can specify one or more object names separated by commas. Do not include any spaces in the name string. For the `EVENTS` type, you can either specify a full event name for an individual event, or specify a correlation ID consisting of a partial event name with the wildcard character (`%`) to include functions associated with any event whose name begins with the specified value.

     > **Note:** The script's output is not cumulative; it creates a separate new package body each time you run it. If you rerun the script, ensure that you specify the event or agent names for all functions that you want to include, repeating any names specified in previous runs that are still applicable, as well as adding any new names.

   For example:

   ```
   sqlplus apps/apps @wfbesfngen EVENTS oracle.apps.ap%,oracle.apps.ar%
   ```

   or:

   ```
   sqlplus apps/apps @wfbesfngen AGENTS WF_IN,WF_OUT,WF_CONTROL
   ```

2. The script generates a file containing a new package body in the directory specified in the `UTL_FILE_DIR` database initialization parameter.

   - For the `EVENTS` type, the script generates a new package body for the WF_BES_DYN_FUNCS package in a file named `WFBESDFNB<timestamp>`

```
.pls.
```

- For the `AGENTS` type, the script generates a new package body for the WF_AGT_DYN_FUNCS package in a file named `WFAGTDFNB<timestamp>` `.pls`.

Review the file to verify that the package body was generated successfully. The header of the file lists the object names for whose associated functions the package body contains static function calls. Ensure that you keep backup copies of the files containing your customized package bodies.

3. Compile the new package body in your database. You should perform this step during a patching window or maintenance downtime when there is no Business Event System activity and no agent listeners are running.

    If the new package body does not compile successfully, you can edit the generated file manually to correct the package body. Alternatively, if the generated package body does not compile, you can revert back to the corresponding original package body provided by Oracle Workflow in the following files:

    - `$FND_TOP/patch/115/sql/WFBESDFNB.pls` - Contains the seeded WF_BES_DYN_FUNCS package body.

    - `$FND_TOP/patch/115/sql/WFAGTDFNB.pls` - Contains the seeded WF_AGT_DYN_FUNCS package body.

If you drop a custom function from your database, you should edit the file containing the corresponding customized package body to comment out the static function call for that function, and then recompile the package body.

> **Note:** If additional standard functions become available in the future, Oracle Workflow may ship updated versions of the seeded package bodies. In this case, you may need to restore the static function calls for your custom functions. For example, you can rerun the `wfbesfngen.sql` script to add your custom functions to the updated seeded package bodies, or recompile your customized package bodies from your backup files. Check the installation instructions when applying Oracle Workflow patches to determine whether you need to take steps to re-enable the static function calls for your custom functions.

Specifying the BPEL Server:

If you define event subscriptions that invoke business process execution language (BPEL) processes, you can optionally set the WF: BPEL Server profile option to specify the host name and port of the BPEL server. When this profile option is set, you can use relative URLs rather than absolute URLs in the subscription parameters to specify the

location of the Web Services Description Language (WSDL) descriptions for the BPEL processes.

At runtime, Oracle Workflow replaces the `bpel://` prefix in the relative URLs with the host name and port for the BPEL server specified in the WF: BPEL Server profile option. In this way, the profile option lets you quickly change the BPEL server setup for all subscriptions that invoke BPEL processes, such as if you move these subscription definitions from a test instance to a production instance. See: Invoking a Web Service, *Oracle Workflow Developer's Guide*.

Set the WF: BPEL Server profile option to the host name and port of the BPEL server using the following format:

`http://<host_name>:<port>/`

> **Note:** The profile option value must end with a slash (/) in order to form the final WSDL description URL correctly.

This profile option must be set at site level. See: Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide*.

Handling Business Event System Errors:

To check the status of a particular event or help investigate errors, you can run a script named `wfbesdbg.sql` that displays debugging information. You can also obtain this information by running a diagnostic test through Oracle Diagnostics. See: wfbesdbg.sql, page 9-5 and Oracle Workflow Diagnostic Tests, page E-1.

In case of a large number of errored events, Oracle Workflow provides special scripts for mass event reprocessing. Do not run these scripts unless you are directed to do so by Oracle Support.

The following scripts are located in the `$FND_TOP/sql` directory.

- `wfevrtry.sql` - This script lets you abort or retry event subscription processing for all locally raised instances of a particular event that are enqueued on the WF_ERROR or WF_JAVA_ERROR queues and that have not yet been processed by an agent listener for these queues.

  Use the script as follows:

  ```
  sqlplus <user/pwd> @wfevrtry <APPSuser> <APPSpwd> <FND_schema>
  <event_name>
  ```

  Replace *<APPSuser>* and *<APPSpwd>* with the user name and password for the APPS user. The user name is usually `apps`. Replace *<FND_schema>* with the ORACLE username that connects to Oracle Application Object Library data in Oracle Applications, usually `applsys`. Replace *<event_name>* with the internal name of the event whose instances have errors.

  The script prompts you to select one of these actions:

- Abort - Abort subscription processing by dequeuing each event instance without performing any further processing.

- Retry with Key - Reraise each event instance with only the event name and event key.

- Retry with Key & Data - Reraise each event instance with only the event name, event key, and event data.

- Retry with Key, Data, & Parameters - Reraise each event instance with the event name, event key, event data, and parameters.

You can choose the level of information to provide to the Event Manager when reraising the event instances. For example, if an error exists in the event data that was originally provided, the event instances can be reraised with only the event name and the event key, forcing the Event Manager to regenerate the event data using the event's generate function. You can also attempt to correct the error before reraising the event instances.

- `wfntfrsp.sql` - This script lets you abort or retry event subscription processing for all instances of a particular event for which the SYSADMIN user has an open Local Event Error notification from the Default Event Error process. The script responds to the notifications with the option you specify.

  Use the script as follows:

  ```
  sqlplus <user/pwd> @wfntfrsp <event_name>
  ```

  Replace `<event_name>` with the internal name of the event whose instances have errors.

  The script prompts you to select one of these actions:

  - Abort - Abort subscription processing by closing the error notification without performing any further processing.

  - Retry with Key - Reraise each event instance with only the event name and event key.

  - Retry with Key & Data - Reraise each event instance with only the event name, event key, and event data.

  - Retry with Key, Data, & Parameters - Reraise each event instance with the event name, event key, event data, and parameters.

  You can choose the level of information to provide to the Event Manager when reraising the event instances. For example, if an error exists in the event data that was originally provided, the event instances can be reraised with only the event name and the event key, forcing the Event Manager to regenerate the event data using the event's generate function. You can also attempt to correct the error before

reraising the event instances.

3

Oracle Workflow Security

This chapter describes the architecture and configuration of security for Oracle Workflow.

This chapter covers the following topics:

- Oracle Workflow Security

Oracle Workflow Security

The ability to control user access to Web and application content and to protect your site against people breaking into your system is critical. This chapter describes the architecture and configuration of security for Oracle Workflow.

For additional information about security, refer to the following documents:

- The Overview of Oracle Applications Security, *Oracle Applications System Administrator's Guide* provides information about Oracle Applications security.

- The *Oracle Database Security Overview* provides information about Oracle Database security.

- The *Oracle Application Server Security Guide* provides an overview of Oracle Application Server security and its core functionality.

- The *Oracle Identity Management Concepts and Deployment Planning Guide* provides guidance for administrators of the Oracle security infrastructure.

About Oracle Workflow Security

This section describes the Oracle Workflow security model.

Oracle Workflow Security Model

Oracle Workflow uses a password-based security model to protect Web and application

content. Oracle Workflow is part of the Oracle Applications security model in which users' privileges and access to functionality are based on responsibilities.

Classes of Users and Their Privileges

For purposes of accessing Oracle Workflow Web pages, Oracle Workflow defines two classes of users: Workflow administrators and Workflow users.

Access to Oracle Workflow administrator features is controlled both by responsibilities, which determine what pages a user can access, and by the workflow administrator role defined in the Workflow Configuration page, which determines what administrative operations a user can perform. To perform administrative operations, users must both have a responsibility that includes Oracle Workflow administrator Web pages and be associated with the workflow administrator role. If users have an appropriate responsibility but are not associated with the workflow administrator role, then they can only view the administrator Web pages. Users must also have an appropriate responsibility to access the Oracle Workflow self-service user Web pages. In some cases users who are associated with the workflow administrator role can perform additional administrative operations in the self-service Web pages as well.

- Workflow administrators - Oracle Applications users who have an Oracle Workflow administrator responsibility and are associated with the workflow administrator role, can:

  - Access the Oracle Workflow Administrator Home page.

  - Set global workflow preferences and their own individual user preferences.

  - View and respond to any user's notifications through the Worklist pages.

  - Define rules to handle notifications automatically in a user's absence.

  - View and update any user's processes in the administrator Status Monitor.

  - View Workflow item type definitions and launch test processes.

  - Manage business events and event subscriptions.

  - Define worklist flexfields rules and perform rule simulations. Access to worklist flexfields rules functionality requires only an Oracle Workflow administrator responsibility.

- Workflow users - Oracle Applications users who have an Oracle Workflow user responsibility, can:

  - Access the Oracle Workflow Self-service Home page.

  - Set their own individual user preferences.

- View and respond to their own notifications through the Worklist pages.

- Define rules to handle their own notifications automatically in their absence.

- View and respond to notifications for other users who have granted access to their worklists.

- View their own processes in the self-service Status Monitor. Users who are associated with the workflow administrator role can also reassign notifications or cancel workflows in the self-service Status Monitor.

Additionally, administrators who manage Oracle Workflow must have the Oracle Applications System Administrator responsibility to access Oracle Applications Manager, or must have an Oracle Workflow administrator responsibility that includes direct access to the Workflow Manager component within Oracle Applications Manager.

Also, administrators and developers who need to run Oracle Workflow scripts and programs or save workflow item type definitions to the database must have the password for the Oracle Workflow schema in the database.

Oracle Workflow also allows users to be assigned specialized workflow monitoring privileges with restricted access to workflow data. Such users act as specialized workflow administrators only within the Status Monitor, and can only view workflows and perform administrative actions as specified in their grants. See: Assigning Specialized Workflow Monitoring Privileges, page 5-18.

Resources Protected

Oracle Workflow provides security to protect the following resources.

- Oracle Workflow Web pages - Oracle Applications users must log into Oracle Applications before they can access Oracle Workflow Web pages.

- Workflow Manager - An Oracle Applications system administrator must log into Oracle Applications Manager in order to access the Workflow Manager component.

- Oracle Workflow Builder - No login is required to run the Oracle Workflow Builder development tool on a client PC. However, in order to view or save item type definitions in the database using Oracle Workflow Builder, a developer must provide the Oracle Workflow schema name and password.

- Administrative scripts and programs - An administrator must provide the Oracle Workflow schema name and password in order to run Oracle Workflow administrative scripts, the Workflow Definitions Loader, or the Workflow XML Loader. Additionally, an Oracle Applications system administrator must log into Oracle Applications or Oracle Applications Manager before running Oracle Workflow concurrent programs.

- E-mail notifications - Oracle Workflow supports sending e-mail notifications to users and processing e-mail responses to update workflow processes. Ultimately, the security of e-mail notifications and responses depends on the security of your e-mail application. Oracle Workflow also provides several features to validate e-mail responses and protect application content from unauthorized updates. For more information, see: E-mail Notification Security, page 2-68.

- Notifications - Oracle Workflow supports electronic signatures for users' responses to notifications. You can optionally require notification responses to be authenticated by either a password-based signature or a certificate-based signature. See: #WF_SIG_POLICY Attribute, *Oracle Workflow Developer's Guide*.

Authorization and Access Enforcement

Users are prompted for a username and password in order to access Oracle Workflow Web pages and Oracle Applications Manager. Users must additionally be assigned a responsibility that includes Oracle Workflow Web pages before they can access these pages.

Users must provide the Oracle Workflow database schema username and password to run administrative scripts and programs and to access workflow definitions in the database through Oracle Workflow Builder.

For information about authorization and validation of e-mail notification responses, see: E-mail Notification Security, page 2-68.

Leveraging Oracle Application Server Security Services

For information about use of Oracle HTTP Server by Oracle Applications, see: Administering Oracle HTTP Server, *Oracle Applications System Administrator's Guide*.

Leveraging Oracle Identity Management Infrastructure

An Oracle Workflow directory service based on users and roles from the unified Oracle Applications environment is automatically implemented for you during installation. For information about setting up Oracle Applications to use Oracle Internet Directory and single sign-on, see: Oracle Single Sign-On Integration, *Oracle Applications System Administrator's Guide - Security*.

Configuring Oracle Workflow Security

You can configure several options in Oracle Workflow to take advantage of the security features you want.

You can set the following global workflow preferences related to security.

- Workflow administrator, which defines the role that has administrator privileges in accessing Oracle Workflow Web pages.

- LDAP preferences, if you are integrating with Oracle Internet Directory. LDAP preferences include LDAP host, LDAP port, LDAP password, LDAP changelog base directory, and LDAP user base directory. LDAP password values are masked as asterisks in the display and are stored in encrypted form.

See: Setting Global User Preferences, page 2-9.

For information about configuring e-mail notification security options, see: E-mail Notification Security, page 2-68.

Directory service views for users and roles from the unified Oracle Applications environment are automatically implemented for you during installation. Oracle Workflow uses a directory service model in which denormalized information is maintained in the Workflow local tables for performance gain. The local Workflow directory service tables store user and role information originating from various other Oracle Applications modules, as well as ad hoc users and roles, so that the Workflow directory service views can access this information with good performance. You should maintain synchronization between the user and role information stored in application tables by the source modules and the information stored in the Workflow local tables. See: Setting Up a Directory Service for Oracle Workflow, page 2-16.

Also, you can optionally give users access to the Advanced Worklist, Personal Worklist, and Notification Search Web pages from any responsibility you choose. To make a page available from a particular responsibility, you must add the appropriate function to the menu associated with that responsibility. Then you can assign that responsibility to your users. See: Adding Worklist Functions to User Responsibilities, page 2-136.

Similarly, you can give users access to the Workflow Monitor Test Application from a responsibility that you choose. To make the Workflow Monitor Test Application available from a particular responsibility, you must add its menu to a top-level menu for that responsibility. Then you can assign that responsibility to your users. See: Testing Status Monitor Access, page 5-34.

You can use a special message attribute with the internal name #WF_SIG_POLICY to require that a user's response to a notification be authenticated by an electronic signature. Otherwise, the response will not be considered valid.

- If you define a notification to require a password-based signature, users must confirm their response by entering their Oracle Applications user name and password.

- If you define a notification to require a certficate-based signature, users must sign their response with a valid X.509 certificate issued by a certificate authority.

See: #WF_SIG_POLICY Attribute, *Oracle Workflow Developer's Guide*.

Additionally, a user can grant access to his or her worklist to another user. That user can then act as a proxy to handle notifications on the owner's behalf. The worklist access feature lets one user allow another user to handle his or her notifications without giving the second user access to any other privileges or responsibilities that the first user has in

Oracle Applications. However, note that a user who has access to another user's worklist can view all the details of that user's notifications and take most actions that the owner can take on the notifications. Ensure that your users take all necessary security considerations into account when they choose to grant worklist access to another user. See: Worklist Access, *Oracle Workflow User's Guide*.

4

Oracle Workflow Home Page

This chapter discusses the Oracle Workflow home page, where administrators can centrally access the Web-based features of Oracle Workflow.

This chapter covers the following topics:

- Accessing the Oracle Workflow Administrator Home Page

Accessing the Oracle Workflow Administrator Home Page

Use the Oracle Workflow administrator home page to link to Oracle Workflow's Web-based administration features in Oracle Applications. This page centralizes your access to the features.

> **Note:** Oracle Application Framework Personalization provides the ability to customize Oracle self-service Web application pages. If you are logged in as a user or with a responsibility for which the Personalize Self-Service Defn profile option is set to Yes, you can choose the Personalize button or Personalize Region links to create your personalizations. For more information, see: Personalization, *Oracle Application Framework Personalization Guide*.

To Access the Oracle Workflow Administrator Home Page:

1. Use a Web browser to navigate to the administrator home page, using a responsibility and navigation path specified by your system administrator. See: Oracle Workflow Administrator Navigation Paths, page A-1.

2. To set your Oracle E-Business Suite general preferences, including language and notification preferences, select Preferences. See: Set Preferences, *Oracle Applications User's Guide*.

3. The Notifications at a Glance region of the home page lists your five highest priority

notifications, ordered by priority and then by date.

- To view and respond to a notification, select the notification subject link in the Subject column. See: To View the Details of a Notification, *Oracle Workflow User's Guide*.

- To view the complete list of all your notifications in the Advanced Worklist, select the Full List button. See: To View Notifications from the Advanced Worklist, *Oracle Workflow User's Guide*.

- To define vacation rules for yourself, select the Vacation Rules link. See: To View and Maintain Vacation Rules, *Oracle Workflow User's Guide*.

- To grant access to your worklist to another user, select the Worklist Access link. See: To Grant Access to Your Worklist, *Oracle Workflow User's Guide*.

- To view another user's worklist, select the Switch User button, and then select the user you want. This button appears only if you have been granted access to another user's worklist.

4. To view workflow definitions and run test workflow processes, choose the Developer Studio tab. See: Testing Workflow Definitions Using the Developer Studio, *Oracle Workflow Developer's Guide*.

5. To manage Business Event System events, subscriptions, systems, and agents, choose the Business Events tab. See: Event Manager, *Oracle Workflow Developer's Guide*.

6. To view the administrator version of the Status Monitor, choose the Status Monitor tab. See: Accessing the Administrator Monitor, page 5-1.

7. To view notifications in the Worklist, choose the Notifications tab. See: To View Notifications from the Advanced Worklist, *Oracle Workflow User's Guide*.

8. To set global workflow preferences, choose the Administration tab, and choose Workflow Preferences in the horizontal navigation. See: Setting Global User Preferences, page 2-9.

9. To define vacation rules that automatically handle notifications in a user's absence, choose the Administration tab, and choose Vacation Rules in the horizontal navigation. See: Defining Vacation Rules for Users, page 6-4.

10. To access notifications belonging to another user, choose the Administration tab, and choose Notification Search in the horizontal navigation. See: Searching for Users' Notifications, page 6-1.

5

Monitoring Workflow Processes

This chapter discusses how to monitor an instance of a workflow process.

This chapter covers the following topics:

- Accessing the Administrator Monitor
- Viewing Workflows in the Status Monitor
- Viewing Child Workflows
- Viewing Activity History
- Viewing a Status Diagram
- Viewing Responses
- Viewing Workflow Details
- Viewing Error Information for a Workflow
- Assigning Specialized Workflow Monitoring Privileges
- Providing Access to the Status Monitor from Applications
- Testing Status Monitor Access
- Testing Standard Access
- Testing Guest Access
- Status Monitor Portlets

Accessing the Administrator Monitor

The Administrator Monitor lets you view and administer runtime workflows. You can use the monitor to review the activities performed for a workflow, check the status of the workflow by viewing the status diagram, review notifications sent by the workflow and participant responses, and review details about the workflow definition and attributes.

If you have workflow administrator privileges, you can also perform administrative

actions within the monitor to control the processing performed for a workflow or handle an error. For example, you can update workflow attributes, suspend and resume a workflow, cancel a workflow, or rewind the workflow to an earlier activity stage. You can also skip or retry an individual activity within a workflow, reassign a notification activity, or suspend and resume a process activity. Workflow administrator privileges are assigned in the Workflow Configuration page. See: Setting Global User Preferences, page 2-9.

> **Note:** If you intervene in a workflow process with an administrative operation such as skipping, retrying, or reassigning an activitiy, then Oracle Workflow updates the audit trail for the process to show that you performed that operation.

Oracle Workflow also allows users to be assigned specialized workflow monitoring privileges with restricted access to workflow data. If you have specialized workflow monitoring privileges, you can only view workflows and perform administrative actions to which you have been granted access. For example, you may only have access to monitor workflows belonging to particular item types, and you may be able to suspend and rewind workflows but not to skip activities. See: Assigning Specialized Workflow Monitoring Privileges, page 5-18.

If you only have specialized workflow monitoring privileges, you cannot perform any actions on workflows that you own yourself, irrespective of any permissions granted to you. Only users with full workflow administrator privileges assigned in the Workflow Configuration page can perform administrative actions on workflows that they own themselves.

> **Note:** If you have full workflow administrator privileges assigned in the Workflow Configuration page, then those privileges override any specialized workflow monitoring privileges assigned to you. That is, if you have full workflow administrator privileges, you can access all workflows, irrespective of any restrictions defined for any specialized privileges.

You can also use the Oracle Workflow Manager component of Oracle Applications Manager as an additional administration tool to review and manage work items. See: Oracle Workflow Manager Overview, page 7-1.

Additionally, Oracle Workflow provides a view called WF_ITEM_ACTIVITY_STATUSES_V that lets you programmatically access workflow status information. See: Oracle Workflow Views, *Oracle Workflow API Reference*.

1. Viewing Workflows in the Status Monitor, page 5-3

2. Viewing Child Workflows, page 5-6

3. Viewing Activity History, page 5-6

Viewing Workflows in the Status Monitor

When you access the Status Monitor, you can search for the workflows you want to review.

> **Note:** If you have full workflow administrator privileges, you can search for workflows owned by any user. If you only have specialized workflow monitoring privileges, you can only search for workflows to which you have been granted access. If you do not have workflow administrator privileges, you can only search for workflows that you own.

To Search for Workflows in the Status Monitor:

1. Use a Web browser to navigate to the administrator Status Monitor, using a responsibility and navigation path specified by your system administrator. See: Oracle Workflow Administrator Navigation Paths, page A-1.

2. In the Workflows page, search for the workflows you want to review. The search criteria are:

   • Workflow Type - Select the workflow item type you want to review. The display name for the workflow type you select populates the Workflow Type field, and the internal name for the workflow type you select populates the Type Internal Name field.

   • Type Internal Name - Enter the internal name of the workflow type you want to review, if you want to enter the internal name directly instead of selecting a value.

   • Workflow Owned By - If you have workflow administrator privileges, select the user who owns the workflows you want to review.

     If you do not have workflow administrator privileges, you can only search for workflows that you own. In this case, Oracle Workflow displays your name as a non-editable value in the Workflow Owned By field.

   • Item Key - Enter the item key that uniquely identifies the workflow you want to

review. You can enter a partial value to search for workflows whose item keys begin with that value.

- User Key - Enter the user key that identifies the workflow you want to review. You can enter a partial value to search for workflows whose user keys begin with that value.

- Workflow Status - Choose the status of the workflows you want to review, or choose Any Status to display workflows in any status.

    - In Process - Workflows that do not have an end date (including errored workflows)

    - Complete - Workflows that have an end date

    - Error - Workflows that do not have an end date and have at least one errored activity

- Workflow Started - Choose Today, This Week (last seven days), Last 2 Weeks (last fourteen days), Last 30 Days, Last 60 Days, or Any Time to specify the start date of the workflows you want to review. All the start date ranges include the current date; for example, Last 2 Weeks includes today as well as the previous thirteen days.

    Note: You must enter at least one of the following criteria when you search in order to limit the size of the results list.

    - Workflow Type

    - Type Internal Name

    - Workflow Owned By

    - Workflow Started - If you search only by this option, you must select a specific start date range. You cannot use Workflow Started with the Any Time value as your only search option.

You can also enter the following additional search criteria to search for workflows by activity characteristics.

- Activity Status - Select an activity status to display workflows with one or more activities in that status, or select Any Status to display workflows with activities in any status. You can choose the following statuses:

    - Active

- Complete

- Deferred

- Error

- Notified

- Suspended

- Waiting

- Waiting for Response From - Enter a role to display workflows with activities that are waiting for a response from the specified recipient.

- Days Without Progress - Enter a number of days to display in-process workflows with activities that have not progressed in the specified time.

3. The results region displays the workflows that match your search criteria.

> **Note:** If you only have specialized workflow monitoring privileges, the results region displays only workflows to which you have been granted access.

- To view the error details for an errored workflow, select the error icon or the error link in the Status column.

- To send e-mail to the owner of a workflow, select the user link in the Owned By column.

- To view child workflows for a workflow, select the child icon in the Child Workflows column.

4. Use the monitor buttons to drill down to additional information for a workflow.

- To review the activities executed within a workflow, select the workflow and select the Activity History button.

- To view the status diagram for a workflow, select the workflow and select the Status Diagram button.

- To review notifications sent by a workflow and participant responses, select the workflow and select the Participant Responses button.

- To view details for a workflow, including the workflow type definition and current workflow attribute values, select the workflow and select the Workflow

Details button.

Viewing Child Workflows

The Child Workflows page shows the workflows that are designated as children of a particular workflow. The page identifies the parent workflow by its workflow type internal name and item key, as well as by its user key, if it has one.

- To view the error details for an errored child workflow, select the error icon or the error link in the Status column.

- To send e-mail to the owner of a child workflow, select the user link in the Owned By column.

Use the monitor buttons to drill down to additional information for a child workflow.

- To review the activities executed within a workflow, select the workflow and select the Activity History button.

- To view the status diagram for a workflow, select the workflow and select the Status Diagram button.

- To review notifications sent by a workflow and participant responses, select the workflow and select the Participant Responses button.

- To view details for a workflow, select the workflow and select the Workflow Details button.

Viewing Activity History

The Activity History page shows information about the activities executed for a workflow. The page identifies the displayed workflow by its workflow type internal name and item key, as well as by its user key, if it has one.

Next, a hierarchical grid displays the process hierarchy formed by the selected workflow and any child workflows associated with it.

- To view the activity history for a workflow in the hierarchy, select the workflow

process name link in the Process Name column.

- To send e-mail to the owner of a workflow, select the user link in the Owned By column.

Specify the activity type and activity status of the activities you want to view. All activity types and statuses are selected by default. To search for specific activities, deselect any activity types and statuses you do not want to view. At least one activity type and one activity status must be selected for a search to be performed.

- To view details about the definition and current status of an activity, select the activity name link in the Activity column.

- To send e-mail to the performer of a notification, select the user link in the Performer column.

If you have the appropriate workflow administrator privileges, you can use the activity administration icons to perform administrative operations on notification or process activities that are not yet complete.

- To reassign a notification activity that is not yet complete, select the reassign icon. See: To Reassign a Notification to Another User, *Oracle Workflow User's Guide*.

> **Note:** The reassign icon appears only for notification activities that are not yet complete. Additionally, you cannot reassign a notification from the Status Monitor if the Expand Roles option is selected for the notification or if it is a voting notification that tallies the recipients' responses. See: Voting Activity, *Oracle Workflow Developer's Guide*.

- To suspend a process activity that is not yet complete, select the suspend icon in the Suspend/Resume column. All further processing for the process is suspended until the process is resumed.

> **Note:** If you suspend the root process for a workflow, all subprocesses for the workflow are suspended as well.

> **Note:** The suspend icon appears only for process activities that are not yet complete.

- To resume a suspended process activity, select the resume icon in the Suspend/Resume column. Processing for the process is resumed and any activities that were transitioned to while the process was suspended are now executed.

> **Note:** If you resume the root process for a workflow, all subprocesses for the workflow are resumed as well.

> **Note:** The resume icon appears only for process activities that are suspended.

If you have the appropriate workflow administrator privileges, you can use the activity administration buttons to perform administrative operations on any activities that are not yet complete.

- To skip an activity and force the workflow to transition to the next activity, select the activity and select the Skip button. In the Skip page, enter the result value to assign to the activity if required.

> **Note:** You cannot skip a suspended process activity. You must resume the process before you can skip it.

- To retry an activity, select the activity and select the Retry button.

> **Note:** You cannot retry a suspended process activity. You must resume the process before you can retry it.

If you have the appropriate workflow administrator privileges and you are viewing a workflow that is not yet complete, you can use the workflow administration buttons to perform administrative operations on the workflow.

- To view the status diagram for the workflow, select the View Diagram button.

- To change the values of any item attributes, select the Update Attributes button. In the Update Workflow Attributes page, enter the new values you want.

> **Note:** You cannot update attributes of type event. However, you can select the event message link to view the event message details.

- To rewind the workflow to an earlier activity stage, select the Rewind button. In the Rewind page, select the activity you want from the list of activities that the Workflow Engine has reached. Oracle Workflow stops processing at the current activity and resumes processing at the specified activity.

> **Note:** You cannot rewind a suspended workflow. You must resume the workflow before you can rewind it.

- To suspend the workflow, select the Suspend Workflow button. All further processing for the workflow is suspended until the process is resumed, and all subprocesses are suspended as well.

- To resume the workflow if it is suspended, select the Resume Workflow button. Processing for the workflow is resumed, and all subprocesses are resumed as well. Any activities that were transitioned to while the workflow was suspended are now executed.

- To cancel the workflow, select the Cancel Workflow button. In the Cancel page, select Apply. The Workflow Engine sets the status of the workflow to #FORCE, and all processing for the workflow is canceled and cannot be restarted.

Related Topics

Accessing the Administrator Monitor, page 5-1

Viewing a Status Diagram

The Status Diagram page shows the process diagram for a workflow, including graphical cues about the status of the workflow and its activities. The page identifies the displayed workflow by its workflow type internal name and item key, as well as by its user key, if it has one.

The Current Status region displays the process title, diagram, and detail tabs for the workflow. Beneath the detail tabs, the administration buttons appear.

Process Title:

The process title appears in the upper left of the Current Status region and displays the workflow process name, type, and user key. If no user key has been set, then the item key is displayed instead. If you drill down into a subprocess in the process diagram, the process title displays the subprocess name.

Process Diagram Window:

The process diagram window is a scrolling canvas that displays the diagram of the workflow process or subprocess currently listed in the process title. This diagram is identical to the diagram created in Oracle Workflow Builder. Note, however, that you cannot use the Workflow Monitor to edit this diagram.

The process diagram window provides graphical cues about the status of the process and its activities.

- An activity icon may be highlighted with a colored box to indicate that it is in an "interesting" state. The following table shows what state each color indicates.

Activity States in the Process Diagram

Color of Box	State	Possible Status Code
Red	Error	ERROR
Green	Active/In Progress	ACTIVE, NOTIFIED, DEFERRED
Yellow	Suspended	SUSPEND
<none>	Normal	COMPLETE, WAITING, NULL

- Any transition (arrow) that has been traversed appears as a thick green line, while an untraversed transition appears as a thin black line.

- Click an activity icon in the diagram to select it and update the detail tab window to display information about the activity.

- Click any empty space in the diagram to deselect the currently selected activity icon and to refresh the detail tab window to display information about the current process as a whole.

- Double-click an activity icon that represents a subprocess to drill down to the diagram for the subprocess. This action automatically updates the process title to reflect the name of the subprocess and updates the detail tab window to display information about the subprocess as a whole.

  Alternatively, you can select the subprocess activity and choose Zoom In to drill down to the diagram for the subprocess. Choose Zoom Out to navigate back to the higher level process.

Detail Tab Window:

The detail tab window, which appears below the process diagram, is a vertically scrollable display area that provides information about a selected process or activity. The information appears as follows for each tab.

Definition Tab

- Current Location - Process display name/activity display name

- Item Type - Workflow type display name

- Activity Name - Activity display name

- Description - Activity description

- Activity Type - Process, Notice, Event, or Function

- Result Type - Result type display name

Usage Tab
- Current Location - Process display name/activity display name

- Start/End - No, Start, or End

- Performer - Role name or item attribute internal name

- Timeout - Timeout value in minutes or item attribute internal name

Status Tab
- Current Location - Process display name/activity display name

- Status - Activity status

- Result - Activity result

- Begin Date - Date activity begins

- End Date - Date activity ends

- Due Date - Date activity is due to time out

- Assigned User - Role name or item attribute internal name

- Error Message - Error message (shown only if activity status is 'Error')

Notification Tab
- Current Location - Process display name/activity display name

- Recipient - Recipient of notification

    > **Note:** If the selected activity is a notification activity where Expand
    > Roles is selected, but no response is required, then the recipient
    > shown is simply the role, rather than the individual users in the
    > role.

- Status - Notification status

- Begin Date - Date notification is delivered

- End Date - Date notification is closed

- Due Date - Date activity is due to time out

Administration Buttons:

If you have the appropriate workflow administrator privileges and you are viewing a workflow that is not yet complete, you can use the administration buttons to perform administrative operations on the workflow.

- To change the values of any item attributes, select the Update Attributes button. In the Update Workflow Attributes page, enter the new values you want.

  > **Note:** You cannot update attributes of type event. However, you can select the event message link to view the event message details.

- To rewind the workflow to an earlier activity stage, select the Rewind button. In the Rewind page, select the activity you want from the list of activities that the Workflow Engine has reached. Oracle Workflow stops processing at the current activity and resumes processing at the specified activity.

  > **Note:** You cannot rewind a suspended workflow. You must resume the workflow before you can rewind it.

- To suspend the workflow, select the Suspend Workflow button. All further processing for the workflow is suspended until the process is resumed, and all subprocesses are suspended as well.

- To resume the workflow if it is suspended, select the Resume Workflow button. Processing for the workflow is resumed, and all subprocesses are resumed as well. Any activities that were transitioned to while the workflow was suspended are now executed.

- To cancel the workflow, select the Cancel Workflow button. In the Cancel page, select Apply. The Workflow Engine sets the status of the workflow to `#FORCE`, and all processing for the workflow is canceled and cannot be restarted.

Related Topics

Accessing the Administrator Monitor, page 5-1

Viewing Responses

The Monitor Responses page shows information about notifications sent by a workflow and responses from workflow participants. For example, you can use this page to view individual responses to a voting activity. The page identifies the displayed workflow by

its workflow type internal name and item key, as well as by its user key, if it has one.

Next, a hierarchical grid displays the process hierarchy formed by the selected workflow and any child workflows associated with it.

- To view the notifications for a workflow in the hierarchy, select the workflow process name link in the Process Name column.

- To send e-mail to the owner of a workflow, select the user link in the Owned By column.

Specify the type and status of the notifications you want to view.

- Select the Response Notifications option to view response-required notifications, the FYI Notifications option to view information-only (FYI) notifications, or both.

- Select the Closed Notifications option to view only closed notifications of the selected type. Deselect this option to view both open and closed notifications.

Then review the notification list.

- To view and respond to a notification in the Notification Details page, select the subject link in the Notification Subject column. The Notification Details page includes the same options that are displayed when accessed through a user's worklist, allowing you to take any action that is available for the notification. See: To View the Details of a Notification, *Oracle Workflow User's Guide*.

- To send e-mail to the recipient of a notification, select the user link in the Recipient column.

- To view details about a notification and the recipient's response, select the details icon in the View Response Details column.

Viewing Notification Response Details:

The Notification Response Details page displays the following details about the selected response:

- Notification Subject - The subject line of the notification.

- Respondent - The user who responded to the notification. To send e-mail to the respondent, select the user link.

- Original Recipient - The user to whom the notification was originally sent. If the original recipient is different than the respondent, then the original recipient reassigned responsibility for replying to the respondent. To send e-mail to the original recipient, select the user link.

- Notification Sent - The date and time when the notification was sent.

- Response Received - The date and time when the response was received.

- Response - The response value that is the result of the notification.

The Signature Details region specifies whether the notification required an electronic signature, and if so, the signature type, either a password-based signature or a certificate-based digital signature. For a notification that required a signature, the region also displays the following information:

- Signature ID - The numerical identifier for the signature request. This information is shown only for certificate-based signatures.

- Signature - The text string representing the signature created using the signer's certificate. This information is shown only for certificate-based signatures.

- Actual Signer - The user who submitted the signature.

- Creation Date - The date when the request for a signature was created.

- Signed Date - The date when the user submitted the signature.

The Additional Response Information region lists any further response values requested in the notification in addition to the result response.

> **Note:** For information-only notifications or open response-required notifications, the Notification Response Details page only shows details about the outbound notification.

Related Topics

Accessing the Administrator Monitor, page 5-1

#WF_SIG_POLICY Attribute, *Oracle Workflow Developer's Guide*

Reviewing Electronic Signature Details, page 6-5

Viewing Workflow Details

The Workflow Details page shows information about a selected workflow. The page identifies the displayed workflow by its workflow type internal name and item key, as well as by its user key, if it has one.

Next, a hierarchical grid displays the process hierarchy formed by the selected workflow and any child workflows associated with it.

- To view the details for a workflow in the hierarchy, select the workflow process name link in the Process Name column.

- To send e-mail to the owner of a workflow, select the user link in the Owned By column.

The Workflow Definition region displays the following information about the workflow type:

- Internal Name - The workflow type internal name.

- Description - The workflow type description.

- Persistence Type - The persistence type, either Permanent if the runtime status information is maintained indefinitely until it is specifically purged, or Temporary if the runtime status information is maintained for a specified number of days after the workflow completion date before it can be purged.

- Persistence Days - If the persistence type is Temporary, the number of days from the time a workflow of this workflow type completes before its status audit trail can be purged.

- Selector - The selector/callback function that determines which process activity to run for the workflow type in a particular situation.

The Workflow Attributes region lists the names and values of the item attributes for the workflow. For an attribute of type event, select the event message link to view the event message details.

If you have the appropriate workflow administrator privileges and you are viewing a workflow that is not yet complete, you can use the administration buttons to perform administrative operations on the workflow.

- To view the status diagram for the workflow, select the View Diagram button.

- To change the values of any item attributes, select the Update Attributes button. In the Update Workflow Attributes page, enter the new values you want.

    Note: You cannot update attributes of type event. However, you can select the event message link to view the event message details.

- To rewind the workflow to an earlier activity stage, select the Rewind button. In the Rewind page, select the activity you want from the list of activities that the Workflow Engine has reached. Oracle Workflow stops processing at the current activity and resumes processing at the specified activity.

    Note: You cannot rewind a suspended workflow. You must resume the workflow before you can rewind it.

- To suspend the workflow, select the Suspend Workflow button. All further

processing for the workflow is suspended until the process is resumed, and all subprocesses are suspended as well.

- To resume the workflow if it is suspended, select the Resume Workflow button. Processing for the workflow is resumed, and all subprocesses are resumed as well. Any activities that were transitioned to while the workflow was suspended are now executed.

- To cancel the workflow, select the Cancel Workflow button. In the Cancel page, select Apply. The Workflow Engine sets the status of the workflow to #FORCE, and all processing for the workflow is canceled and cannot be restarted.

Related Topics

Accessing the Administrator Monitor, page 5-1

Viewing Error Information for a Workflow

The Workflow Errors page shows error information for an errored workflow. The page identifies the displayed workflow by its workflow type internal name and item key, as well as by its user key, if it has one.

The page displays the following error details:

- Failed Activity - The name of the activity that encountered an error.

- Activity Type - The activity type, either Function, Notification, Event, or Process.

- Error Name - The internal name of the error.

- Error Message - The error message describing the error.

- Error Stack - Context information to help you locate the source of the error.

    Note: If you navigated to the Workflow Errors page from an error status link for a workflow, the page shows details for each errored activity within the workflow. If you navigated to the page from an error status link for a single errored activity, the page shows details for only that activity.

To view the status diagram for the workflow, select the View Diagram button.

If the Workflow Errors page shows only one errored activity, and you have the appropriate workflow administrator privileges, you can use the activity administration buttons to respond to the error.

- To reassign an errored notification activity, select the Reassign button. See: To

Reassign a Notification to Another User, *Oracle Workflow User's Guide*.

> **Note:** The Reassign button appears only for a notification activity. Additionally, you cannot reassign a notification from the Status Monitor if the Expand Roles option is selected for the notification or if it is a voting notification that tallies the recipients' responses. See: Voting Activity, *Oracle Workflow Developer's Guide*.

- To suspend an errored process activity, select the Suspend button. All further processing for the process is suspended until the process is resumed.

> **Note:** If you suspend the root process for a workflow, all subprocesses for the workflow are suspended as well. The Suspend button appears only for an errored process activity.

- To resume an errored process activity that is suspended, select the Resume button. Processing for the process is resumed and any activities that were transitioned to while the process was suspended are now executed.

> **Note:** If you resume the root process for a workflow, all subprocesses for the workflow are resumed as well. The Resume button appears only for an errored process activity that is suspended.

- To skip the errored activity and force the workflow to transition to the next activity, select the Skip button. In the Skip page, enter the result value to assign to the activity if required.

> **Note:** You cannot skip a suspended process activity. You must resume the process before you can skip it.

- To retry the errored activity, select the Retry button.

> **Note:** You cannot retry a suspended process activity. You must resume the process before you can retry it.

Related Topics

Accessing the Administrator Monitor, page 5-1

Assigning Specialized Workflow Monitoring Privileges

You can designate certain users as administrators only for particular types of workflows by assigning those users specialized workflow monitoring privileges with restricted access to workflow data. You can base the restrictions on a defined set of item types or on criteria specific to a particular functional area.

Ensure that users who act as specialized workflow administrators have access to the administrator version of the Status Monitor, either through an Oracle Workflow responsibility or from another application. See: Oracle Workflow Administrator Navigation Paths, page A-1 and Providing Access to the Status Monitor from Applications, page 5-20.

You must also grant permissions to specialized workflow administrators to enable them to perform administrative actions within the Status Monitor for workflows to which they have access. However, note that users cannot perform any actions on workflows that they own themselves, irrespective of any permissions granted to them. Only users with full workflow administrator privileges assigned in the Workflow Configuration page can perform administrative actions on workflows that they own themselves.

> **Note:** If a user has full workflow administrator privileges assigned in the Workflow Configuration page, then those privileges override any specialized workflow monitoring privileges assigned to that user. That is, a user with full workflow administrator privileges can access all workflows, irrespective of any restrictions defined for any specialized privileges. See: Setting Global User Preferences, page 2-9.

To Grant Restricted Access to Workflow Monitoring Data:

You can restrict access to workflow monitoring data based only on item types, only on functional criteria, or on both item types and functional criteria. However, because grants based on functional criteria depend on item attribute values, these grants are most effective when combined with grants for item types that share the same item attributes.

For more information about creating instance sets and grants, see: Defining Data Security Policies, *Oracle Applications System Administrator's Guide - Security* and Assigning Permissions to Roles, *Oracle Applications System Administrator's Guide - Security*.

1. If you want to restrict access based on item types, perform the following steps.

    - Create an instance set on the object `WORKFLOW_ITEMS` with a predicate that includes the parameter in which you will specify the item type you want. For example, the following excerpt shows a sample predicate:

      ```
      &TABLE_ALIAS.ITEM_TYPE = &GRANT_ALIAS.PARAMETER1
      ```

- Create a grant using the instance set you created. First, specify appropriate security context information such as grantee and responsibility. Then specify the following data context information:

  - Object - `WORKFLOW_ITEMS`

  - Data Context Type - `Instance Set`

  - Instance Set - The instance set you created on `WORKFLOW_ITEMS`.

  - PARAMETER1 - The internal name of the item type to which you want to grant access, such as `WFDEMO`.

  - Permission Set: `Business workflow item permission set`

2. If you want to restrict access based on criteria specific to a particular functional area using item attributes, perform the following steps.

   - Create an instance set on the object `WORKFLOW_ITEM_ATTR_VALUES` with a predicate that defines those criteria. For example, the following excerpt shows a sample predicate defining criteria for HR data, using the `CURRENT_PERSON_ID` item attribute:

   ```
   &TABLE_ALIAS.NAME='CURRENT_PERSON_ID'
   and EXISTS (SELECT 'Y' FROM per_people_f
               WHERE person_id = &TABLE_ALIAS.TEXT_VALUE
               AND TRUNC (SYSDATE) BETWEEN effective_start_date
                 AND effective_end_date)
   ```

   As another example, if the workflows for a particular organization are marked with an item attribute named `ORG_ID`, the following excerpt shows a sample predicate that allows access only to workflows associated with the user's current organization context.

   ```
   &TABLE_ALIAS.NAME = 'ORG_ID'
   and &TABLE_ALIAS.TEXT_VALUE = substr
     (sys_context('USERENV','CLIENT_INFO'),1,10)
   ```

   - Create a grant using the instance set you created. First, specify appropriate security context information such as grantee and responsibility. Then specify the following data context information:

     - Object - `WORKFLOW_ITEM_ATTR_VALUES`

     - Data Context Type - `Instance Set`

     - Instance Set - The instance set you created on `WORKFLOW_ITEM_ATTR_VALUES`.

     - Permission Set - `Business workflow item attribute permission`

```
        set
```

To Grant Permissions for Administrative Actions within the Status Monitor:

If you want to assign a user privileges for all administrative actions within the Status Monitor, assign that user the role WF_ADMIN_ROLE.

If you want to assign a user privileges only for specific administrative actions, grant that user the corresponding permissions, as shown in the following table.

Permissions for Administrative Actions in the Status Monitor

Action	Permission Name	Permission Code
Skip	Skip Workflow Activity	WF_SKIP
Retry	Retry Activity	WF_RETRY
Rewind	Rewind Workflow	WF_REWIND
Suspend	Suspend Workflow	WF_SUSPEND
Cancel	Cancel Workflow	WF_CANCEL
Update	Update Workflow Item Attributes	WF_UPDATE_ATTR
Monitor	Monitor Data	WF_MON_DATA

See: Assigning Permissions to Roles, *Oracle Applications System Administrator's Guide - Security*.

Providing Access to the Status Monitor from Applications

Application developers can provide access to the administrator and self-service versions of the Status Monitor from within an application. The Status Monitor supports the following access modes:

- Standard access - Loosely coupled access from an Oracle Application Framework-based Web page to a Status Monitor page within the full Oracle Workflow application

- Guest access - Tightly coupled access from an Oracle Application Framework-based

Web page to a freestanding Status Monitor page for a particular workflow, without access to the rest of the Oracle Workflow application

Menu Setup

Before you can add Status Monitor access to your application, you must set up the menu for the appropriate responsibility to include Status Monitor functionality through Oracle Workflow menus.

For standard or guest access from an Oracle Application Framework-based Web page or an Oracle E-Business Suite form, the menu for the responsibility through which users access that page or form must also include the menu for the Status Monitor version and access mode you want. Otherwise, users will encounter function security errors when navigating to the Status Monitor from your page or form. To avoid these errors, add the appropriate Oracle Workflow menu to the top-level menu for your responsibility.

The menus for the Status Monitor are:

- Workflow Administrator Application (WF_ADMINISTRATOR_APPLICATION) - Required for standard access to the Administrator Monitor

- Workflow Guest Monitor Application (WF_G_MONITOR_APPLICATION) - Required for guest access to the Administrator Monitor

- Workflow Self-Service Application (WF_SELF_SERVICE_APPLICATION) - Required for standard access to the Self-Service Monitor

- Workflow Guest Self-Service Monitor Application (WF_SSG_MONITOR_APPLICATION) - Required for guest access to the Self-Service Monitor

Access through URLs

You can provide standard or guest access to the Status Monitor from an Oracle Application Framework-based Web page. You can also provide guest access to the administrator version of the Status Monitor from Oracle Workflow notifications or PL/SQL applications that can call a URL.

In your application code, call an API to obtain or redirect to a URL for an Oracle Workflow Status Monitor page. For standard and guest access, Oracle Workflow provides Java methods in the `oracle.apps.fnd.wf.monitor.webui.Monitor` class to obtain or redirect to appropriate URLs. For guest access to the Administrator Monitor, Oracle Workflow also provides PL/SQL functions in the WF_MONITOR package to obtain appropriate URLs. See: Workflow Monitor APIs, *Oracle Workflow API Reference*.

Display the Status Monitor by redirecting to the Status Monitor page you want or using the URL you obtained in whatever way is appropriate within your application code.

Access from Oracle E-Business Suite Forms

You can provide standard or guest access from an Oracle E-Business Suite application to the Status Monitor by using self-service functions. You can call these functions from an Oracle E-Business Suite form, or add the menus that contain the functions to another menu associated with another responsibility. Use the function FND_FUNCTION.EXECUTE to execute a Status Monitor function.

Related Topics

Accessing the Administrator Monitor, page 5-1

Accessing the Self-Service Monitor, *Oracle Workflow User's Guide*

Overview of Function Security, *Oracle Applications System Administrator's Guide*

Overview of Menus and Function Security, *Oracle Applications Developer's Guide*

Overview of Form Development Steps, *Oracle Applications Developer's Guide*

Menus Window, *Oracle Applications Developer's Guide*

Standard Access, page 5-22

Guest Access, page 5-26

Standard Access

Standard Status Monitor access provides loosely coupled access from an Oracle Application Framework-based Web page or Oracle E-Business Suite form, to a Status Monitor page within the full Oracle Workflow application. In this mode, the specified Status Monitor page is displayed with the full Oracle Workflow menu, allowing users to navigate out of the Status Monitor and perform other tasks within Oracle Workflow. Locator links, also known as breadcrumbs, let users navigate from the Status Monitor back to the calling application.

With standard access, users are fully authenticated. Only users with workflow administrator privileges, as specified in the Workflow Configuration page, can view workflows owned by others and perform administrative operations in the Status Monitor. See: Setting Global User Preferences, page 2-9.

When you provide standard access to the Status Monitor from your application, you can optionally specify a workflow item type and item key to query and specify which page you want to initially display.

- If you specify both the item type and item key for a workflow, that workflow is automatically queried in the Status Monitor. You can choose to initially display the workflow in the main Workflows search page, the Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor, the Status Diagram page, or the Monitor Responses page. If you provide an item type and item key but do not specify an initial page, the workflow is initially displayed

in the Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor.

> **Note:** The specified workflow is only displayed if the user owns that workflow or has workflow administrator privileges to view workflows owned by others. If the user does not have the necessary privileges to view the workflow, an error message is displayed. In this case the user can optionally select another Workflow tab.

- If you specify only a workflow item type, the main Workflows search page is displayed, and workflows of the specified type that were started within the last two weeks are automatically queried.

- If you do not specify a workflow item type, the main Workflows search page is displayed. No automatic query is performed.

Standard Access in Java

Oracle Workflow provides Java methods to redirect to the Status Monitor with standard access or to obtain URLs for standard access to the Status Monitor. These methods are defined in the Java class called `oracle.apps.fnd.wf.monitor.webui.Monitor`.

- *Monitor.redirectToAdvanceUrl( )* - Redirects to the Administrator Monitor with standard access.

- *Monitor.redirectToSimpleUrl( )* - Redirects to the Self-Service Monitor with standard access.

- *Monitor.getAdvanceUrl( )* - Returns a URL for standard access to the Administrator Monitor, in a form that can be used within an Oracle Application Framework application page. For example, you can set this URL as a destination link on an `OAWebBean`. The URL is returned in the following format:

  `/OA_HTML/OA.jsp?OAFunc=[parameters...]`

- *Monitor.getSimpleUrl( )* - Returns a URL for standard access to the Self-Service Monitor, in a form that can be used within an Oracle Application Framework application page. For example, you can set this URL as a destination link on an `OAWebBean`. The URL is returned in the following format:

  `/OA_HTML/OA.jsp?OAFunc=[parameters...]`

When calling these methods, you must provide the following parameters to indicate how you want to display the Status Monitor:

- `pageContext` - The `OAPageContext` of the calling page.

- `itemType` - The internal name of a workflow item type to automatically query in

the Status Monitor (optional).

- `itemKey` - An item key to automatically query in the Status Monitor (optional).

- `firstPage` - The Status Monitor page that you want to initially display (optional).
  - `MAIN` - Main Workflows search page

  - `HISTORY` - Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor

  - `DIAGRAM` - Status Diagram page

  - `RESPONSES` - Monitor Responses page

    > **Note:** For access to the `HISTORY`, `DIAGRAM`, or `RESPONSES` pages, you must specify an item type and item key.

- `retainCallingAM` - Specify `true` or `false` to indicate whether the `OAApplicationModule` of the calling page should be retained while working in the Status Monitor. If you specify `true`, `retainAM=Y` is appended to the Status Monitor URL; if you specify `false`, `retainAM=N` is appended to the URL. The default is `true`.

All method calls for standard access should be made from within an Oracle Application Framework-based Web page.

Standard Access from Oracle E-Business Suite Forms

You can provide standard access from an Oracle E-Business Suite application to the Status Monitor by using self-service functions. You can call these functions from an Oracle E-Business Suite form, or add the menus that contain the functions to another menu associated with another responsibility.

The following table lists the functions that provide standard access to the Status Monitor. These functions do not require any parameters.

Standard Access Functions

Function	Description
WF_STATUS_MONITOR	Displays the Workflows search page in the administrator version of the Status Monitor.

Function	Description
WF_SS_STATUS_MONITOR	Displays the Workflows search page in the self-service version of the Status Monitor.

You can call the function FND_FUNCTION.EXECUTE to execute a standard access function. See: FND_FUNCTION.EXECUTE, *Oracle Applications Developer's Guide*.

The following code example demonstrates how to execute the WF_STATUS_MONITOR function using FND_FUNCTION.EXECUTE.

```
FND_FUNCTION.EXECUTE(FUNCTION_NAME => 'WF_STATUS_MONITOR');
```

See: Overview of Form Development Steps, *Oracle Applications Developer's Guide* and Menus Window, *Oracle Applications Developer's Guide*.

Menu Setup

If you use a standard access function within your responsibility, you must add the menu containing that function to the top-level menu for your responsibility. The WF_STATUS_MONITOR function is seeded on the Workflow Administrator Application (WF_ADMINISTRATOR_APPLICATION) menu, and the WF_SS_STATUS_MONITOR function is seeded on the Workflow Self-Service Application (WF_SELF_SERVICE_APPLICATION) menu.

> **Note:** You cannot add the Status Monitor functions to your menu directly. To include these functions, you must add the Oracle Workflow menu that contains the function you want.

The Workflow Administrator Application menu is seeded on the Workflow Administrator (New) menu (FND_WFADMIN_NEW) associated with the Workflow Administrator Web (New) responsibility. Similarly, the Workflow Self-Service Application menu is seeded on the Workflow User (New) menu (FND_WFUSER_NEW) associated with the Workflow User Web (New) responsibility.

Migrating to Standard Access Functions

If your application used self-service functions to provide access to the previous version of the Workflow Monitor, you can migrate to the new Status Monitor by replacing the functions you previously called with the appropriate new functions. The following table shows the correspondence between functions for the previous Workflow Monitor and the standard access functions for the new Status Monitor.

Migrating to Standard Access Functions

Previous Function	New Function
FND_FNDWFIAS (if called without any parameters)	WF_STATUS_MONITOR or WF_SS_STATUS_MONITOR

Related Topics

Migrating to Guest Access Functions, page 5-33

Guest Access

Guest Status Monitor access provides tightly coupled access from an Oracle Application Framework-based Web page or Oracle E-Business Suite form, to a freestanding Status Monitor page for a particular workflow, without access to the rest of the Oracle Workflow application. In this mode, the specified Status Monitor page is displayed without the full Oracle Workflow menu, so that users cannot navigate out of the Status Monitor to other Oracle Workflow pages.

Guest access lets users navigate among the top-level Status Monitor pages, including the Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor, Status Diagram page, Monitor Responses page, and Workflow Details page. Users can also drill down from the top-level pages to related detail or action pages, such as navigating from the Notification History page to the Cancel page. Additionally, locator links, also known as breadcrumbs, let users navigate from the Status Monitor back to the calling application. However, users cannot navigate from the Status Monitor to other parts of Oracle Workflow.

With guest access, users are fully authenticated. However, in this mode you programmatically control which workflow process users can view and whether users can perform administrative operations in the Status Monitor, rather than having these privileges controlled by the workflow administrator setting in the Workflow Configuration page.

When you provide guest access to the Status Monitor from your application, you must specify a workflow item type and item key to automatically query. Guest access does not include search capabilities, so users can only view the workflow process you specify. You can choose to initially display the specified workflow in either the Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor, the Status Diagram page, or the Monitor Responses page. If you do not specify an initial page, the workflow is initially displayed in the Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor.

> **Note:** The workflow is only displayed if the specified item type, item key, and administrator mode are valid. Otherwise, an error message is displayed. The Workflow tabs are not displayed, so the user cannot navigate to any other part of Oracle Workflow.

You must also set the administrator mode to determine whether to grant the user privileges to perform administrative operations within the Status Monitor. You can choose one of the following options:

- Never grant administrator privileges, regardless of whether the user belongs to the workflow administrator role specified in the Workflow Configuration page. This option is the default if you do not specify an administrator mode.

- Always grant administrator privileges, regardless of whether the user belongs to the workflow administrator role specified in the Workflow Configuration page.

- Check whether the user belongs to the workflow administrator role specified in the Workflow Configuration page and grant administrator privileges accordingly.

Guest Access in Java

Oracle Workflow provides Java methods to obtain URLs for guest access to the Status Monitor. These methods are defined in the Java class called `oracle.apps.fnd.wf.monitor.webui.Monitor`.

- *Monitor.redirectToGuestAdvUrl( )* - Redirects to the Administrator Monitor with guest access.

- *Monitor.redirectToGuestSimpleUrl( )* - Redirects to the Self-Service Monitor with guest access.

- *Monitor.getGuestAdvanceUrl( )* - Returns a URL for guest access to the Administrator Monitor, in a form that can be used within an Oracle Application Framework application page. For example, you can set this URL as a destination link on an `OAWebBean`. The URL is returned in the following format:

  `/OA_HTML/OA.jsp?OAFunc=[`*parameters...*`]`

- *Monitor.getGuestSimpleUrl( )* - Returns a URL for guest access to the Self-Service Monitor, in a form that can be used within an Oracle Application Framework application page. For example, you can set this URL as a destination link on an `OAWebBean`. The URL is returned in the following format:

  `/OA_HTML/OA.jsp?OAFunc=[`*parameters...*`]`

When calling these methods, you must provide the following parameters to indicate how you want to display the Status Monitor:

- `pageContext` - The `OAPageContext` of the calling page.

- `itemType` - The internal name of a workflow item type to automatically query in the Status Monitor.

- `itemKey` - An item key to automatically query in the Status Monitor.

- `adminMode` - Specify `'Y'` to grant administrator privileges to the user accessing the Status Monitor, `'N'` to withhold administrator privileges from the user, or `'U'` to check whether the user belongs to the workflow administrator role specified in the Workflow Configuration page and grant administrator privileges accordingly. The default is `'N'`.

- `firstPage` - The Status Monitor page that you want to initially display.
  - `HISTORY` - Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor
  - `DIAGRAM` - Status Diagram page
  - `RESPONSES` - Monitor Responses page

  The default is `HISTORY`.

- `retainCallingAM` - Specify `true` or `false` to indicate whether the `OAApplicationModule` of the calling page should be retained while working in the Status Monitor. If you specify `true`, `retainAM=Y` is appended to the Status Monitor URL; if you specify `false`, `retainAM=N` is appended to the URL. The default is `true`.

All method calls for guest access should be made from within an Oracle Application Framework-based Web page.

Example

The following code excerpt shows an example of how to provide guest access to the Status Monitor in Java code. This example calls the *getGuestAdvanceUrl()* method in the `oracle.apps.fnd.wf.monitor.webui.Monitor` class.

```
/*****************************************************************
**
** Guest Access
**
** Assumes all method calls are made from within an Oracle
** Application Framework page.
**
*****************************************************************/

...
import oracle.apps.fnd.wf.monitor.webui.Monitor;
...

  // This example assumes we want to set the destination link on
  // an OA text bean.Following the link will take the user to the
  // advanced monitor on the "Activity History" page.

  public void processRequest(OAPageContext pageContext,
                             OAWebBean webBean)
  {
    super.processRequest(pageContext, webBean);

 ...

      String itemType = [ however this value is obtained in page ];
      String itemKey = [ however this value is obtained in page ];
      String firstPage = "HISTORY";
      String returnToLabel = "Return to Application XYZ";

      // Set to "U" to have Monitor code figure out if the current
      // user should have Workflow Administrator privileges based
      // on the Administrator role designation in Workflow.

      String adminMode = "U";

      // Will add a parameter "retainAM=Y" to the resulting url
      // so the developer doesn't have to do this manually if he
      // wants to retain the calling Application Module when the
      // user navigates to the Status Monitor.

      boolean retainCallingAM = true;

    String url = null;

    try
    {
      url = Monitor.getGuestAdvanceUrl(pageContext, itemType,
                                  itemKey, adminMode, firstPage,
                                  returnToLabel, retainCallingAM);
    }
    catch (MonitorURLException me)
    {
      // Handle not being able to obtain a valid redirectUrl for
      // the parameters.
    }

    // Set the url string on the web bean.

    OAStaticStyledTextBean monitorLink =
      (OAStaticStyledTextBean)findIndexedChildRecursive
      ("AdvancedMonitorLink");
```

```
        monitorLink.setDestination(url);

          ...

      } // end processRequest()
```

Guest Access in PL/SQL

Oracle Workflow also provides PL/SQL functions to obtain URLs for guest access to the Administrator Monitor. These functions are defined in the PL/SQL package called WF_MONITOR. See: Workflow Monitor APIs, *Oracle Workflow API Reference*.

- *WF_MONITOR.GetAdvancedEnvelopeURL( )* - Returns a complete URL for guest access to the Activity History page in the Administrator Monitor.

- *WF_MONITOR.GetDiagramURL( )* - Returns a complete URL for guest access to the Status Diagram page in the Administrator Monitor.

- *WF_MONITOR.GetEnvelopeURL( )* - Returns a complete URL for guest access to the Monitor Responses page in the Administrator Monitor.

When calling these methods, you must provide the following parameters to indicate how you want to display the Status Monitor:

- `x_agent` - This parameter is no longer used. Set this parameter to `null`.

- `x_item_type` - The internal name of a workflow item type to automatically query in the Status Monitor.

- `x_item_key` - An item key to automatically query in the Status Monitor.

- `x_admin_mode` - Specify `'YES'` to grant administrator privileges to the user accessing the Status Monitor, or `'NO'` to withhold administrator privileges from the user. The default is `'NO'`.

You can use these URLs to provide access to the Administrator Monitor from a PL/SQL application, for example, or include a URL in a workflow notification message to allow a user to access the Administrator Monitor from the notification.

> **Note:** In Oracle E-Business Suite, you can call the function FND_UTILITIES.OPEN_URL to open a Web browser and have it connect to a specified URL, such as a Status Monitor URL. See: FND_UTILITIES:Utility Routine, *Oracle Applications Developer's Guide*.

Guest Access from Notifications

You can provide guest access to the Administrator Monitor from a workflow notification. To do so, define a message attribute of type URL, and include or attach this attribute in the notification message. Obtain a guest access URL using one of the

WF_MONITOR PL/SQL functions, and set the value of the message attribute to this URL. The user who receives the notification can access the Administrator Monitor by viewing the notification, either through the Worklist Web pages or through an e-mail application, and clicking the link in the message. If users are not already logged into Oracle Applications, they must first log in before they can access the Administrator Monitor from the link. See: To Define a Message Attribute, *Oracle Workflow Developer's Guide*.

Guest Access from Oracle E-Business Suite Forms

You can provide guest access from an Oracle E-Business Suite application to the Status Monitor by using self-service functions. You can call these functions from an Oracle E-Business Suite form, or add the menus that contain the functions to another menu associated with another responsibility.

The following table lists the functions that provide guest access to the Status Monitor.

Guest Access Functions

Function	Description
WF_G_ACTIVITIES	Displays the Activity History page for the specified workflow in the administrator version of the Status Monitor.
WF_G_DIAGRAM	Displays the Status Diagram page for the specified workflow in the administrator version of the Status Monitor.
WF_SSG_ACTIVITIES	Displays the Notification History page for the specified workflow in the self-service version of the Status Monitor.
WF_SSG_DIAGRAM	Displays the Status Diagram page for the specified workflow in the self-service version of the Status Monitor.

When you call one of the guest access functions, you must pass the function the following parameters:

- `itemType` - A valid workflow item type, determined by your application. The item type and item key together identify the workflow process to display. You must specify the same item type as you used to obtain the encrypted access key. You should use the *ICX_CALL.Encrypt()* function to encrypt this value.

- `itemKey` - A valid item key, determined by your application. The item type and

item key together identify the workflow process to display. You must specify the same item key as you used to obtain the encrypted access key. You should use the *ICX_CALL.Encrypt()* API to encrypt this value.

- `wm` - The encrypted administrator mode that determines whether the user should have privileges to perform administrative operations in the Status Monitor. Call the PL/SQL function *WF_FWKMON.GetEncryptedAdminMode()* to obtain the encrypted value for the administrator mode you want, either `Y` or `N`. You must specify the same administrator mode value to encrypt as you used to obtain the encrypted access key. See: GetEncryptedAdminMode, *Oracle Workflow API Reference*.

- `wa` - An encrypted access key for a specified item type, item key, and administrator mode combination. Call the PL/SQL function *WF_FWKMON.GetEncryptedAccessKey()* to obtain this value for the item type, item key, and administrator mode you want. See: GetEncryptedAccessKey, *Oracle Workflow API Reference*.

> **Note:** Because users are authenticated in guest access, you can call the PL/SQL function *WF_FWKMON.IsMonitorAdministrator()* to determine whether a the user has administrator privileges based on the workflow administrator setting in the Workflow Configuration page. If you use this function, you should use its result when obtaining both the encrypted access key and the encrypted administrator mode, in order to avoid a discrepancy between these two values. See: IsMonitorAdministrator, *Oracle Workflow API Reference*.
>
> You can also choose to grant or withhold administrator privileges in the Status Monitor by specifying the administrator mode as `Y` or `N`, respectively, regardless of the workflow administrator setting in the Workflow Configuration page.

- `retainAM` - Specify `Y` or `N` to indicate whether the `OAApplicationModule` of the calling page should be retained while working in the Status Monitor.

- `fExt` - An external flag used within Oracle Workflow. Set this parameter to `X`.

You can call the function FND_FUNCTION.EXECUTE to execute a guest access function specifying your parameters. See: FND_FUNCTION.EXECUTE, *Oracle Applications Developer's Guide*.

> **Note:** When you call FND_FUNCTION.EXECUTE, you should use the *WFA_HTML.conv_special_url_chars()* API to convert any special characters in the parameters you pass to the Status Monitor.

The following code example demonstrates how to execute the WF_G_ACTIVITIES

function using FND_FUNCTION.EXECUTE.

```
itemType := icx_call.encrypt('<your_item_type>');

itemKey := icx_call.encrypt('<your_item_key>');

adminMode := wf_fwkmon.isMonitorAdministrator('<user_name>');

wm := wf_fwkmon.getEncryptedAdminMode(adminMode);

wa := wf_fwkmon.getEncryptedAccessKey('<your_item_type>',
   '<your_item_key>', adminMode);

FND_FUNCTION.EXECUTE(
   FUNCTION_NAME => 'WF_G_ACTIVITIES',
   OPEN_FLAG => 'Y',
   SESSION_FLAG => 'Y',
   OTHER_PARAMS =>
     'itemType='||(wfa_html.conv_special_url_chars(itemType))
     ||'&'||'itemKey='||(wfa_html.conv_special_url_chars(itemKey))
     ||'&'||'wm='||(wfa_html.conv_special_url_chars(wm))
     ||'&'||'wa='||(wfa_html.conv_special_url_chars(wa))
     ||'&'||'retainAM=Y'
     ||'&'||'fExt=X'
   );
```

See: Overview of Form Development Steps, *Oracle Applications Developer's Guide* and Menus Window, *Oracle Applications Developer's Guide*.

Menu Setup

If you use a guest access function within your responsibility, you must add the menu containing that function to the top-level menu for your responsibility. The WF_G_ACTIVITIES and WF_G_DIAGRAM functions are seeded on the Workflow Guest Monitor Application (WF_G_MONITOR_APPLICATION) menu, and the WF_SSG_ACTIVITIES and WF_SSG_DIAGRAM functions are seeded on the Workflow Guest Self-Service Monitor Application (WF_SSG_MONITOR_APPLICATION) menu.

> **Note:** You cannot add the Status Monitor functions to your menu directly. To include these functions, you must add the Oracle Workflow menu that contains the function you want.

The Workflow Guest Monitor Application menu is seeded on the Workflow Administrator (New) menu (FND_WFADMIN_NEW) associated with the Workflow Administrator Web (New) responsibility. Similarly, the Workflow Guest Self-Service Monitor Application menu is seeded on the Workflow User (New) menu (FND_WFUSER_NEW) associated with the Workflow User Web (New) responsibility.

Migrating to Guest Access Functions

If your application used self-service functions to provide access to the previous version of the Workflow Monitor, you can migrate to the new Status Monitor by replacing the functions you previously called with the appropriate new functions. The following table shows the correspondence between functions for the previous Workflow Monitor and

the guest access functions for the new Status Monitor.

Migrating to Guest Access Functions

Previous Function	New Function
FND_WFMON_ADV	WF_G_ACTIVITIES
FND_WFMON_DIAGRAM	WF_G_DIAGRAM or WF_SSG_DIAGRAM
FND_FNDWFIAS (if called with the ITEM_TYPE, ITEM_KEY, ADMIN_MODE, and ACCESS_KEY parameters)	WF_SSG_ACTIVITIES

Related Topics

Migrating to Standard Access Functions, page 5-25

Testing Status Monitor Access

You can use the Workflow Monitor Test Application to test the monitor access you want to provide from your application. The Programmatic Access page simulates calls to obtain URLs for navigating from an Oracle Application Framework-based Web page to a Status Monitor page or calls to redirect from an Oracle Application Framework-based Web page to a Status Monitor page. You can retrieve a test URL or perform a trial redirection. The Programmatic Access page lets you test access to both the administrator and the self-service versions of the Status Monitor, using standard or guest access.

The Advanced Worklist page is also included in the Workflow Monitor Test Application to let you test access from a workflow notification to the Status Monitor. You can test links from notifications to the administrator version of the Status Monitor, using guest access. In the Worklist, navigate to a notification that contains a link to the Status Monitor, and select that link to test the access it provides. See: To View Notifications from the Advanced Worklist, *Oracle Workflow User's Guide*.

Menu Setup

The Workflow Monitor Test Application is available in Oracle Applications, but it is not seeded on any Oracle Applications menu. Before you can use this module, your system administrator must add its menu to a top-level menu for a responsibility. The menu for the Workflow Monitor Test Application module is named Workflow Monitor Test Application (WFMON_TEST_APPLICATION). For example, you can add this menu to the Workflow Administrator (New) menu (FND_WFADMIN_NEW), which is associated with the Workflow Administrator Web (New) responsibility, or to the

Workflow User (New) menu (FND_WFUSER_NEW), which is associated with the Workflow User Web (New) responsibility.

Additionally, when you navigate from the Workflow Monitor Test Application module to the Status Monitor using standard or guest access, you must perform the navigation from within a responsibility that also includes the menu for the Status Monitor version and access option you want to test. Otherwise, you will encounter function security errors. To avoid these errors, ensure that the appropriate Oracle Workflow menus are added to the same top-level menu as the WFMON_TEST_APPLICATION menu.

The menus for the Status Monitor are:

- Workflow Administrator Application (WF_ADMINISTRATOR_APPLICATION) - Required for standard access to the Administrator Monitor

- Workflow Guest Monitor Application (WF_G_MONITOR_APPLICATION) - Required for guest access to the Administrator Monitor

- Workflow Self-Service Application (WF_SELF_SERVICE_APPLICATION) - Required for standard access to the Self-Service Monitor

- Workflow Guest Self-Service Monitor Application (WF_SSG_MONITOR_APPLICATION) - Required for guest access to the Self-Service Monitor

    Note: The Workflow Administrator (New) menu includes the Workflow Administrator Application and Workflow Guest Monitor Application menus by default. Additionally, the Workflow User (New) menu includes the Workflow Self-Service Application and Workflow Guest Self-Service Monitor Application menus by default.

1. Testing Standard Access, page 5-35

2. Testing Guest Access, page 5-37

Related Topics

Accessing the Administrator Monitor, page 5-1

Accessing the Self-Service Monitor, *Oracle Workflow User's Guide*

Overview of Function Security, *Oracle Applications System Administrator's Guide*

Overview of Menus and Function Security, *Oracle Applications Developer's Guide*

Testing Standard Access

You can test standard access from an Oracle Application Framework-based Web page

to the Status Monitor in the Programmatic Access page. Standard access provides loosely coupled access to a Status Monitor page within the full Oracle Workflow application.

To Test Standard Access:

1. Use a Web browser to log on to a responsibility specified by your system administrator and choose the Programmatic Access link.

2. Select the Test "Standard" Access link.

3. In the Test "Standard" Access region, enter the test options you want. Each option corresponds to a method parameter for the method in the `oracle.apps.fnd.wf.monitor.webui.Monitor` class that is being simulated. See: Standard Access in Java, page 5-23.

   The test options include:

   - Item Type - Optionally enter the internal name of a workflow item type to automatically query in the Status Monitor.

   - Item Key - Optionally enter an item key to automatically query in the Status Monitor.

   - First Page - Specify the Status Monitor page that you want to initially display.

     - `MAIN` - Main Workflows search page

     - `HISTORY` - Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor

     - `DIAGRAM` - Status Diagram page

     - `RESPONSES` - Monitor Responses page

       > **Note:** For access to the `HISTORY`, `DIAGRAM`, or `RESPONSES` pages, you must specify an item type and item key. If you specify an item type and item key but do not specify a first page, the First Page option defaults to `HISTORY`.

   - Retain Calling AM - Specify `true` or `false` to indicate whether to retain the calling application module when calling a Get URL method. If you enter `true`, `retainAM=Y` is appended to the URL; if you enter `false`, `retainAM=N` is appended to the URL.

4. Select the test action you want to perform.

- Get Advanced Monitor URL - Corresponds to a call to the Java method *Monitor.getAdvanceUrl( )*. This call returns a URL for the Administrator Monitor page specified by the test options in the following format, suitable for use within an Oracle Application Framework application page:

  `/OA_HTML/OA.jsp?OAFunc=[parameters...]`

  For example, you can set this URL as a destination link on an `OAWebBean` using the call `<OAFrameworkBean>.setDestination(String url)`.

  The parameter `retainAM=Y` or `retainAM=N` is appended to the URL depending on the value you specified for the Retain Calling AM option.

  When you perform this action, Oracle Workflow retrieves a URL according to the test options using *Monitor.getAdvanceUrl( )* and displays that URL as a hyperlink in a text message bean. To navigate to the Status Monitor, select the link.

- Get Simple Monitor URL - Corresponds to a call to the Java method *Monitor.getSimpleUrl( )*. This call returns a URL for the Self-Service Monitor page specified by the test options in the following format, suitable for use within an Oracle Application Framework application page:

  `/OA_HTML/OA.jsp?OAFunc=[parameters...]`

  For example, you can set this URL as a destination link on an `OAWebBean` using the call `<OAFrameworkBean>.setDestination(String url)`.

  The parameter `retainAM=Y` or `retainAM=N` is appended to the URL depending on the value you specified for the Retain Calling AM option.

  When you perform this action, Oracle Workflow retrieves a URL according to the test options using *Monitor.getSimpleUrl( )* and displays that URL as a hyperlink in a text message bean. To navigate to the Status Monitor, select the link.

- Redirect to Advanced Monitor - Corresponds to a call to the Java method *Monitor.redirectToAdvanceUrl( )*. When you perform this action, Oracle Workflow redirects you to the Status Monitor as specified by the test options.

- Redirect to Simple Monitor - Corresponds to a call to the Java method *Monitor.redirectToSimpleUrl( )*. When you perform this action, Oracle Workflow redirects you to the Status Monitor as specified by the test options.

5. Select the Perform "Standard" Action button.

Testing Guest Access

You can test guest access from an Oracle Application Framework-based Web page to the Status Monitor in the Programmatic Access page. Guest access provides tightly

coupled access to a freestanding Status Monitor page for a particular workflow, without access to the rest of the Oracle Workflow application.

To Test Guest Access:

1. Use a Web browser to log on to a responsibility specified by your system administrator and choose the Programmatic Access link.

2. Select the Test "Guest" Access link.

3. In the Test "Guest" Access region, enter the test options you want. Each option corresponds to a method parameter for the method in the `oracle.apps.fnd.wf.monitor.webui.Monitor` class that is being simulated. See: Guest Access in Java, page 5-27.

   The test options include:

   - Item Type - Enter the internal name of the workflow item type to automatically query in the Status Monitor.

   - Item Key - Enter the item key to automatically query in the Status Monitor.

   - First Page - Specify the Status Monitor page that you want to initially display.

     - `HISTORY` - Activity History page in the Administrator Monitor or Notification History page in the Self-Service Monitor

     - `DIAGRAM` - Status Diagram page

     - `RESPONSES` - Monitor Responses page

     If you leave the First Page field blank, this option defaults to `HISTORY`.

   - Retain Calling AM - Specify `true` or `false` to indicate whether to retain the calling application module when calling a Get URL method. If you enter `true`, `retainAM=Y` is appended to the URL; if you enter `false`, `retainAM=N` is appended to the URL.

   - Administrator Mode - Specify a value to indicate whether the user should have privileges to perform administrative operations when accessing the Status Monitor.

     - `Y` - The user is granted administrator privileges, regardless of whether the user belongs to the workflow administrator role or not.

     - `N` - The user is not granted administrator privileges, regardless of whether the user belongs to the workflow administrator role or not.

     - `U` - Oracle Workflow checks whether the user belongs to the workflow

administrator role as specified in the Workflow Configuration page and grants administrator privileges accordingly.

If you leave the Administrator Mode field blank, this option defaults to `N`.

4. Select the test action you want to perform.

- Get Advanced Monitor URL - Corresponds to a call to the Java method *Monitor.getGuestAdvanceUrl( )*. This call returns a URL for the Administrator Monitor page specified by the test options in the following format, suitable for use within an Oracle Application Framework application page:

  `/OA_HTML/OA.jsp?OAFunc=[`*parameters...*`]`

  For example, you can set this URL as a destination link on an `OAWebBean` using the call `<OAFrameworkBean>.setDestination(String url)`.

  The parameter `retainAM=Y` or `retainAM=N` is appended to the URL depending on the value you specified for the Retain Calling AM option.

  When you perform this action, Oracle Workflow retrieves a URL according to the test options using *Monitor.getGuestAdvanceUrl( )* and displays that URL as a hyperlink in a text message bean. To navigate to the Status Monitor, select the link.

- Get Simple Monitor URL - Corresponds to a call to the Java method *Monitor.getGuestSimpleUrl( )*. This call returns a URL for the Self-Service Monitor page specified by the test options in the following format, suitable for use within an Oracle Application Framework application page:

  `/OA_HTML/OA.jsp?OAFunc=[`*parameters...*`]`

  For example, you can set this URL as a destination link on an `OAWebBean` using the call `<OAFrameworkBean>.setDestination(String url)`.

  The parameter `retainAM=Y` or `retainAM=N` is appended to the URL depending on the value you specified for the Retain Calling AM option.

  When you perform this action, Oracle Workflow retrieves a URL according to the test options using *Monitor.getGuestSimpleUrl( )* and displays that URL as a hyperlink in a text message bean. To navigate to the Status Monitor, select the link.

- Redirect to Advanced Monitor - Corresponds to a call to the Java method *Monitor.redirectToGuestAdvUrl( )*. When you perform this action, Oracle Workflow redirects you to the Status Monitor as specified by the test options.

- Redirect to Simple Monitor - Corresponds to a call to the Java method *Monitor.redirectToGuestSimpleUrl( )*. When you perform this action, Oracle Workflow redirects you to the Status Monitor as specified by the test options.

5. Select the Perform "Guest" Action button.

Status Monitor Portlets

Oracle Workflow provides two Status Monitor portlets that you can add to an Oracle Portal page to view summaries of workflow status information. The Error Workflows portlet provides administrator information about errored workflows, while the Workflows portlet provides self-service information about workflows owned by a user.

Adding a Portlet to an Oracle Portal Page

Before you can add an Oracle Application Framework portlet to an Oracle Portal page, your system administrator must assign you a responsibility that includes the menu function for that portlet on its menu. The Error Workflows portlet is seeded in the Workflow Administrator Web (New) responsibility, and the Workflows portlet is seeded in the Workflow User Web (New) responsibility. The system administrator may assign you either one of these responsibilities or another responsibility to which the appropriate portlet has been added.

After a responsibility that includes an Oracle Workflow portlet has been assigned to you, you can use Oracle Portal to add that portlet to an Oracle Portal page. In the Portlet Repository, select `Oracle Application Framework Provider` under `Other Providers` to access the Oracle Workflow portlets. For more information, please refer to the Oracle Portal online help, or see: Adding a Portlet to a Page, *Oracle Application Server Portal Tutorial*.

Error Workflows Portlet

The Error Workflows portlet provides administrator information about errored workflows. The portlet displays all errored workflows started within the last week. If you have workflow administrator privileges, the list shows workflows owned by any user; if you do not have workflow administrator privileges, the list shows only workflows that you own. You can select the item key link in the Item Key column to view the activity history for a workflow in the Activity History page of the Administrator Monitor.

Obtaining Access to the Error Workflows Portlet

The menu function for the Error Workflows portlet is WF_ADMIN_STATUS_PORTLET. This function is seeded on the Workflow Administrator (New) menu (FND_WFADMIN_NEW), which is associated with the Workflow Administrator Web (New) responsibility. To give you access to the Error Workflows portlet, your system administrator must perform one of the following steps:

• Assign you the Workflow Administrator Web (New) responsibility

- Add the FND_WFADMIN_NEW menu to the menu for another existing or custom responsibility and assign you that responsibility

- Add the WF_ADMIN_STATUS_PORTLET function to another existing or custom menu, associate that menu with a responsibility, and assign you that responsibility

Workflows Portlet

The Workflows portlet provides self-service information about workflows that you own. The portlet displays all workflows owned by you that were started within the last two weeks. You can select the workflow identifier link in the Workflow column to view the notification history for a workflow in the Notification History page of the Self-Service Monitor. The workflow identifier is the user key if one is specified for the workflow, or the item key if no user key is specified.

Obtaining Access to the Workflows Portlet

The menu function for the Workflows portlet is WF_SS_STATUS_PORTLET. This function is seeded on the Workflow User (New) menu (FND_WFUSER_NEW), which is associated with the Workflow User Web (New) responsibility. To give you access to the Error Workflows portlet, your system administrator must perform one of the following steps:

- Assign you the Workflow User Web (New) responsibility

- Add the FND_WFUSER_NEW menu to the menu for another existing or custom responsibility and assign you that responsibility

- Add the WF_SS_STATUS_PORTLET function to another existing or custom menu, associate that menu with a responsibility, and assign you that responsibility

6

Administering Notifications

This chapter describes how to administer users' notifications.

This chapter covers the following topics:

- Searching for Users' Notifications
- Defining Vacation Rules for Users
- Reviewing Electronic Signature Details
- Defining Specialized Worklist Views with Worklist Flexfields
- Embedding the Personal Worklist in an Oracle Application Framework Page

Searching for Users' Notifications

Use the Notifications administrator page to search for and access notifications sent to your users.

> **Note:** You must have workflow administrator privileges to access other users' notifications in the Notifications page. If you do not have administrator privileges, you can only search for and access your own notifications. Workflow administrator privileges are assigned in the Workflow Configuration page. See: Setting Global User Preferences, page 2-9.

> **Note:** If you want users to be able to search for their notifications through this page, you can give them access to this page from any responsibility you choose. See: Adding Worklist Functions to User Responsibilities, page 2-136.

To search for notifications:

1. Navigate to the Notifications page by choosing the Administration tab in the Oracle Workflow administrator Web pages and choosing Notification Search in the horizontal navigation.

2. Search for the notifications you want to access. The following search criteria are available only if you have workflow administrator privileges:

   - Notification ID - Enter the numerical notification ID for a specific notification. Note that if you specify a notification ID, all other search criteria are ignored.

   - Owner - Select the role that owns the notifications.

   - To - Select the role to which the notifications were sent.

     Note: Usually, the Owner role and the To role for a notification are the same. However, you can specify different roles in the Owner field and the To field to search for notifications that were reassigned in Delegate mode. See: Setting the WF: Notification Reassign Mode Profile Option, page 2-137.

   The following search criteria are always available:

   - From - Select the role from which the notifications were sent. The From role for a notification is determined by the #FROM_ROLE message attribute. See: #FROM_ROLE Attribute, *Oracle Workflow Developer's Guide*.

   - Status - Select the status of the notifications. You can search for notifications that are open, closed, or canceled, or to which an invalid reply was submitted, or choose All to display notifications in any status.

   - Workflow Type - Select the workflow item type to which the notifications belong. The display name for the workflow type you select populates the Workflow Type field, and the internal name for the workflow type you select populates the Type Internal Name field.

   - Type Internal Name - Enter the internal name of the workflow type to which the notifications belong, if you want to enter the internal name directly instead of selecting a value.

   - Subject - Enter the subject line of the notifications. This field is case-sensitive. You can use the percent sign (%) as a wildcard character to search for a partial subject line value.

   - Sent - Choose Today, This Week (last seven days), Last 2 Weeks (last fourteen

days), Last 30 Days, Last 60 Days, or Any Time to specify when the notifications were sent. All the sent date ranges include the current date; for example, Last 2 Weeks includes today as well as the previous thirteen days.

- Due Date - Choose Last 2 Weeks (last fourteen days), This Week (last seven days), Today, Next 2 Weeks (next fourteen days), Next 30 Days, Next 60 Days, or Any Time to specify when the notifications should be completed. All the due date ranges include the current date; for example, Next 2 Weeks includes today as well as the next thirteen days.

- Priority - Select High, Normal, or Low as the notification priority, or choose All to display notifications of any priority.

> **Note:** If you have workflow administrator privileges, you must enter at least one of the following criteria when you search in order to limit the size of the results list.
>
> - Notification ID
>
> - Owner
>
> - To
>
> - From

3. To view and respond to a notification, select the notification subject link in the Subject column. See: To View the Details of a Notification, *Oracle Workflow User's Guide*.

> **Note:** If you take action on another user's notification, such as responding to, reassigning, or closing the notification or requesting more information, then Oracle Workflow updates the notification's action history to show that you performed that action.

> **Note:** To respond to a notification that requires an electronic signature, you must be logged in as the intended recipient. You cannot respond to a notification that requires a signature when you are accessing another user's notifications, because in this case the user responding will not match the intended recipient, so authorization will fail.
>
> However, if you need to act on another user's notifications for administrative reasons, you can reassign notifications that require signatures to yourself, and then submit responses to those

notifications using your own signature.

Defining Vacation Rules for Users

Use vacation rules to handle notifications automatically when users are not available to manage their notifications directly, such as when they are on vacation. You can define rules to perform the following actions automatically when a notification arrives:

- Reassign the notification to another user

    Note: A workflow may include special logic to restrict reassignment of notifications. If so, Oracle Workflow does not apply any vacation rules to reassign those notifications, but simpy delivers the notifications to the worklist of the original recipient. See: #HIDE_REASSIGN Attribute, *Oracle Workflow Developer's Guide*
    .

- Respond to the notification with a predefined response, or close a notification that does not require a response

    Note: If this notification requires a password-based signature or certificate-based digital signature in the response, you cannot respond to the notification through a vacation rule. In this case Oracle Workflow simply delivers the notification to the worklist of the original recipient. The recipient must use the Notification Details page to submit a response and provide the required signature. See: To View the Details of a Notification, *Oracle Workflow User's Guide*.

- Deliver the notification to the original recipient's worklist as usual, with no further action

Use the Vacation Rules administrator page to define rules for automatic notification processing for your users. Each rule is specific to a role.

> **Note:** You must have workflow administrator privileges to define vacation rules for other users in the Vacation Rules page. Workflow administrator privileges are assigned in the Workflow Configuration page. See: Setting Global User Preferences, page 2-9.

A vacation rule can apply to messages of all item types, to all messages belonging to a specific item type, or to a specific type of message belonging to a specific item type.

Each time a notification is sent to a user, Oracle Workflow tests the notification against that user's vacation rules. First Oracle Workflow checks whether the user has any active rules for that specific message type. If not, it checks whether the user has any active rules for that specific item type. Finally, it checks whether the user has any active rules for messages of all item types. As soon as it finds a match, Oracle Workflow applies the rule and discontinues any further rule matching.

If a rule reassigns a notification, Oracle Workflow performs rule matching again against the new recipient role's list of rules. Oracle Workflow maintains a count of the number of times it forwards a notification to detect perpetual forwarding cycles. If a notification is automatically forwarded more than ten times, Oracle Workflow assumes that a forwarding cycle has occurred and ceases executing any further forwarding rules, marking the notification as being in error.

To define vacation rules for a user:

1. Navigate to the Vacation Rules page by choosing the Administration tab in the Oracle Workflow administrator Web pages and choosing Vacation Rules in the horizontal navigation.

2. Search for the role for which you want to define vacation rules.

   The list of existing rules for the selected role includes rules defined by the individual user as well as rules defined by an administrator for that user. A rule's active or inactive status depends on whether the current date falls within the rule's effective dates.

3. To update a rule, select the Update icon for that rule. See: To Create or Update a Vacation Rule, *Oracle Workflow User's Guide*.

4. To delete a rule, select the Delete icon for that rule.

5. To create a new rule, select the Create Rule button. See: To Create or Update a Vacation Rule, *Oracle Workflow User's Guide*.

Related Topics

Vacation Rules, *Oracle Workflow User's Guide*

Setting Up Notification Handling Options, page 2-138.

Reviewing Electronic Signature Details

A notification can require that a user's response be signed electronically, either with a password-based signature or with a certificate-based digital signature. Use the Signature Evidence Store to review details about the electronic signatures requested or submitted for notifications that require signatures. For example, you can check the status of a signature request and review details that provide evidence of the signature.

You can also view some signature details in the Notification Response Details page in the Status Monitor. See: Viewing Responses, page 5-12.

You can only review signatures for notifications that have not been purged. To preserve electronic signature evidence for future reference, the Purge Obsolete Workflow Runtime Data concurrent program and the Oracle Workflow purging APIs by default do not delete any notifications that required signatures or their associated signature information. If you anticipate needing access to signature evidence after the associated workflow processes are complete, ensure that you choose to preserve signature data when purging. If you do not need to maintain signature evidence, you can choose to delete signature-related information when purging. See: Purging Workflow Data, page 7-81 and Workflow Purge APIs, *Oracle Workflow API Reference*.

To review electronic signature details:

1. Navigate to the Electronic Signature page by choosing the Administration tab in the Oracle Workflow administrator Web pages and choosing Signature Evidence Store in the horizontal navigation.

2. Search for the signatures you want to review. The search criteria are:

    - Notification ID - Enter the numerical notification ID for a specific notification that requires a signature. Note that if you specify a notification ID, all other search criteria are ignored.

    - Signature Policy - Select the policy that identifies the type of signature and signed text required by a notification, or select Any to display signatures for any policy.

        - `PSIG_ONLY` - This policy requires a password-based signature for signed response text that contains only the notification header information and the response values entered by the user.

        - `PSIG_BODY` - This policy requires a password-based signature for signed response text that contains the notification header information, the response values entered by the user, and the notification message body.

        - `PKCS7X509_ONLY` - This policy requires a certificate-based digital signature for signed response text that contains only the notification header information and the response values entered by the user.

        - `PKCS7X509_BODY` - This policy requires a certificate-based digital signature for signed response text that contains the notification header information, the response values entered by the user, and the notification message body.

    - Requested Signer - Select the user to whom the notification requesting a signature was sent.

- Status - Select the status of the signatures you want to review, or select Any to display signatures in any status.

  - Requested - Oracle Workflow has requested a signature from a user by sending a notification that requires a signature.

  - Signed - The user has submitted a signature with the notification response.

  - Verified - Oracle Workflow has verified that the signature was well formed, that it was created with a private key corresponding to the offered signing certificate, and that it is signing the plain text that it purports to sign.

  - Authorized - Oracle Workflow has confirmed that the user who submitted the signature is authorized to sign the notification by checking that the certificate is assigned to a user who is a member of the recipient role for the notification.

  - Validation Attempted - Oracle Workflow has attempted to confirm that the certificate used to create the signature was valid at the time the signature was received, meaning it had not expired or been revoked. To validate a certificate, Oracle Workflow checks that the certificate does not appear on a certificate revocation list (CRL) issued by the certificate authority after the time the signature was received. If the currently available CRL was issued before the time the signature was received, Oracle Workflow sets the signature status to Validation Attempted and checks the CRL again later.

  - Validated - Oracle Workflow has successfully validated the signature against a CRL issued by the certificate authority after the time the signature was received.

  - Request Failed - The request for a signature was not successfully created. An error may have occurred in notification processing.

  - Signature Failed - The user attempted to submit a signature but did not successfully complete the signature.

  - Signature Cancelled - The user or an administrator canceled the submitted signature. The notification may have been canceled.

  - Verification Failed - Oracle Workflow could not verify the signature because the signature was not well formed, it was not created with a private key corresponding to the offered signing certificate, or it did not sign the plain text that it purported to sign.

  - Authorization Failed - Oracle Workflow could not confirm that the user who submitted the signature was authorized to sign the notification, because the certificate used to create the signature was not assigned to a

user who was a member of the recipient role for the notification.

- Validation Failed - Oracle Workflow could not confirm that the certificate used to create the signature was valid at the time the signature was received. The certificate may have been expired or revoked.

- Error - Oracle Workflow encountered an error in signature processing.

- Creation Date - Enter the date when the request for a signature was created.

- Signed Date - Enter the date when the user submitted the signature.

- Verified Date - Enter the date when Oracle Workflow confirmed that the signature was well formed, that it was created with a private key corresponding to the offered signing certificate, and that it signed the plain text that it purported to sign.

- Last Validation Date - Enter the most recent date when Oracle Workflow attempted to check that the certificate used to create the signature was valid at the time the signature was received.

- Validated Complete Date - Enter the date when Oracle Workflow successfully validated the signature against a CRL issued by the certificate authority after the time the signature was received.

> **Note:** You must enter at least one of the following criteria when you search in order to limit the size of the results list.
>
> - Notification ID
>
> - Requested Signer
>
> - Creation Date
>
> - Signed Date
>
> - Verified Date

3. Review the list of signatures that match your search criteria.

Related Topics

Setting Up for Electronic Signatures, page 2-140

#WF_SIG_POLICY Attribute, *Oracle Workflow Developer's Guide*

Electronic Signatures, *Oracle Workflow User's Guide*

Defining Specialized Worklist Views with Worklist Flexfields

Use worklist flexfields to define specialized worklist views that display information specific to particular types of notifications. Worklist flexfields are columns in the notification table, WF_NOTIFICATIONS, in which you can store information from different message attributes for different notifications. Storing information in the notification table lets you display that information in a Personal Worklist view. Through such a view, users can quickly review important details about several notifications at once, without navigating to the Notification Details page for each notification. Users can also sort their worklists by the displayed message attributes.

For example, if an expense notification includes message attributes for the expense total and purpose, an administrator can map these message attributes to worklist flexfields. The administrator can then define a worklist view that includes only expense notifications and displays the expense total and purpose for each notification alongside standard notification properties, such as the subject line and sent date.

> **Note:** Worklist flexfields are separate from the key flexfields and descriptive flexfields used in Oracle Applications. For information about key and descriptive flexfields, see the *Oracle Applications Flexfields Guide*.

Users must have access to the Personal Worklist to take advantage of a specialized worklist view. See: Adding Worklist Functions to User Responsibilities, page 2-136.

To define a specialized worklist view using worklist flexfields, perform these steps:

1. Define a worklist flexfields rule that maps message attributes from one or more workflow item types to worklist flexfields columns. See: Defining a Worklist Flexfields Rule, page 6-16.

2. Optionally define a securing function to secure access to your new worklist view. See: Defining a Securing Function, page 6-22.

3. Create a Personal Worklist view that displays the columns mapped by a worklist flexfields rule and that includes only notifications from the corresponding item types. See: Creating a Personalized View for the Personal Worklist, page 6-23.

4. If you secured access to the view with a securing function, restart Oracle HTTP Server. See: Restarting Oracle HTTP Server, page 6-24.

Some Oracle Applications products also provide seeded worklist flexfields rules and Personal Worklist views. For more information, consult your product-specific documentation or help.

Message Attributes in Worklist Flexfields Rules

Worklist flexfields rules can map both send and respond message attributes. If a respond message attribute has a default value, the corresponding worklist flexfield displays that value while the notification is open.

You can map message attributes with the following data types:

- Text

- Number

- Form

> **Note:** When displayed in a Personal Worklist view, a form attribute appears as a form icon that drills down to the specified Oracle Applications form or Oracle Self-Service Web Applications page. Users must use a responsibility with the appropriate security to open a linked form or Oracle Application Framework page.

- URL

> **Note:** When displayed in a Personal Worklist view, a URL attribute appears as a URL icon that links to the specified URL.

- Date

> **Note:** A message attribute of type date must have the following format to be used in a worklist flexfields rule:
>
> ```
> dd-mon-yyyy hh24:mi:ss
> ```
>
> The Personal Worklist requires date attributes to be stored in this format to display them properly according to each user's preference.

Worklist flexfields rules cannot map message attributes of type lookup, role, document, or event.

How Worklist Flexfields Rules Operate

Worklist flexfields rules make message attributes available for display in the Personal Worklist. A worklist flexfields rule operates in combination with other rules that cover the same item type. The phase numbers and customization levels of the rules determine how their effects combine to produce a net set of available message attributes.

- The phase numbers for worklist flexfields rules control the order in which rules for the same item type take effect. If two rules for the same item type map message attributes to the same worklist flexfield column, then the rule with the higher phase number overrides the rule with lower phase number in that column. Other column mappings from the lower-phase rule may still take effect if they do not conflict with any column mappings from higher-phase rules.

  Rules seeded by Oracle Applications use phase numbers from 1 to 99. You can assign your rules phase numbers of 100 or higher.

  > **Note:** Do not assign the same phase number to more than one rule for the same item type. To ensure that the rules you want take effect, assign a different phase number to each rule for an item type.

- The customization level for a rule determines which worklist flexfields columns the rule can map and whether you can update the rule definition. Oracle Workflow uses the customization level to protect Oracle Applications seed data and to preserve your customizations in an upgrade.

  Rules seeded by Oracle Applications can have a customization level of Core or Limit. For rules that you define, Oracle Workflow automatically sets the customization level to User.

Core rules represent key Oracle Applications features. You cannot make any changes to the rule definitions. Core rules use a different set of worklist flexfields columns than limit and user rules, so core rules cannot override or be overridden by limit and user rules. However, a core rule with a higher phase number can override a core rule for the same item type with a lower phase number. Core rules use phase numbers from 1 to 99.

Core rules can map the following worklist flexfields columns.

Worklist Flexfields Columns for Core Rules

Data Type	Column Names
Text	PROTECTED_TEXT_ATTRIBUTE1
	PROTECTED_TEXT_ATTRIBUTE2
	PROTECTED_TEXT_ATTRIBUTE3
	PROTECTED_TEXT_ATTRIBUTE4
	PROTECTED_TEXT_ATTRIBUTE5
	PROTECTED_TEXT_ATTRIBUTE6
	PROTECTED_TEXT_ATTRIBUTE7
	PROTECTED_TEXT_ATTRIBUTE8
	PROTECTED_TEXT_ATTRIBUTE9
	PROTECTED_TEXT_ATTRIBUTE10
Number	PROTECTED_NUMBER_ATTRIBUTE1
	PROTECTED_NUMBER_ATTRIBUTE2
	PROTECTED_NUMBER_ATTRIBUTE3
	PROTECTED_NUMBER_ATTRIBUTE4
	PROTECTED_NUMBER_ATTRIBUTE5
Form	PROTECTED_FORM_ATTRIBUTE1
	PROTECTED_FORM_ATTRIBUTE2
	PROTECTED_FORM_ATTRIBUTE3
	PROTECTED_FORM_ATTRIBUTE4
	PROTECTED_FORM_ATTRIBUTE5
URL	PROTECTED_URL_ATTRIBUTE1
	PROTECTED_URL_ATTRIBUTE2
	PROTECTED_URL_ATTRIBUTE3
	PROTECTED_URL_ATTRIBUTE4
	PROTECTED_URL_ATTRIBUTE5

Data Type	Column Names
Date	PROTECTED_DATE_ATTRIBUTE1
	PROTECTED_DATE_ATTRIBUTE2
	PROTECTED_DATE_ATTRIBUTE3
	PROTECTED_DATE_ATTRIBUTE4
	PROTECTED_DATE_ATTRIBUTE5

Usually, core rules map message attributes that appear in a seeded worklist view within a particular application. You can also include message attributes mapped by core rules in your own worklist views. If a core rule does not meet your requirements, you can choose not to include its message attributes in your views; however, Oracle Workflow still stores the message attributes in the mapped columns.

Limit rules represent optional Oracle Applications features. You can update the status of limit rules to Enabled or Disabled, but you cannot make any other changes to the rule definitions.

User rules are the custom rules that you define. You can update any property in the rule definitions.

> **Note:** You cannot delete worklist flexfields rules. If you no longer want a rule to take effect, disable the rule.

Limit rules and user rules share the same set of worklist flexfields columns. However, limit rules use phase numbers from 1 to 99, while user rules use phase numbers of 100 or higher. Consequently, a limit rule can be overridden both by another limit rule for the same item type with a higher phase number and by a user rule for the same item type. However, a user rule can only be overridden by another user rule for the same item type with a higher phase number. A limit rule cannot override a user rule.

Limit rules and user rules can map the following worklist flexfields columns.

Worklist Flexfields Columns for Limit and User Rules

Data Type	Column Names
Text	TEXT_ATTRIBUTE1
	TEXT_ATTRIBUTE2
	TEXT_ATTRIBUTE3
	TEXT_ATTRIBUTE4
	TEXT_ATTRIBUTE5
	TEXT_ATTRIBUTE6
	TEXT_ATTRIBUTE7
	TEXT_ATTRIBUTE8
	TEXT_ATTRIBUTE9
	TEXT_ATTRIBUTE10
Number	NUMBER_ATTRIBUTE1
	NUMBER_ATTRIBUTE2
	NUMBER_ATTRIBUTE3
	NUMBER_ATTRIBUTE4
	NUMBER_ATTRIBUTE5
Form	FORM_ATTRIBUTE1
	FORM_ATTRIBUTE2
	FORM_ATTRIBUTE3
	FORM_ATTRIBUTE4
	FORM_ATTRIBUTE5
URL	URL_ATTRIBUTE1
	URL_ATTRIBUTE2
	URL_ATTRIBUTE3
	URL_ATTRIBUTE4
	URL_ATTRIBUTE5

Data Type	Column Names
Date	DATE_ATTRIBUTE1
	DATE_ATTRIBUTE2
	DATE_ATTRIBUTE3
	DATE_ATTRIBUTE4
	DATE_ATTRIBUTE5

The message attributes mapped by limit rules may appear in a seeded worklist view within a particular application. You can also include these message attributes in your own worklist views. If a limit rule does not meet your requirements and you want to override some of its mappings while allowing others to take effect, define a user rule that maps the attributes you want to the relevant columns in place of the attributes you do not need. If you no longer want any mappings from a limit rule to take effect, disable that rule.

Define user rules for any message attributes you want that are not made available by default through core or limit rules. While defining a rule, you can check whether its column mappings conflict with any existing rules for the same item type, and whether the new rule will override or be overridden by the conflicting rules. Review each conflict to decide whether to accept the current overrides or update the rule definitions to make a different rule take effect. See: To Create or Update a Worklist Flexfields Rule, page 6-18.

When you finish defining a worklist flexfields rule, Oracle Workflow submits the Denormalize Worklist Flexfields concurrent program (FNDWFDCC) once for each workflow item type in the rule. The program stores the message attributes for any currently open notifications from that item type in the mapped columns, except for columns overridden by another rule. Subsequently, whenever Oracle Workflow sends a new notification from an item type covered by the rule, the Notification System stores the message attributes for the notification in the mapped columns.

To review the net set of message attributes that are currently available for a particular item type, or for a particular message within an item type, perform a worklist flexfields rules simulation. The simulation results also let you drill down to review any overridden rules for each column and create or update rules if necessary. After you are satisfied with the available attributes, use the simulation results to choose the columns to include when you create a Personal Worklist view. See: To Simulate the Effect of Worklist Flexfields Rules, page 6-21.

Example - Worklist Flexfields Rules
This example demonstrates how worklist flexfields rules operate, using the sample Requisition item type. For more details about this sample item type, see: The Requisition Item Type, *Oracle Workflow Developer's Guide*.

Suppose a core rule named EXC01 is seeded for the Requisition item type with a phase of 50 and the following column mapping:

- PROTECTED_TEXT_ATTRIBUTE1 column : Requisition Number attribute

Also, suppose a limit rule named EXL02 is seeded for the Requisition item type with a phase of 60 and the following column mappings:

- TEXT_ATTRIBUTE1 column : Requisition Description attribute

- NUMBER_ATTRIBUTE1 column : Requisition Amount attribute

You prefer to display the Note attribute instead of the Requisition Description attribute, and you also want to display the Monitor URL attribute. To do so, you define a user rule named EXU03 for the Requisition item type with a phase of 110 and the following column mappings:

- TEXT_ATTRIBUTE1 column : Note attribute

- URL_ATTRIBUTE1 column : Monitor URL attribute

These rules combine to produce a net set of four message attributes that you can display in a Personal Worklist view for the Requisition item type.

Example Set of Available Message Attributes

Column	Message Attribute	Effective Rule
PROTECTED_TEXT_ATTRIBUTE1	Requisition Number	EXC01
NUMBER_ATTRIBUTE1	Requisition Amount	EXL02
TEXT_ATTRIBUTE1	Note	EXU03
URL_ATTRIBUTE1	Monitor URL	EXU03

Defining a Worklist Flexfields Rule:

To View and Maintain Worklist Flexfields Rules

1. Use a Web browser to navigate to the Worklist Flexfields Rules page, using a responsibility and navigation path specified by your system administrator. See: Oracle Workflow Administrator Navigation Paths, page A-1.

2. Search for the rules you want to display. The search criteria are:

- Rule Name - Enter the internal name of the rule. You can enter a partial value to search for rules whose internal names contain that value. This field is case-sensitive.

- Display Name - Select the user-friendly name of the rule. You can enter a partial value to search for rules whose display names contain that value. This field is case-sensitive.

- Phase - Enter the phase number that determines whether the rule overrides or is overridden by other rules.

- Status - Select whether to display enabled rules or disabled rules.

- Level - Select the customization level of the rule.

- Workflow Type - Select a workflow item type that contains notifications to which the rule applies.

- Message Attribute - Select a message attribute that the rule maps to a worklist flexfields column. If you specified a workflow type, you can only select a message attribute belonging to that workflow type. Otherwise, you can select a message attribute belonging to any workflow type

- Column Name - Select a worklist flexfields column to which the rule maps a message attribute.

You must enter at least one of the following criteria when you search to limit the size of the results list.

- Rule Name

- Display Name

- Level

- Workflow Type

If you search only by the Level option or the Workflow Type option, you must select a specific value for that option. You cannot use one of these criteria with the Any value as your only search option.

The Workflow Type, Message Attribute, and Column Name search options only list values for which a rule exists.

3. To update a rule, choose the update icon for that rule. See: To Create or Update a Worklist Flexfields Rule, page 6-18.

4. To create a new rule, select the Create Rule button. See: To Create or Update a

Worklist Flexfields Rule, page 6-18.

To Create or Update a Worklist Flexfields Rule

To Enter General Properties

1. Navigate to the Create Worklist Flexfields Rule: Enter General Properties page or to the Update Worklist Flexfields Rule: Enter General Properties page. The Create Worklist Flexfields Rule pages and the Update Worklist Flexfields Rule pages are identical, except that the fields in the Update Worklist Flexfields Rule pages are populated with previously defined information for the selected rule.

2. Enter the internal name that uniquely identifies the rule and the user-friendly display name for the rule.

3. Review the customization level for the rule.

   - Core - You cannot make any changes to the rule definition. This level is used only for rules seeded by Oracle Applications.

   - Limit - You can update the rule status to Enabled or Disabled, but you cannot make any other changes to the rule definition. This level is used only for rules seeded by Oracle Applications.

   - User - You can update any property in the rule definition. This level is automatically set for rules that you define.

4. Select Enabled or Disabled as the rule status.

5. Enter a phase number for the rule to specify the order in which rules for the same workflow item type take effect. Rules with a higher phase number override rules with a lower phase number. Rules seeded by Oracle Applications use phase numbers from 1 to 99. You can assign your rules phase numbers of 100 or higher.

   > **Note:** Do not assign the same phase number to more than one rule for the same item type. To ensure that the rules you want take effect, assign a different phase number to each rule for an item type.

6. Enter an optional description for the rule.

7. Identify the application that owns the rule by entering the application name in the Owner Name field and the application ID in the Owner Tag field.

To Select Filter Criteria

8. Search for the workflow item types that contain the notifications to which the rule

should apply. You can enter a partial value in the Workflow Type field to search for item types whose display names begin with that value. This field is case-sensitive.

9. Select the item types you want in the Available Filter Criteria list and move them to the Selected Filter Criteria list.

   Select an item type in either list to view its description.

   If you perform a new search to show different item types in the Available Filter Criteria list, Oracle Workflow still preserves the item types that you already added to the Selected Filter Criteria list.

To Select Message Attributes

10. To remove a workflow item type from the filter criteria for the rule, choose the remove icon for that item type in the Selected Filter Criteria list.

    When you remove an item type, Oracle Workflow removes any message attributes belonging to that item type from the Available list and the Selected list.

11. Optionally specify the data type of the message attributes to display in the Available list.

12. Select the message attributes to map in the Available list and move them to the Selected list. You can select a maximum of ten text attributes, five number attributes, five form attributes, five URL attributes, and five date attributes.

    The lists show the display name and data type for each message attribute. Select a message attribute in either list to view in the Description field the display name and internal name of the message to which the attribute belongs.

    If multiple messages in the selected workflow item types have a message attribute with the same internal name, display name, and data type, that message attribute appears only once in the lists. In this case the Description field indicates that the message attribute occurs in multiple messages

    > **Note:** Oracle Workflow treats all message attributes with the same internal name and data type as the same attribute for purposes of worklist flexfields column mapping. Although attributes with different display names appear separately in the Available and Selected lists, if you select at least one attribute with a particular internal name and data type, all attributes that share that internal name and data type will be included in the column mapping.

    If you display message attributes of a different data type in the Available list, Oracle Workflow still preserves the message attributes that you already added to the Selected list.

To Map Attributes to Columns

13. For each message attribute, select the worklist flexfields column in which to store the attribute value. The Mapped Column field for each attribute displays only columns that match the attribute data type. You can use each column only once in a rule.

If multiple messages in the selected workflow item types have a message attribute with the same internal name and data type, that message attribute appears only once. However, if any of the repeated message attributes have different display names, the list shows all the display names.

14. To remove a message attribute from the column mappings for the rule, choose the remove icon for that attribute.

15. To review any conflicts with other rules' column mappings, choose the Find Conflicts button.

16. In the Find Worklist Flexfields Rule Map Conflicts page, review the columns that other rules map to different attributes for the same workflow item types. The Conflict field indicates whether the current rule overrides or is overridden by the other rule, based on the rule phase numbers.

> **Note:** Check that no two rules for the same item type have the same phase number. To ensure that the rules you want take effect, each rule for an item type must have a different phase number.

To resolve a conflict:

- If you want a worklist view to display the mapped attributes from all rules simultaneously, change the column mappings for one of the rules to use separate columns for the different attributes.

- If you no longer want any column mappings from a particular rule to take effect, disable that rule.

- If you want to override some column mappings from a particular rule while allowing others to take effect, either decrease the phase number for that rule or increase the phase number for the overriding rule.

- If the appropriate rules already override any others, accept the existing rule definitions.

17. To return to your rule definition, choose the Return to Pending Rule link.

18. If you need to change this rule definition to resolve conflicts, return to the previous pages to make your changes.

19. To complete the rule definition, choose Finish. Oracle Workflow submits the Denormalize Worklist Flexfields concurrent program (FNDWFDCC) once for each workflow item type in the rule. The program stores the message attributes for any currently open notifications from that item type in the mapped columns, except for columns overridden by another rule. Use the displayed concurrent request IDs to track the progress of these requests.

To Simulate the Effect of Worklist Flexfields Rules

A worklist flexfields rules simulation lets you review the net effect of the enabled rules for a particular item type, or optionally for a particular message. The simulation shows which message attributes are available in which worklist flexfields columns after all relevant rules are applied, with higher-phase rules overriding lower-phase rules wherever they conflict.

- Perform a Core Rules simulation to learn which message attributes are already available in worklist flexfields columns by default. Although you cannot change these rules, you can use the simulation results to avoid duplicating the mapped attributes in user rules.

- Perform a Limit and User Rules simulation to verify that any other message attributes you require are available in worklist flexfields columns. If necessary, you can adjust the rules' effect by enabling or disabling limit rules and creating, updating, enabling, or disabling user rules.

- After you are satisfied with the net set of message attributes available for the item type, use the simulation results to choose the columns to include when you create a Personal Worklist view.

1. Use a Web browser to navigate to the Worklist Flexfields Rules Simulation page, using a responsibility and navigation path specified by your system administrator. See: Oracle Workflow Administrator Navigation Paths, page A-1.

2. Specify the criteria for the simulation.
   - Select the workflow item type to review.
   - Optionally select a message within that item type to review.
   - Select the customization level to review, either core rules or limit and user rules together.

3. Choose Go to perform the simulation.

4. Review the list of mapped attributes and columns, which shows the net effect of the enabled rules for the selected item type and customization level. If you specified a message, the list shows only mapped attributes belonging to that message.

5. The list displays only the rule that takes effect for each column. To view any overridden rules that also map to that column, choose the overridden rules icon.

6. In the Find Worklist Flexfields Rule Conflicts page, review the columns that overridden rules map to different attributes than the effective rule. The Conflict field indicates that the effective rule overrides the other rules, based on the rule phase numbers.

> **Note:** Check that no two rules for the same item type have the same phase number. To ensure that the rules you want take effect, each rule for an item type must have a different phase number.

To resolve a conflict:

- If you want a worklist view to display the mapped attributes from all rules simultaneously, change the column mappings for one of the rules to use separate columns for the different attributes.

- If you no longer want any column mappings from a particular rule to take effect, disable that rule.

- If you want to override some column mappings from a particular rule while allowing others to take effect, either decrease the phase number for that rule or increase the phase number for the overriding rule.

- If the appropriate rules already override any others, accept the existing rule definitions.

7. To return to the simulation results, choose the Return to Worklist Flexfields Rule Simulation link.

8. To update a rule, choose the update icon for that rule. See: To Create or Update a Worklist Flexfields Rule, page 6-18.

9. To create a new rule, select the Create Rule button. See: To Create or Update a Worklist Flexfields Rule, page 6-18.

Defining a Securing Function:

If you want to secure access to your specialized worklist view, define a securing function to associate with the view. Use the Form Functions window in Oracle Applications (System Administrator: Application > Function) to define a function with the following properties:

- Type: Subfunction

- Context Dependence: Responsibility

Add both the securing function and the Personal Worklist function (
`WF_WORKLIST_CUSTOM`) to a menu, and assign the responsibility associated with that menu to your users. Your specialized worklist view will appear in the list of views only when users access the Personal Worklist from that responsibility.

You can associate several worklist views with the same securing function if you want those views to be available through the same responsibility.

If you do not secure your specialized worklist view, it will appear in the list of views whenever any user accesses the Personal Worklist from any responsibility.

See: Overview of Function Security, *Oracle Applications System Administrator's Guide*, Overview of Menus and Function Security, *Oracle Applications Developer's Guide*, Form Functions Window, *Oracle Applications Developer's Guide*, and Adding Worklist Functions to User Responsibilities, page 2-136.

Creating a Personalized View for the Personal Worklist:

Use Oracle Application Framework Personalization to create a Personal Worklist view that displays the worklist flexfields to which you mapped message attributes. To make the view available to your users, create it as an administrator-seeded user-level personalization.

Worklist flexfields rules are connected to a Personal Worklist view through the item types they both reference. You do not need to associate rules directly with a view. Rather, to display meaningful information in a view, limit the view to include only related item types that are covered by worklist flexfields rules, and add only worklist flexfields columns that are mapped by those rules.

For more information about creating personalizations, see: Personalization, *Oracle Application Framework Personalization Guide*.

1. Log in as a user for whom the Personalize Self-Service Defn profile option is set to Yes.

2. Navigate to the Personal Worklist and choose the Personalize Page global link or the Personalize region link for the "Customizable and searchable worklist" region. In the Page Hierarchy Personalization page, select the Seeded User Views icon for the "Table: Customizable and ..." item. In the Personalize Views page, select the Create View button.

3. In the Create View page, define the properties of your view.

   • If you defined a securing function for the view, specify the function in the view's general properties.

   • Select the columns to display in the view, including the worklist flexfields columns that are mapped to message attributes. Rename the columns as appropriate to identify the attributes stored in them.

     A view can include worklist flexfields columns that are mapped by different

rules. However, to display meaningful values in the view, you should select columns that are mapped by rules for the same item types.

> **Note:** Perform a worklist flexfields rules simulation to review the worklist flexfields column mappings that are in effect for a particular item type. See: To Simulate the Effect of Worklist Flexfields Rules, page 6-21.

- Specify the columns by which to sort the view, including worklist flexfields columns if appropriate.

- To display only notifications from the relevant item types, add the Type parameter or the Type Internal Name parameter to the search query for the view, and specify the item type display name or internal name, respectively. You can add multiple instances of these parameters to include multiple item types in the view. In this case, select the `Search results where each may contain any value entered` option.

  If you do not limit the item types included in the view, then the view may display notifications from other item types with blank values or unrelated values in the worklist flexfields columns.

For more information, see: Create View, Update View, and Duplicate View Pages, *Oracle Application Framework Personalization Guide*.

Restarting Oracle HTTP Server:

If you associated a securing function with the view, you must stop and restart Oracle HTTP Server after completing the menu updates and saving the view, to make the changes take effect. See: Administering Oracle HTTP Server Powered by Apache, *Oracle Applications System Administrator's Guide - Configuration*.

Users must then access the Personal Worklist through a responsibility that includes the securing function for the view to appear in their list of available views.

Embedding the Personal Worklist in an Oracle Application Framework Page

If you have Oracle Application Framework set up in Oracle JDeveloper for custom development, you can embed the Personal Worklist as a region in an Oracle Application Framework page. In this way you can provide users access to the worklist from within your own application.

For more information about building and personalizing an Oracle Application Framework page, refer to Oracle*MetaLink* note 391554.1, *Oracle Application Framework Documentation Resources, Release 12*.

To Embed the Personal Worklist in an Oracle Application Framework Page:

1. Create the Oracle Application Framework region definition for the top-level page in which you want to embed the Personal Worklist.

   The Personal Worklist and all its subsidiary pages use their own application modules and controllers. However, you must use a separate application module and controller for your top-level page.

2. Define page features specific to your application, including branding, headers, and footers. The subsidiary pages accessed from the Personal Worklist will have the same features you define here for the top-level page.

3. Add the Personal Worklist by creating a new region in the page and setting the `Extends` property to the following value:

   `/oracle/apps/fnd/wf/worklist/webui/FullWorklistPG.FullWorklistRN`

   The Personal Worklist code automatically provides a link to let users return from the worklist pages to your application.

4. To personalize the embedded Personal Worklist region, create a personalization for the top-level page at function level or at responsibility level. For example, you can filter the worklist to display only notifications from item types owned by your application, or use a specialized view to display message attributes specific to your application.

7

Oracle Workflow Manager

This chapter describes how to use the Oracle Workflow Manager component of Oracle Applications Manager.

This chapter covers the following topics:

- Oracle Workflow Manager Overview
- Service Components
- Notification Mailers
- Agent Listeners
- Java Agent Listeners
- Web Services Outbound
- Background Engines
- Purging Workflow Data
- Workflow Control Queue Cleanup
- Active Work Items
- Deferred Work Items
- Suspended Work Items
- Errored Work Items
- Agents
- Queue Propagation

Oracle Workflow Manager Overview

Oracle Workflow Manager is a component of Oracle Applications Manager that allows system administrators to manage Oracle Workflow for multiple Oracle Applications instances from a single console.

Using Oracle Workflow Manager, administrators can control Workflow system services, such as notification mailers, agent listeners, and other service components, background engines, purging obsolete Workflow data, and cleanup of the Workflow control queue. Administrators can also monitor work item processing by viewing the distribution of all work items by status and drilling down to additional information. Additionally, they can monitor event message processing for local Business Event System agents by viewing the distribution of event messages by status as well as queue propagation schedules. With this ability to monitor work items and event messages, a system administrator can identify possible bottlenecks easily.

To access Oracle Workflow Manager, log into Oracle Applications Manager and select an applications system. Then, you can follow one of the following navigation paths:

- Choose Workflow Manager from the pull-down menu in the Applications Dashboard page and click the Go button.

- Choose Site Map, choose the Administration tab, and then choose the Home link in the Workflow region of the Site Map page. You can also choose one of the other links in the Workflow region to navigate directly to the corresponding page within Oracle Workflow Manager.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go

You can also use other Oracle Applications Manager features to help manage Oracle Workflow.

- Use Oracle Diagnostics to run diagnostic tests that check the setup of your Oracle Workflow installation and review debugging information.

- Use Oracle Applications Logging to review Oracle Workflow logs. Oracle Workflow uses the Oracle Applications Logging framework to standardize and centralize in the database logging activities related to the Oracle Workflow Business Event System and Oracle XML Gateway.

> **Note:** The Java middle tier components of Oracle Workflow, including notification mailers and agent listeners, also use Oracle Applications Logging; however, due to the high volume of messages that pass through these components, their information is logged to the file system by default.

Gathering Oracle Workflow Statistics

Some Oracle Workflow Manager graphs and lists may summarize large volumes of data, depending on the level of activity in your Oracle Applications instance. To enhance performance in displaying these statistics, Oracle Workflow Manager periodically runs concurrent programs to gather the statistics and displays the graphs and lists based on the latest data from the concurrent programs.

- Workflow Agent Activity Statistics Concurrent Program (FNDWFAASTATCC) - Gathers statistics for the Agent Activity graph in the Workflow System status page and for the agent activity list in the Agent Activity page.

- Workflow Mailer Statistics Concurrent Program (FNDWFMLRSTATCC) - Gathers statistics for the throughput graph in the Notification Mailer Throughput page.

- Workflow Work Items Statistics Concurrent Program (FNDWFWITSTATCC) - Gathers statistics for the Work Items graph in the Workflow System status page, for the Completed Work Items list in the Workflow Purge page, and for the work item lists in the Active Work Items, Deferred Work Items, Suspended Work Items, and Errored Work Items pages.

These concurrent programs are scheduled to run every 24 hours by default. They do not require any parameters. You can optionally cancel the default scheduled requests and run the programs with a different schedule if you want to gather statistics at a different frequency.

Each of these graphs and lists displays the date and time when its statistics were last updated, as well as a refresh icon that you can select to refresh the statistics immediately if necessary. However, note that if your Oracle Applications instance contains very large volumes of workflow data, you may encounter delays or page timeouts when refreshing the data.

> **Note:** Oracle Workflow Manager statistics that typically represent smaller volumes of data, such as work item details and work item activity details, are queried directly rather than through the concurrent programs.

Oracle Workflow System Status

The Workflow System status page provides a high-level view of the status of your Oracle Workflow instance. The page displays the date and time when the system status information was last updated. To refresh this information, click the refresh icon. To add the information from this page to your support cart, click the Add to Support Cart button.

> **Note:** The system status information is queried directly, separately from the concurrent programs that gather other Oracle Workflow statistics.

The Workflow System status page shows the up, down, or unavailable summary status of the following Workflow features:

- Notification Mailers - To manage notification mailer service components, click the Notification Mailers status icon.

- Agent Listeners- To manage agent listener service components, click the Agent Listeners status icon.

- Service Components - To manage all types of service components, click the Service Components status icon.

- Background Engines - To view Workflow Background Process concurrent requests, click the Background Engines status icon.

- Purge - To view summary information about Purge Obsolete Workflow Runtime Data concurrent requests and completed work items, click the Purge status icon.

- Control Queue Cleanup - To view Workflow Control Queue Cleanup concurrent requests, click the Control Queue Cleanup status icon.

For service component features, including notification mailer service components, agent listener service components, and all types of service components grouped together, the summary status icons represent the following statuses:

- Down - At least one service component of this type has a status of Stopped with Error or System Deactivated. You should investigate the error.

- Up - At least one service component of this type has a status of Running or Suspended, and no service components of this type have a status of Stopped with Error or System Deactivated.

- Unavailable - No service components of this type have a status of Running, Suspended, Stopped with Error, or System Deactivated. For example, if all service components of this type either have not yet been completely configured, or have stopped without errors, then the Unavailable summary status is displayed.

To submit a concurrent request through Oracle Self-Service Web Applications for a feature that runs as a concurrent program, choose the program you want from the Submit Request For pull-down menu and click the Go button. You can submit requests for the following programs:

- Background Engines

- Purge

- Control Queue Cleanup

Related Database Parameters

This region displays information about database initialization parameters required for Oracle Workflow. For each parameter, the list shows the parameter name, actual parameter value, recommended value, and description. If the actual value does not match the recommended value, the recommended value is marked with a warning

indicator icon.

The JOB_QUEUE_PROCESSES parameter defines the number of SNP job queue processes for your instance. Oracle Workflow requires job queue processes to handle propagation of Business Event System event messages by AQ queues and for notification mailers. The recommended number of processes for Oracle Workflow is ten or more.

> **Note:** In Oracle Database 10*g*, you do not need to set the AQ_TM_PROCESSES parameter.

Workflow Metrics

This region displays summary information about work items and Business Event System agent activity.

Work Items

This graph displays the distribution of all work items with the following statuses: Active, Deferred, Suspended, and Error.

- To show this graph if it is hidden, click the Show link.

- To hide this graph if it is shown, click the Hide link.

- The graph header displays the date and time when the work item statistics were last updated. To refresh this information, click the refresh icon. See: Gathering Oracle Workflow Statistics, page 7-2.

- To view the distribution of item types within a status, either click the bar for that status in the graph, or click the status name link.

- To view the number of work items with a particular status, position the mouse pointer over the bar for that status in the graph.

  > **Note:** A work item can be counted in more than one status. For example, all work items that do not have an end date are counted as Active work items, including deferred, suspended, and errored work items as well as running work items. Also, if an activity within an item is deferred, and the work item as a whole is suspended, the work item is included in the count for both the Deferred and Suspended statuses. Consequently, the total of the counts for all the statuses is greater than the actual number of work items.

Agent Activity

This graph displays the distribution of all event messages on Business Event System

agents with the following statuses: Ready, Waiting, Expired, Undeliverable, and Error.

> **Note:** Messages are not explicitly assigned a status of Error. The Error bar in the graph represents messages of any status on the WF_ERROR agent.

- To show this graph if it is hidden, click the Show link.

- To hide this graph if it is shown, click the Hide link.

- The graph header displays the date and time when the agent activity statistics were last updated. To refresh this information, click the refresh icon. See: Gathering Oracle Workflow Statistics, page 7-2.

- To view the distribution of event messages with different statuses on different agents, either click the bar for a status in the graph, or click an event message status name link.

- To view the number of event messages with a particular status, position the mouse pointer over the bar for that status in the graph.

Related Links

This region provides links to other Oracle Workflow management features.

Configuration

Click the Service Components link to configure service components, including notification mailers and agent listeners.

Click the Queue Propagation link to view database initialization parameters required for queue propagation and a list of propagation schedules for Business Event System agents.

Throughput

- Click the Work Items link to view the distribution of completed work items across different item types.

- Click the Notification Mailers link to view the notification mailer throughput. This graph shows the throughput of the notification mailers by displaying the distribution of notifications in the WF_NOTIFICATIONS table with the following statuses:

  - Processed - Outbound notifications for which an e-mail message has been sent by a notification mailer service component.

  - Waiting - Outbound notifications for which an e-mail message has not yet been

sent.

The graph header displays the date and time when the notification mailer throughput statistics were last updated. To refresh this information, click the refresh icon. See: Gathering Oracle Workflow Statistics, page 7-2.

To view the number of notifications with a particular status, position the mouse pointer over the bar for that status in the graph.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Related Links > Throughput > Notification Mailers

- Click the Agent Activity link to view the distribution of event messages with different statuses on different agents.

Service Components

The Generic Service Component Framework helps to simplify and automate the management of background Java services. Service component containers and their service components are run through Generic Service Management (GSM), which you can control through Oracle Applications Manager (OAM).

A service component container is an instance of a service that manages the running of the individual service components that belong to it. The container monitors the status of its components and handles control events for itself and for its components. These actions are recorded in a log for the container.

A service component is an instance of a Java program which has been defined according to the Generic Service Component Framework standards so that it can be managed through this framework. Currently, Oracle Workflow provides four service component types: Workflow Mailer, Workflow Agent Listener, Workflow Java Agent Listener, and Workflow Web Services Outbound.

Oracle Workflow provides several seeded service components of these types, within seeded containers, to perform standard processing. You can optionally create additional service components to perform custom processing. If you create custom service components, you can either assign them to the seeded containers, or, based on the volume to be handled by the seeded containers, you can also choose to create your own custom containers.

All service components have certain attributes required by the Generic Service Component Framework. General definition attributes for a component include the component name, startup mode, container type, inbound agent, outbound agent, and correlation ID. Detail attributes include the container that owns the component, the maximum idle time for an on-demand component, maximum error count, number of inbound and outbound processing threads, component log level, read timeout period, minimum sleep time, maximum sleep time, error sleep time, and whether to close connections when the read timeout period expires.

A service component can have one of three startup modes.

- Automatic - When a component container is started, it will automatically start its automatic service components. It will also monitor these components and restart them automatically when necessary.

- On-Demand - A component container will start its on-demand service components if those components have messages waiting to be processed. For example, an on-demand notification mailer service component will be started if there are messages waiting on the WF_NOTIFICATION_OUT queue. The container will stop an on-demand service component when that component's maximum idle time has been exceeded.

- Manual - You must manually start and stop the service component through Workflow Manager. The component container does not start or stop its manual service components.

All service components use the Oracle Applications GSM container type. A component can have either an inbound agent to process inbound messages, an outbound agent to process outbound messages, or both. An Oracle Advanced Queuing (AQ) correlation ID can be assigned to a component to limit its processing to only messages marked with that correlation ID.

Oracle Workflow provides three predefined containers in which you can create components, the Workflow Mailer Service, the Workflow Agent Listener Service, and the Workflow Document Web Services Service. For an on-demand service component, you can specify the maximum amount of time that the service component can remain idle before it is stopped by its container. A service component can have either one inbound processing thread, to enable inbound processing, or none, to disable inbound processing. A service component can have one or more outbound processing threads, to enable outbound processing depending on the volume of outbound messages, or none, to disable outbound processing. Some types of service components perform only inbound processing or only outbound processing. For example, agent listeners only process inbound event messages and consequently should always have an outbound thread count of zero.

A diagnostic log is recorded for each component container, from the time the container starts to the time it stops. When a container is restarted, a new log is begun. You can view the log through Workflow Manager. Each log entry is marked with the container ID, and, if applicable, with the ID of the service component that generated it. You can specify the level of detail of the information you want to record for each component container. You can also specify a separate log level for an individual service component within the container. The log levels you can select, in order from most detailed to least detailed, are as follows:

- 1 - Statement

- 2 - Procedure

- 3 - Event

- 4 - Exception

- 5 - Error

- 6 - Unexpected

The default log level for both containers and service components is Error. This level is the recommended setting for normal usage.

A processing thread for a service component runs in a loop in which it reads messages from the queue associated with its assigned agent and then waits during a specified amount of sleep time before checking the queue for messages again. The read timeout period defines the amount of time the service component continues attempting to read messages from the queue, after the last message has been dequeued, before timing out. If another message is received before this time expires, that message is processed and the timeout period begins again. If the timeout period expires and no more messages have been received, the service component stops reading and its sleep time begins.

The minimum sleep time for a service component defines the minimum amount of time during which the service component waits, after its read timeout period expires, before it checks the queue for messages again. If a queue receives messages infrequently, you can choose to increase the sleep time between read attempts when no messages are received by setting a maximum sleep time greater than the minimum sleep time. In this case, the service component initially waits for the minimum sleep time after it finishes reading messages from its queue. If no messages are read in subsequent attempts, then the sleep time between read attempts gradually increases until the maximum sleep time is reached. Increasing the sleep time can help enhance performance if messages are received infrequently. You can also set the maximum sleep time parameter to 0 (zero) to indicate that the sleep time should not be increased. In this case, the service component always waits for the minimum sleep time between read attempts.

The error sleep time for a service component defines the amount of time during which the service component waits, after an error occurs, before it attempts to begin processing again. Additionally, a service component processing thread can either close its connections after its read timeout period expires, when its sleep time begins, or the connections can remain open until the processing thread stops.

A service component may also have additional configuration parameters that are specific to the type of processing it performs. For example, a notification mailer service component has configuration parameters to specify the inbound and outbound e-mail servers it uses.

Among both the common and the type-specific configuration parameters, some parameters can be refreshed dynamically while a service component is running. These parameters are identified by a refresh icon in the configuration pages for the component. For example, the component log level, inbound thread count, and outbound thread count are refreshable parameters.

The control events you can perform for a service component include:

- Starting a service component

- Suspending a running service component, so that the threads stop processing but connections are not closed

- Resuming a suspended service component

- Refreshing a running service component with changed parameters

- Stopping a running or suspended service component

A service component may also have additional control commands that are specific to the type of processing it performs. For example, Workflow Mailer components include a command to launch summary notifications.

You can perform these control events manually at runtime by choosing the relevant command for the component in the Service Components page. You can also schedule single or repeating control events when you are configuring a service component.

A service component can have one of the following statuses.

- Not Configured - Some required configuration parameters for the component have not been completed. The component cannot be started until its configuration is complete.

- Starting - The component is preparing to run.

- Running - The component is running normally. You can choose to suspend processing for a component in this state, refresh the configuration parameters for the component that are dynamically refreshable, or stop the component.

- Suspending - The component is preparing to suspend its processing.

- Suspended - The component's thread has stopped processing, but its connections remain open. When a component is suspended, you can either resume its processing or stop it altogether.

- Resuming - The component is preparing to resume processing and return to a Running status.

- Stopping - The component is preparing to stop running.

- Stopped - The component was stopped normally, without errors.

- Stopped with Error - The component reached the maximum number of errors specified in its Max Error Count parameter and has stopped. The component container will restart an automatic component in this status, or an on-demand

component in this status that has messages waiting to be processed.

- System Deactivated - An automatic or on-demand component was deactivated automatically by its container because the component was stopped with an error the maximum number of times specified in the container's SVC_COMP_MAX_ERROR_COUNT service parameter. A component in this status will not be restarted automatically until the container is restarted.

- User Deactivated - An automatic or on-demand component was manually stopped by a user. It will not be restarted automatically. If you want to restart it, you must do so manually.

A component with a status of Starting, Running, Suspending, Suspended, Resuming, or Stopping is considered to be active. While a component is active, you cannot edit the component name, startup mode, container type, inbound agent, outbound agent, correlation ID, container, or, for an on-demand component, the maximum idle time. You must stop the component before you can change these attributes. However, you can edit the component's other configuration parameters while it is active. If you edit any refreshable parameters, the component will be dynamically refreshed with the new parameter values.

You can manually stop a component from any status. Also, if a container stops for any reason, all of its components are stopped as well.

If the status of a service component changes to Stopped with Error or System Deactivated, Oracle Workflow posts a system alert to the System Alerts and Metrics page in Oracle Applications Manager.

Viewing Service Components

The Service Components page shows the service components that are defined in your Oracle Workflow installation.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon

To add the information from this page to your support cart, click the Add to Support Cart button.

For each service component, the list displays the service component name, status, type, startup mode, container type, and container. Click any column heading to sort the list by that column.

- To filter the service components displayed in the list, select a service component property from the Filter pull-down menu, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  - Service component name

  - Service component status

- • Service component type display name

- • Service component type internal name

- To verify that the statuses displayed for the service components in the list are current, click the Verify All button.

- To create a new service component, click the Create button.

- To edit a service component's configuration, select the service component and click the Edit button. The steps to edit the configuration depend on the service component type.

- To view the diagnostic log of the service component container in which this service component is running, select the service component and click the View Log button. The log includes log messages for this component and any other component belonging to that container.

- To view details about a service component, either click the service component link in the Name column, or select the service component and click the View Details button. The information that is displayed depends on the service component type.

- To review the events that have been scheduled to control the running of the agent listener, click the View Event History button. For each event, the Event History page displays the event name, status, user who requested the event, component status before the event was processed, date the event processing was completed, container for the service component, container type, and any event parameters for a refresh event. You can use this event history as an audit trail to review who scheduled control events for the agent listener. The status of an event may be Pending, Skipped, In Progress, Completed, or Error. In some cases, an event may be skipped if the component is not in an appropriate status at the time for which the event is scheduled. For example, a refresh event cannot be executed if the component is stopped at the scheduled time.

- To delete a service component, select the service component and click the Delete button. If the service component is currently active, you must stop it before you can delete it.

  > **Note:** Several of the seeded service components are required by Oracle Workflow and Oracle XML Gateway and cannot be deleted. If you want to disable them, you can stop them manually using the Stop command from the command pull-down menu. However, note that stopping these components disables the features they support. For example, stopping the Workflow Error Agent Listener and Workflow Java Error Agent Listener disables error handling

for the Business Event System.

- To manually control the running of a service component, select the service component, choose the command you want from the command pull-down menu, and click the Go button. You can choose the following commands:

  - Refresh

  - Resume

  - Start

  - Stop

  - Suspend

  - Launch Summary Notifications (Workflow Mailer service components only)

- To manage the service instances for the container of a service component through GSM, click the container link in the Container column.

Creating Service Components

The Pick Component Type page lets you choose the type of service component you want to create. This page lists the name and description of each available type. Select the type that you want and click the Continue button. The steps to complete the service component configuration depend on the type you select.

Oracle Workflow provides the following service component types.

- Workflow Mailer - Service components that perform send and respond e-mail processing for the Notification System.

- Workflow Agent Listener - Service components that process inbound messages on Business Event System agents in the database.

- Workflow Java Agent Listener - Service components that process inbound messages on Business Event System agents in the middle tier.

- Workflow Web Services Outbound - Service components that process outbound Web service messages.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > Create

Reviewing Service Component Details

The Component Details page lets you review the configuration of a service component.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > (B) View Details

The Component Details page displays the configuration parameters defined for the service component and any special status information, as well as the control events that are currently scheduled for the service component. For each event, the list shows the event name, initial start time, whether the event is currently running, the next scheduled execution time for a repeating event, the last previous execution time for a repeating event, the interval in minutes between executions of a repeating event, the number of times the event has failed, the job ID of the DBMS job used to schedule the event, and the PL/SQL API that DBMS job runs.

- To add the information from this page to your support cart, click the Add to Support Cart button.

- For Workflow Mailer service components only, to send a test message, click the Test Mailer button. In the Test Notification Mailer page, select the recipient role to which the message should be sent, and click the Send Test Message button. Then check the e-mail account for the recipient role to verify that the test message was received. The test message does not require a response, so you can close it after reviewing it. However, you can optionally respond with a comment to acknowledge the message.

  To send a test message successfully, you must select a recipient role that either has a valid e-mail address defined, or that has members with valid e-mail addresses defined. The recipient role must also have a notification preference that includes individual e-mail notifications.

  If you set an override e-mail address for the notification mailer, the Test Notification Mailer page displays that address. In this case the test message is sent to the override address rather than the e-mail address of the recipient role. However, you must still select a recipient role to enable the notification mailer to send the test message.

- For Workflow Mailer service components only, to set an override address where you want to send all outgoing e-mail notifications, click the Set Override Address button. Use an override address when you are testing workflow definitions or mailer processing so that you can automatically receive all the test notifications at one e-mail address, instead of having to check or change each individual recipient's e-mail address. To ensure that the override address is accessible and that its use is authorized, you must verify the request before the notification mailer can use the address.

  In the Set Override Address page, review the current override address, if any. Enter the e-mail address you want to set as the new override address, and choose Submit.

Then check the e-mail account you specified for the verification e-mail message.

In the Verify Override Address page, enter the verification code shown in the e-mail message, and choose Apply. If necessary, you can use the link provided in the verification e-mail message to navigate back to the Verify Override Address page. You must log in to Oracle Applications Manager before you can access this page.

To remove the override address, navigate to the Set Override Address page and choose the Clear Override Address button. The notification mailer then resumes sending e-mail notifications to the individual recipients' e-mail addresses.

- To review the events that have been scheduled to control the running of the service component, click the View Event History button. For each event, the Event History page displays the event name, status, user who requested the event, component status before the event was processed, date the event processing was completed, container for the service component, container type, and any event parameters for a refresh event. You can use this event history as an audit trail to review who scheduled control events for the service component. The status of an event may be Pending, Skipped, In Progress, Completed, or Error. In some cases, an event may be skipped if the component is not in an appropriate status at the time for which the event is scheduled. For example, a refresh event cannot be executed if the component is stopped at the scheduled time.

- To view the diagnostic log of the Generic Service Management (GSM) service component container in which this component is running, click the View Log button. The log includes log messages for this component and any other component belonging to that container.

- To change the values of the configuration parameters or the scheduled events, click the Edit button and navigate to the appropriate page within the service component configuration wizard.

- To return to the Service Components page, click the OK button.

Service Instances for Service Component Containers

You can use Oracle Applications Manager to control service component containers as service instances of type Generic Service Component Container in GSM.

Editing Service Parameters for a Container

Among other properties, a GSM service instance can have work shifts assigned to it. A work shift in turn can have service parameters associated with it. For a service instance that is a service component container, these service parameters apply to the container as a whole to determine how the container manages the components that belong to it.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service

Components status icon > container link > (B) Edit > (B) Edit Service Parameters

The Edit Service Parameters page initially displays the service parameters that can be specified for a container in the Edit Service Parameters field, together with their seeded default values. In most cases, you do not need to change these values. However, you can optionally edit these values in the Edit Service Parameters field if you choose.

You can also optionally delete any of the service parameters from the Edit Service Parameters field. In this case, for all parameters except the proxy setting parameters, the parameter values are obtained from the global settings stored in the WF_RESOURCES table. The default values in the WF_RESOURCES table are the same as the initial default values in the Edit Service Parameters page.

In the Edit Service Parameters field, the service parameter names and values should be specified separated by colons, in the following format:

`<name1>=<value1>:<name2>=<value2>: . . . <nameN>=<valueN>`

The following service parameters can be specified for a container:

- SVC_WRITE_DIAG_TO_GSM_LOG - Specify `Y` if you want to write diagnostic information to the GSM log file in all cases. The default value is `Y`. Specify `N` if you want to let the FND: Debug Log Filename (AFLOG_FILENAME) profile option determine where to write the log, either to a specified file or to the database if no file is specified. For more information about FND: Debug Log profile options, please refer to the *Oracle Applications System Administrator's Guide*.

- SVC_CONTAINER_LOOP_SLEEP - Specify the sleep time in seconds during which the container waits, after it finishes reading control messages from its GSM queue, before it checks that queue for messages again. The default sleep time is 10 seconds.

- SVC_CONTAINER_READ_TIMEOUT - Specify the maximum amount of time in seconds that the container continues to block on the GSM queue after processing the last message. If another message is received before this time expires, that message is processed and the timeout period begins again. If the timeout period expires and no more messages have been received, the container stops blocking on the queue and its sleep time begins. The default timeout period is 10 seconds.

- SVC_CONTAINER_LOG_LEVEL - Specify the level of detail to record for the container in its log. The default value is `5` (Error). The valid levels, in order from most detailed to least detailed, are:

  - 1 - Statement

  - 2 - Procedure

  - 3 - Event

  - 4 - Exception

- 5 - Error

- 6 - Unexpected

- SVC_COMP_MONITOR_LOOP_SLEEP - Specify the sleep time in seconds during which the container waits, after it starts any automatic components that need to be started, before it checks its automatic components again. The default value is 60 seconds.

- SVC_COMP_MONITOR_ONDEMAND_FREQ - Specify the interval in seconds to determine how often the container checks whether its on-demand components need to be started or stopped. This activity is more costly than monitoring the automatic components and should usually be performed less frequently. The default value is 300 seconds.

- SVC_COMP_MAX_ERROR_COUNT - The container-level maximum error count. If any automatic or on-demand component in the container is stopped with an error the specified number of times, the component status will be set to System Deactivated, and the container will no longer automatically restart the component. The default value is 5.

You can also optionally specify the following service parameters for proxy settings. You should set these parameters if components in this container need to use a proxy server to access web content that is outside a firewall. For example, a mailer component may need to access outside web content that is to be included in an e-mail notification. The Generic Service Component Framework uses the values you set in these service parameters to set the relevant Java System Properties.

- SVC_PROXY_SET - Specify `true` to indicate that you want to use a proxy for your connections. The default value is `NONE`.

- SVC_PROXY_HOST - Specify the host machine for the proxy. The default value is `NONE`.

- SVC_PROXY_PORT - Specify the port on which the proxy is listening. The default value is `NONE`.

Selecting the Log Level for a Container

You can use the Service Status page to control the running of a service component container, including changing the log level for the container. The log level controls how much information is recorded in the log. Note that the log level you select here applies only to the log messages for the container. You can assign separate log levels to the individual components within the container.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > container link > (B) View Status

The log level with which the container starts is determined by the value of the SVC_CONTAINER_LOG_LEVEL service parameter. If no value is defined for that parameter, the log level is obtained from the default setting stored in the WF_RESOURCES table. The default container log level, which is also the recommended setting, is Error.

If the container is running, you can optionally specify a different container log level for the current session. To change the log level, select the level you want from the Change Log Level To pull-down menu and click the Go button. The log levels you can select, in order from most detailed to least detailed, are as follows:

- 1 - Statement

- 2 - Procedure

- 3 - Event

- 4 - Exception

- 5 - Error

- 6 - Unexpected

Note that the log level you set dynamically in the Service Status page applies only for the duration of the current container session, and does not change the log level stored for the container in the service parameters. To set the log level permanently, so that the container starts with that log level in each new session, edit the value of the SVC_CONTAINER_LOG_LEVEL service parameter in the Edit Service Parameters page. See: Editing Service Parameters for a Container, page 7-15.

If the log level has been changed dynamically for the current session, the Service Status page may not display the log level that is currently in effect for the container. However, you can always review the current log level in the container log file by choosing View Log in the Service Components page or the Component Details page.

Creating Service Component Containers

If you create custom service components, you can choose to create custom containers to manage those service components. You create a container as a GSM service instance of type Generic Service Component Container in Oracle Applications Manager.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > container link > (B) Create New

Among other properties, a GSM service instance can have work shifts assigned to it. A work shift in turn can have service parameters associated with it. For a service instance that is a service component container, these service parameters apply to the container as a whole to determine how the container manages the components that belong to it. If you create a custom container, you should specify service parameters for the work shifts for your new service instance in order to specify how to run the new container. To

enter service parameters easily, copy the service parameters from one of the seeded Oracle Workflow containers to your new container.

After creating a customer container, you can assign service components to it using the appropriate service component configuration wizard. Ensure that your custom containers are running in order to run the service components belonging to them.

Notification Mailers

A notification mailer is a Java program that performs e-mail send and response processing for the Oracle Workflow Notification System, using the JavaMail API. You need to implement one or more notification mailers only if you want to have your workflow users receive their notifications by e-mail, as well as from the Worklist Web pages.

Managing Notification Mailers

The notification mailer program is defined as a service component type in the Generic Service Component Framework. This framework helps to simplify and automate the management of background Java services.

Oracle Workflow provides one seeded notification mailer service component, called Workflow Notification Mailer. Most of the configuration parameters for this mailer are set to default values. You can enter several of the remaining required parameters using AutoConfig. After installation, you then only need to enter the e-mail inbox password in order to complete the configuration of this mailer. Alternatively, if you only want to send outbound messages and do not need to receive inbound messages, you only need to disable inbound processing in order to complete the configuration of this mailer. If the mail servers and Business Event System components used by the notification mailers are set up, and the Workflow Mailer Service container to which the Workflow Notification Mailer belongs is started, the seeded notification mailer automatically starts running once its configuration is complete.

You cannot delete the seeded Workflow Notification Mailer or edit its name, assigned agents, correlation ID value, or container. However, if necessary you can optionally update other configuration parameters, schedule control events, or manually choose control commands to start, stop, suspend, resume, or refresh this notification mailer.

> **Note:** Oracle Alert also uses the Workflow Notification Mailer to send and receive alert e-mail messages. If you use Oracle Alert, ensure that the configuration of the Workflow Notification Mailer meets your alert requirements. See: Setup Steps, *Oracle Alert User's Guide*.

You can also optionally create additional notification mailer service components. For example, you can create a notification mailer that processes only messages that belong to a particular workflow item type, or create additional mailers that process the same types of message to increase throughput.

The correlation ID for a notification mailer determines which messages it can process. To dedicate a notification mailer to processing messages from a particular item type, set that item type as the correlation ID. To create a general notification mailer that can process messages from any item type, leave the correlation ID blank. The seeded Workflow Notification Mailer has a blank correlation ID so that it can run as a general mailer.

> **Note:** If you run a general notification mailer and a dedicated notification mailer for a particular item type at the same time, a message from that item type may still be processed by the general mailer if that mailer is the first to access the message. If you want only the dedicated notification mailer to process messages from that item type, disable any general mailers. In this case, however, ensure that you define dedicated mailers for all item types used in your Oracle Applications installation.

To ensure consistency in message handling, all notification mailers that can process the same messages must share the same values for certain parameters. Multiple mailers can process the same messages in the following cases:

- A general mailer runs at the same time as any dedicated mailers.

- Multiple general mailers run at the same time.

- Multiple dedicated mailers for the same item type run at the same time.

In these cases, the notification mailers must share the same values for the following parameters:

- HTML Agent

- Attach Images to Outbound E-mails

- Attach Stylesheet to Outbound E-mail

- Autoclose FYI

- Direct Response

- Reset NLS

- Inline Attachments

- All message template parameters

However, these mailers can have different values for the From and Reply-to Address parameters. The headers of each notification e-mail message will contain the From and Reply-to Address values of the notification mailer that actually sent the message, unless

the message itself has the special `#WFM_FROM` and `#WFM_REPLYTO` message attributes defined to override the notification mailer's parameters. See: Notification Mailer Attributes, *Oracle Workflow Developer's Guide*.

You can also configure any notification mailer service component to process only inbound messages, or only outbound messages. You associate inbound and outbound mailers with each other by assigning them the same mailer node name. The mailer node name indicates which inbound mailer can process incoming responses to outbound messages sent by a particular outbound mailer.

You can optionally assign the same node name to multiple mailers for load balancing purposes. However, each mailer that performs inbound processing for a node must have its own inbox.

- If you enable both outbound and inbound processing for the same mailer, that mailer will automatically use the same node name for both types of processing, enabling it to process incoming responses to the outbound messages it sent. You can optionally also create other notification mailers that share the same node name.

- If you create an outbound-only mailer, but you still want to perform response processing for e-mail responses to the outbound messages it sends, you should create at least one other mailer with the same node name that does perform inbound message processing. Otherwise, there will be no inbound mailer that can process incoming responses to outbound messages sent by this outbound mailer.

- If you only want to implement outbound message processing, without inbound e-mail response processing, then you can configure an outbound-only mailer without creating a corresponding inbound mailer. In this case you should configure the mailer to use message templates for response-required notifications that do not request a response by e-mail, but instead direct recipients to respond from the Notification Details Web page. For example, you can configure the mailer to send response-required notifications using the Workflow View From UI message template, which is an alternative template provided by Oracle Workflow in the System: Mailer item type, or create your own custom message templates. The outbound-only mailer can still use the standard message templates to send outbound summary notifications or For Your Information (FYI) notifications that do not require a response.

- Create an inbound-only mailer only if you have also created at least one mailer with the same node name that performs outbound message processing. If no outbound mailer shares the same node name, no incoming response messages will be marked with that node name, and the inbound-only mailer will have no messages to process.

Dedicated mailers for different item types should use different node names.

If you create custom notification mailer service components, you can either assign them to the seeded container for notification mailers, named Workflow Mailer Service, or, based on the volume to be handled by the seeded container, you can also choose to

create your own custom containers.

Setting Up a Notification Mailer

Currently, Oracle Workflow supports the Simple Mail Transfer Protocol (SMTP) for outbound messages and the Internet Message Access Protocol (IMAP) for inbound messages. You must have an SMTP server set up in order to send Oracle Workflow notification e-mail messages, and an IMAP server set up if you want to receive e-mail notification responses. Users can receive e-mail notifications using various e-mail clients, although notifications may be displayed differently in different clients, depending on the features each client supports.

> **Note:** Oracle Workflow supports IMAP version 4 (IMAP4) compliant mail servers. Ensure that your mail server uses this IMAP version. For more information, see the JavaMail API Design Specification: http://java.sun.com/products/javamail/JavaMail-1.2.pdf

> **Note:** If you have certain types of software installed, you may already have the necessary mail server functionality available. For example, products such as Oracle Email, Microsoft Exchange, or Lotus Notes include IMAP services. You can use a UNIX server as an SMTP server by configuring the Sendmail program.
>
> Additionally, you can choose to use IMAP server software that is available for download from some sources. For example, the University of Washington offers the UW IMAP Server as a public service, and Carnegie Mellon University offers the Cyrus IMAP Server. You might choose this option if your enterprise uses UNIX Sendmail e-mail accounts, for instance. For more information, see: http://www.washington.edu/imap/, http://asg.web.cmu.edu/cyrus/, and http://www.imap.org/.

> **Note:** Third party software products are mentioned as examples only. Oracle makes no recommendation or endorsement of these third party software products.

To set up a notification mailer, you must perform the following steps.

1. Set up an SMTP mail server to send outbound messages.

2. Set up an IMAP4 compliant mail server with an e-mail account for the notification mailer if you want to receive inbound messages.

   The notification mailer requires three folders in this e-mail account: the inbox, a folder to store processed messages, and a folder to store discarded messages. If the

e-mail account does not already include folders named PROCESS and DISCARD, Oracle Workflow automatically creates these two folders when you complete the basic notification mailer configuration. You can optionally specify other folders for the notification mailer using the advanced configuration wizard.

> **Note:** If you create the PROCESS and DISCARD folders manually before configuring the notification mailer, use your e-mail client to create these folders. A notification mailer may not be able to access folders that were created using command line tools outside the e-mail client.

3. You can use AutoConfig to enter the following configuration parameters for the seeded Workflow Notification Mailer service component during installation. For more information about running AutoConfig, see *Using AutoConfig to Manage System Configurations with Oracle Applications Release 12*, Oracle*MetaLink* note 387859.1 and AutoConfig, *Oracle Applications Concepts*.

   - SMTP Server

   - IMAP Server (if you want to receive inbound messages)

   - Inbox Username (if you want to receive inbound messages)

   - Reply To E-mail Address (if you want to receive inbound messages)

   - HTML Agent Name - This parameter defaults to the value you enter for the Applications Servlet Agent parameter in AutoConfig. Use the following format:

     `http://<server_name:port>/OA_HTML/`

     > **Note:** When you enter the SMTP Server and IMAP Server parameters, specify the actual host name for each server. Do not use `localhost` as the setting for these parameters. You can optionally specify the port number to use on each server. If you do not specify a port number, the notification mailer uses port 143 on the IMAP server and port 25 on the SMTP server by default. Specify each server in the following format: `<server_name>[:<port_number>]`

4. Ensure that the Business Event Local System status is set to Enabled in the Workflow Configuration page, and that the JOB_QUEUE_PROCESSES database initialization parameter, which is required for the Business Event System, is set to an appropriate value. The Business Event Local System status is set to Enabled by default, and usually you do not need to change this status. If notification processing is not being completed, however, you should check this preference value.

5. **(Recommended)** You can optionally set the WF: Workflow Mailer Framework Web Agent profile option to the host and port of the Web server that notification mailers should use to generate the content for Oracle Application Framework regions that are embedded in notifications. If this profile option is not set, notification mailers will use the same Web agent specified in the Application Framework Agent profile option. However, if necessary for load balancing purposes, you can optionally specify a different Web agent for notification mailers to use. The WF: Workflow Mailer Framework Web Agent profile option should be set at site level. See: Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide*.

6. Before a service component can run, the container which manages it must first be started. The seeded Workflow Notification Mailer service component belongs to a container named Workflow Mailer Service, while the seeded agent listener service components that are also required for notification mailer processing belong to a container named Workflow Agent Listener Service. You should ensure that these two containers are running. If you create your own custom containers for custom service components, ensure that those containers are running as well. Use the Service Instances page to start the containers as service instances in Generic Service Management (GSM).

7. When the Workflow Agent Listener Service container is running, it automatically starts seeded agent listener service components named Workflow Deferred Notification Agent Listener, Workflow Error Agent Listener, and Workflow Inbound Notifications Agent Listener, which are required for notification mailer processing. Ensure that these agent listeners are running.

8. Use the notification mailer configuration wizard to configure your notification mailer service component. The Basic Configuration page lets you configure a notification mailer quickly by entering only the minimum required parameters, while the advanced configuration wizard lets you specify additional parameters to control how the notification mailer processes messages.

   If you entered configuration parameters for the seeded Workflow Notification Mailer through AutoConfig, you only need to enter the password for the e-mail inbox in order to complete the configuration for that mailer and begin running it. If you did not enter parameters for the seeded mailer through AutoConfig, then in order to complete the configuration for that mailer you need to enter only the SMTP server, IMAP server, e-mail inbox username, e-mail inbox password, and reply-to e-mail address. All other configuration parameters for the seeded Workflow Notification Mailer are initially set to default values and do not need to be changed, although you can optionally do so if you choose.

   > **Note:** The IMAP server, e-mail inbox username, e-mail inbox password, and reply-to e-mail address are required only if you want to receive inbound messages. Alternatively, if you only want to send outbound messages and do not need to receive inbound

messages, you only need to disable inbound processing in order to complete the configuration of the Workflow Notification Mailer.

9. **(Optional)** By default, the seeded Workflow Notification Mailer has a Launch Summary Notifications event scheduled to send summary notifications once a day. You can optionally use the notification mailer configuration wizard to modify the start time and interval for this event's schedule, or to schedule the Launch Summary Notifications event at the interval you choose for any notification mailer service component. When this event is processed, a summary notification is sent to each role with a notification preference of SUMMARY or SUMHTML, listing all the notifications that are currently open for that role.

10. **(Optional)** You can configure a notification mailer to connect to the SMTP server and IMAP server through Secure Sockets Layer (SSL) to encrypt the data exchanged. See: Connecting to Mail Servers Through SSL, page 2-56.

11. **(Optional)** You can optionally set the internal mailer parameter named `HTML_DELIMITER` to specify which characters the notification mailer uses to delimit response values in response templates for HTML-formatted e-mail notifications. Valid values for the `HTML_DELIMITER` parameter are:

    - `DEFAULT` - The notification mailer uses the default delimiters, currently set as the single quote (`'`) for both the opening and the closing delimiter. The notification mailer also uses the default delimiters if the `HTML_DELIMITER` parameter value is left null.

    - `APOS` - The notification mailer uses the single quote, or apostrophe (`'`), as both the opening and the closing delimiter. This setting is currently the same as the default.

    - `QUOTE` - The notification mailer uses the double quote (`"`) as both the opening and the closing delimiter.

    - `BRACKET` - The notification mailer uses the left bracket (`[`) as the opening delimiter and the right bracket (`]`) as the closing delimiter.

    Using single quotes as the delimiters accommodates e-mail applications that cannot process double quotes in the <A HREF="mailto:"> tag for the response template link, but can accept single quotes. However, if you want users to be able to use apostrophes or single quotes in their response values without entering an escape character, you can use double quotes or brackets as the delimiters, depending on what your e-mail application supports. See: To Respond to an HTML E-mail Notification, *Oracle Workflow User's Guide*.

    > **Note:** If the `HTML_DELIMITER` parameter is set to an invalid value,

the notification mailer throws an exception at startup. Any notifications created during this time are rendered with the default delimiters instead.

By default, the HTML_DELIMITER parameter is set to the value DEFAULT. Use the afsvcpup.sql script to change the parameter value to specify the delimiters you want to use. See: To Set Internal Mailer Parameters, page 2-50.

If a particular notification message has the special #WFM_HTML_DELIMITER message attribute defined, however, the notification mailer will use the #WFM_HTML_DELIMITER attribute value to determine which delimiters to use for that notification, instead of using the HTML_DELIMITER parameter value.

> **Note:** The HTML_DELIMITER parameter only controls the response templates for HTML-formatted notifications. This parameter does not apply to plain text notifications.

12. **(Optional)** The seeded Workflow Notification Mailer uses the Automatic startup mode by default and will be started automatically when you complete its configuration. If you select the Manual startup mode for a notification mailer service component, use the Service Components page to start that notification mailer. You can also use this page to manage any notification mailer service component.

Outbound Notification Mailer Processing

When the Workflow Engine determines that a notification message must be sent, it raises an event in the Business Event System called oracle.apps.wf.notification.send. Oracle Workflow provides a seeded subscription to this event, which is defined to be deferred immediately so that the workflow process that owns the notification can continue. The event is placed on the standard WF_DEFERRED agent. Oracle Workflow provides a seeded agent listener named Workflow Deferred Notification Agent Listener that runs on this agent to continue notification processing. This agent listener is dedicated solely to processing deferred notification events.

When the event is dequeued from WF_DEFERRED and the subscription is processed, the subscription requires the event data for the event, causing the generate function for the event to be executed. The generate function for this event performs the following actions:

• Resolves the notification recipient role to a single e-mail address, which itself can be a mail list.

• Checks the notification preference of the recipient to determine whether an e-mail notification is required, and in what type of format.

- Switches its database session to the recipient role's preferred language and territory as defined in the directory service.

- Generates an XML representation of the notification message and any optional attachments using the appropriate message template.

Finally, the subscription places the event message on the standard WF_NOTIFICATION_OUT agent.

A notification mailer service component polls the WF_NOTIFICATION_OUT agent for messages that must be sent by e-mail. When the notification mailer dequeues a message from this agent, it uses a Java-based notification formatter to convert the XML representation of the notification into a MIME (Multipurpose Internet Mail Extensions) encoded message and sends the message by the Simple Mail Transfer Protocol (SMTP).

The e-mail notifications are based on message templates defined in Oracle Workflow Builder. Oracle Workflow provides a set of standard templates in the System: Mailer item type, which are used by default. It is not recommended to modify the standard templates. However, you can customize the message templates used to send your e-mail notifications by creating your own custom message templates in a custom item type using the Workflow Builder. Then assign these templates to a particular notification in a workflow process by defining special message attributes. In this case the templates assigned to the notification override any other templates.

You can also create your own custom message templates in the System: Mailer item type using the Workflow Builder, and assign these templates to a particular notification mailer service component in the mailer configuration parameters. The templates assigned to a mailer override the default System: Mailer templates. However, if any notifications have templates specifically assigned to them through message attributes, the notification-level templates still override the templates assigned to the mailer.

If the notification mailer cannot deliver an e-mail notification because the recipient's e-mail address is invalid, it performs the following actions:

- Sets the mail status of the notification to `FAILED`. This mail status indicates that an exception prevented this e-mail notification from being delivered but does not prevent the mailer from processing other notifications.

- Adds the e-mail address to its invalid e-mail address list. To avoid unnecessary processing, each notification mailer stores a list of e-mail addresses to which it could not deliver messages, and does not attempt to send any further messages to those addresses. Instead, for any subsequent notifications to the listed addresses, the notification mailer simply sets the mail status directly to `FAILED`.

  > **Note:** Each notification mailer can store up to 100 e-mail addresses in its invalid e-mail address list. If the notification mailer encounters additional invalid addresses when the list is already full, the notification mailer removes the oldest addresses from the

list and adds the new addresses in their place. Also, the notification mailer clears the list by removing all addresses whenever you stop and restart the mailer.

- Changes the notification preference of the recipient to DISABLED. To further help avoid unnecessary processing, if a recipient has a notification preference of DISABLED, Oracle Workflow does not generate a complete XML representation of any notifications to that recipient, and a notification mailer does not attempt to send e-mail notifications to that recipient. Instead, the notification mailer simply sets the mail status of the notifications directly to FAILED. The change in notification preference also indicates to the user that e-mail notifications cannot be delivered. The user must correct the invalid e-mail address and then reset the notification preference in order to receive e-mail notifications.

- Sends a notification to the SYSADMIN user with the information that an e-mail notification could not be sent to one or more recipients, that the notification preference for those recipients has been set to DISABLED, and that those recipients' original notification preferences, which are listed, should be reset after the issues that caused the failures are corrected. See: User Notification Preference Update Report Message, page 2-134.

After correcting invalid e-mail addresses and resetting DISABLED notification preferences, you can run the Resend Failed Workflow Notifications concurrent program to retry open notifications that previously could not be sent. See: Handling Mailer Errors, page 2-73.

Inbound Notification Mailer Processing

Notification mailers can also process e-mail responses from users, using the Internet Message Access Protocol (IMAP). A notification mailer uses a Java-based e-mail parser to interpret the text of each message and create an XML representation of it.

A notification mailer uses three folders in your response mail account for response processing: one to receive incoming messages, one to store processed messages, and one to store discarded messages.

A notification mailer does the following to process response messages:

- Logs into its IMAP e-mail account.

- Checks the inbox folder for messages. If a message exists, the notification mailer reads the message, checking for the notification ID (NID) and node identifier in the NID line.

- If the message is not a notification response, meaning it does not contain an NID line, the notification mailer moves the message to the discard folder and treats it as

an unsolicited message. For the first unsolicited message from a particular e-mail address, the notification mailer also sends a warning message back to the sender of the message. However, to avoid sending unnecessary warnings due to bounced or auto-reply messages, each mailer node stores a list of e-mail addresses from which it has received unsolicited mail, and does not send any further warning messages to those addresses. Instead, if the node receives a second unsolicited message from a particular address, the notification mailer discards the message and raises the oracle.apps.wf.mailer.unsolicited event. You can optionally define a subscription to this event if you want to perform some other action in response to the second unsolicited message. For all subsequent unsolicited messages, the notification mailer simply discards the message.

> **Note:** Each mailer node can store up to 100 e-mail addresses in its warned list. If the node receives unsolicited messages from additional addresses when the list is already full, the notification mailer removes the oldest addresses from the list and adds the new addresses in their place. Also, the notification mailer clears the list by removing all addresses when you start the mailer for the first time, and again whenever you stop and restart its container. In these cases, the mailer may send another warning message if it receives further unsolicited e-mail from an address that is no longer on the warned list.

> **Note:** You can optionally use the Send Warning for Unsolicited E-mail mailer parameter to prevent notification mailers from sending any warning messages at all. See: Notification Mailer Configuration Wizard, page 7-30.

- If the message is a notification response, but for a different node, the notification mailer leaves the message in the inbox and adds the e-mail's Unique Message ID (UID) to its ignore list.

- If the message is a notification response for the current node, meaning it contains an NID line including the node identifier of the current node, the notification mailer processes the message.

The notification mailer performs the following steps for messages that belong to its node.

- Retrieves the notification ID.

- Checks to see if the message bounced by referring to the tags specified in the configuration parameters, if any. If the message bounced, the notification mailer updates the notification's status and stops any further processing, based on the specifications of the tag list.

- Checks the Oracle Workflow database for this notification based on the NID line.

  - If the notification does not exist, meaning the notification ID or the access key in the NID line is invalid, the notification mailer moves the message to the discard folder. If the NID line is incorrectly formatted, the notification mailer moves the message to the discard folder and treats it as an unsolicited message.

  - If the notification exists, but is closed or canceled, the notification mailer moves the message to the processed folder and sends a Workflow Closed Mail or Workflow Canceled Mail message to the recipient role, respectively.

    > **Note:** You can optionally use the Send E-mails for Canceled Notifications mailer parameter to prevent notification mailers from sending any notification cancellation messages. See: Notification Mailer Configuration Wizard, page 7-30.

  - If the notification exists and is open, the notification mailer generates an XML representation of the message and places it on the standard WF_NOTIFICATION_IN agent as an event called oracle.apps.wf.notification.receive.message. The notification mailer then moves the message for the completed notification to the processed folder.

    > **Note:** If the character encoding of the response message is not compatible with the database codeset, the notification mailer may not be able to parse the response and recognize the response values. Ensure that the character encoding of messages in your mail client is compatible with the codeset of your database.

Finally, if there are no more unprocessed messages in the inbox, the notification mailer logs out of the e-mail account.

Oracle Workflow provides a seeded agent listener named Workflow Inbound Notifications Agent Listener that runs on the WF_NOTIFICATION_IN agent to continue notification processing for the valid response messages placed on that agent. When an event message is dequeued from WF_NOTIFICATION_IN, Oracle Workflow executes a seeded subscription that calls the appropriate notification response function. This function verifies the response values with the definition of the notification message's response attributes in the database. If a response value is invalid, or if no response value is included, the notification mailer sends a Workflow Invalid Mail message to the recipient role, or, for an invalid response to a request for more information, the notification mailer sends a Workflow Invalid Open Mail (More Information Request) message to the recipient role. If the responses are valid, the notification response function records the response and completes the notification.

Notification Mailer Configuration Wizard

Use the notification mailer configuration wizard to configure a new notification mailer service component, or to edit the configuration of an existing notification mailer service component. The notification mailer configuration wizard begins with the Basic Configuration page, which lets you configure a notification mailer quickly by entering only the minimum required parameters.

From the Basic Configuration page, you can also navigate to the advanced configuration wizard to specify additional parameters that control how the notification mailer processes messages. The advanced configuration wizard lets you define general and detail attributes, define e-mail server and message generation parameters, schedule control events, and define tags to assign statuses to unusual messages.

Some parameters appear in both the Basic Configuration page and the advanced configuration wizard. Both the Basic Configuration page and the advanced configuration wizard also let you send a test message.

> **Note:** If you are configuring the seeded Workflow Notification Mailer and you entered configuration parameters for this mailer through AutoConfig, then you only need to enter the password for the e-mail inbox in order to complete the configuration for that mailer. If you did not enter parameters for the seeded mailer through AutoConfig, then in order to complete the configuration for that mailer you need to enter only the SMTP server, IMAP server, e-mail inbox username, e-mail inbox password, and reply-to e-mail address. All other configuration parameters for the seeded Workflow Notification Mailer are initially set to default values and do not need to be changed, although you can optionally do so if you choose.
>
> Note that the IMAP server, e-mail inbox username, e-mail inbox password, and reply-to e-mail address are required only if you want to receive inbound messages. Alternatively, if you only want to send outbound messages and do not need to receive inbound messages, you only need to disable inbound processing in order to complete the configuration of the Workflow Notification Mailer.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Notification Mailers status icon > (B) Create > (B) Continue

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Notification Mailers status icon > (B) Edit

Basic Configuration

This page lets you configure a notification mailer quickly by entering only the minimum required parameters in a single page. You must set parameters marked with an asterisk (*) to appropriate values for your environment before you can run the

notification mailer.

Details

- **Name** - The name of the service component. This name must be unique. The name of the seeded notification mailer service component is `Workflow Notification Mailer`, and you cannot change this value.

Outbound E-mail Account (SMTP)

- **Server Name** - The name of the outbound SMTP mail server. Note that you must specify the actual host name for the server. Do not use `localhost` as the setting for this parameter. You can optionally specify the port number to use on that server. If you do not specify a port number, the notification mailer uses port 25 by default. Specify the server in the following format: `<server_name>[:<port_number>]`

  For example: `mysmtpserver.mycompany.com:25`

- **Outbound SSL Enabled** - Select this parameter to enable the notification mailer to use Secure Sockets Layer (SSL) for connections to the SMTP server. Deselect this parameter to use non-SSL connections.

  > **Note:** When you enable SSL, the notification mailer connects to the SMTP server through port 465 by default. You can optionally specify a different port number along with the SMTP server name in the Outbound E-mail Account (SMTP): Server Name parameter.

  Before you can use SSL, you must also complete additional setup steps. See: Connecting to Mail Servers Through SSL, page 2-56.

Inbound E-mail Account (IMAP)

- **Inbound Processing** - Select this parameter to enable inbound e-mail processing with this notification mailer. Deselect this parameter to disable inbound e-mail processing for this notification mailer and dedicate the notification mailer solely to outbound processing.

  If you disable inbound processing, you can leave the other inbound parameters blank.

- **Server Name** - The name of the inbound IMAP mail server. Note that you must specify the actual host name for the server. Do not use `localhost` as the setting for this parameter. You can optionally specify the port number to use on that server. If you do not specify a port number, the notification mailer uses port 143 by default. Specify the server in the following format: `<server_name>[:<port_number>]`

  For example: `myimapserver.mycompany.com:143`

- **Username** - The user name of the mail account that the notification mailer uses to receive e-mail messages.

- **Password** - The password for the mail account specified in the Username parameter. The password value is masked as asterisks in the display and is stored in encrypted form.

- **Reply-To Address** - The address of the e-mail account that receives incoming messages, to which notification responses should be sent. This value must be a full RFC822-compliant e-mail address.

  If a particular notification message has the special #WFM_REPLYTO message attribute defined, however, the notification mailer will use the #WFM_REPLYTO attribute value as the reply address for that message, instead of the Reply-To Address parameter value.

  > **Note:** If you enable inbound processing, Oracle Workflow by default sets the From parameter, which is displayed in the From field of the message headers, to the name portion of the reply-to address. For example, if the reply-to address is `Workflow@mycompany.com`, the notification mailer sets the From parameter to `Workflow`.

  > If you disable inbound processing, Oracle Workflow by default sets both the Reply-To Address parameter and the From parameter to `nobody@<server_name>`, where `<server_name>` is the name of the outbound SMTP mail server.

  > To specify a different From value, navigate to the advanced configuration wizard.

- **Inbound SSL Enabled** - Select this parameter to enable the notification mailer to use SSL for connections to the IMAP server. Deselect this parameter to use non-SSL connections.

  > **Note:** When you enable SSL, the notification mailer connects to the IMAP server through port 993 by default. You can optionally specify a different port number along with the IMAP server name in the Inbound E-mail Account (IMAP): Server Name parameter.

  Before you can use SSL, you must also complete additional setup steps. See: Connecting to Mail Servers Through SSL, page 2-56.

  > **Note:** The notification mailer requires three folders in the IMAP mail account: the inbox, a folder to store processed messages, and a folder to store discarded messages. If you enable inbound processing and the

mail account you specify in the Username parameter does not already include folders named PROCESS and DISCARD, Oracle Workflow automatically creates these two folders. To specify other folders for the notification mailer, navigate to the advanced configuration wizard.

> **Note:** If you enable inbound processing, the notification mailer uses the Workflow Open Mail (Templated) message, which provides a response template for sending responses by e-mail, as the default message template for e-mail notifications that require a response. If you disable inbound processing, the notification mailer uses the Workflow Open Mail (Outlook Express) message, which provides a link in HTML notifications for entering responses in the Notification Details page, as the default message template for e-mail notifications that require a response. To specify other message templates, navigate to the advanced configuration wizard.

> Note that the plain text version of the Workflow Open Mail (Outlook Express) message requests a response by e-mail. If you disable inbound processing, ensure that your users do not have a notification preference of MAILTEXT or MAILATTH. Alternatively, if you disable inbound processing and you want users to receive plain text notifications, use the advanced configuration wizard to specify a message template that directs recipients to respond from the Notification Details Web page, such as the standard Workflow View From UI message template or a custom message template.

To cancel any changes on this page, click the Cancel button.

To save this configuration, click the Apply button.

To send a test message, click the Test Mailer button. In the Test Notification Mailer page, select the recipient role to which the message should be sent, and click the Send Test Message button. Then check the e-mail account for the recipient role to verify that the test message was received. The test message does not require a response, so you can close it after reviewing it. However, you can optionally respond with a comment to acknowledge the message.

> **Note:** To send a test message successfully, you must select a recipient role that either has a valid e-mail address defined, or that has members with valid e-mail addresses defined. The recipient role must also have a notification preference that includes individual e-mail notifications.

> If you set an override e-mail address for the notification mailer, the Test Notification Mailer page displays that address. In this case the test message is sent to the override address rather than the e-mail address of the recipient role. However, you must still select a recipient role to

enable the notification mailer to send the test message. See: Reviewing Service Component Details, page 7-13.

To set additional parameters for this notification mailer in the advanced configuration wizard, click the Advanced button.

Define

This page lets you define general attributes for the service component. Some attributes are already set to required values and cannot be modified. You must set attributes marked with an asterisk (*) to appropriate values for your environment before you can run the service component.

- **ID** - The configuration wizard displays the identifier for the service component.

- **Status** - The configuration wizard displays the status of the service component.

- **Name** - The name of the service component. This name must be unique. You can only edit the name when the notification mailer is not running. The name of the seeded notification mailer service component is `Workflow Notification Mailer`, and you cannot change this value.

- **Startup Mode** - Select Automatic, Manual, or On-Demand as the startup mode for the service component. You can only edit the startup mode when the notification mailer is not running. The seeded Workflow Notification Mailer is assigned the Automatic startup mode by default, but you can optionally change this value.

- **Container Type** - The container type to which this service component belongs, which is always Oracle Applications Generic Service Management (Oracle Applications GSM).

- **Inbound Agent** - The Business Event System agent for inbound processing. The inbound agent for a notification mailer service component is always WF_NOTIFICATION_IN.

- **Outbound Agent** - The Business Event System agent for outbound processing. The outbound agent for a notification mailer service component is always WF_NOTIFICATION_OUT.

- **Correlation ID** - Optionally select an item type to specify that this notification mailer should only process messages that belong to that item type. If you enter a partial value, this notification mailer will process messages that belong to any item type whose internal name begins with that value. You can enter an item type as the correlation ID if you want to increase throughput for that particular item type by dedicating a notification mailer to it.

  If you leave the correlation ID blank, this notification mailer will process messages

from any item type. The seeded Workflow Notification Mailer does not have any correlation ID specified, so that it can operate generally to process all messages; you cannot change this setting.

Both dedicated and general notification mailer components are compatible with each other. You can run several dedicated and general notification mailers at the same time if you choose. In this case, note that even if you have configured a dedicated mailer for a particular item type, a message from that item type may still be processed by a general mailer if that mailer is the first to access the message.

To cancel any changes on this page, click the Cancel button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

Details

This page lets you define detail attributes for the service component. You must set attributes marked with an asterisk (*) to appropriate values for your environment before you can run the service component. A refresh icon identifies attributes that can be refreshed dynamically while the service component is running.

- **ID** - The configuration wizard displays the identifier for the service component.

- **Status** - The configuration wizard displays the status of the service component.

- **Name** - The configuration wizard displays the name defined for the service component.

- **Container** - The container to which the service component will belong. Oracle Workflow provides a container called Workflow Mailer Service for notification mailer service components.

- **Maximum Idle Time** - If you selected the On-Demand startup mode for the service component, enter the maximum time in minutes that the service component can remain idle before it is stopped. An on-demand component that is stopped in this way will be restarted by its container when it is needed again to process new messages.

- **Max Error Count** - The number of consecutive errors the service component can encounter before its container stops it and changes its status to Stopped with Error. If an error is resolved and processing can continue, the error count is reset. The default value for the maximum error count is 10.

- **Inbound Thread Count** - Set the inbound processing thread count to 1 (one) to enable inbound message processing with this notification mailer. Select 0 (zero) to disable inbound message processing for this notification mailer and dedicate the notification mailer solely to outbound processing. If you selected the Inbound Processing parameter in the Basic Configuration page, the inbound thread count is

set to 1; if you deselected the Inbound Processing parameter, the inbound thread count is set to 0.

The inbound thread count cannot be greater than 1, because only one thread can access the e-mail inbox at a time. If you disable inbound message processing for this notification mailer, but you still want to perform e-mail response processing, you should create at least one other notification mailer with the same node name that does perform inbound message processing. Otherwise, there will be no inbound mailer that can process incoming responses to outbound messages sent by this outbound mailer.

- **Outbound Thread Count** - Specify the number of outbound processing threads you want to execute simultaneously with this notification mailer. You can set the outbound thread count to 1 (one) or more depending on the volume of outbound messages you need to send. Specify 0 (zero) to disable outbound message processing for this notification mailer and dedicate the notification mailer solely to inbound processing. If you disable outbound message processing for this notification mailer, you should create at least one outbound notification mailer with the same node name. Otherwise, no inbound response messages will be marked with that node name and this inbound mailer will have no messages to process. The default value for the outbound thread count is 1.

- **Log Level** - Select the level of detail for the information you want to record in the service component container log. The recommended log level, which is also the default value, is Error. Usually the log level only needs to be changed if you want to record additional detailed information for debugging purposes. You can choose the following levels:
    - 1 - Statement

    - 2 - Procedure

    - 3 - Event

    - 4 - Exception

    - 5 - Error

    - 6 - Unexpected

- **Processor Read Wait Timeout** - Specify the amount of time in seconds that the service component's processing thread continues to wait, after reading the last message from its assigned queue, before timing out. If another message is received before this time expires, that message is processed and the timeout period begins again. If the timeout period expires and no more messages have been received, the service component stops reading and its sleep time begins. The default read timeout period for a notification mailer is 10 seconds.

- **Processor Min Loop Sleep** - Specify the minimum sleep time in seconds during which the service component waits, after its read timeout period expires, before it checks its queue for messages again. The default minimum sleep time for a notification mailer is 5 seconds.

- **Processor Max Loop Sleep** - Specify the maximum sleep time in seconds if you want to increase the sleep time between read attempts when no messages are received. If you specify a maximum sleep time that is greater than the minimum sleep time, then the service component initially waits for the minimum sleep time after it finishes reading messages from its queue. If no messages are read in subsequent attempts, then the sleep time between read attempts gradually increases until the maximum sleep time is reached. Increasing the sleep time can help enhance performance if messages are received infrequently. You can also specify 0 (zero) for this parameter to indicate that the sleep time should not be increased. In this case, the service component always waits for the minimum sleep time between read attempts. The default maximum sleep time for a notification mailer is 60 seconds.

- **Processor Error Loop Sleep** - Specify the sleep time in seconds during which the service component waits, after an error occurs, before it attempts to begin processing again. The default error sleep time for a notification mailer is 60 seconds.

- **Processor Close on Read Timeout** - Select this parameter to specify that the service component should close its connections after its read timeout period expires, when its sleep time begins. Deselect this parameter to specify that the connections should remain open until the processing thread stops.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

E-mail Servers

This page lets you define e-mail server parameters for the notification mailer. Some parameters are already set to required values and cannot be modified. You must set parameters marked with an asterisk (*) to appropriate values for your environment before you can run the notification mailer. A refresh icon identifies attributes that can be refreshed dynamically while the service component is running. If the notification mailer is currently running, then parameters marked with a refresh icon will be refreshed immediately when you select the Next button.

General

- **Mailer Node Name** - The node identifier name used by this notification mailer. The maximum length for a node name is eight characters. The node name cannot

include any spaces or any of the following characters: left bracket ([), right bracket (]), slash (/), or at sign (@). The node name is included with the outgoing notification ID in outbound messages, and is used in inbound messages to identify the notification mailer that should process the messages. If you use the inbound and outbound thread count parameters to create notification mailers that are dedicated to either inbound or outbound processing, you should ensure that you assign the same node name to at least one outbound mailer and one inbound mailer, so that inbound mailer can process responses to messages sent by the outbound mailer. You can optionally assign the same node name to multiple mailers for load balancing purposes. However, each mailer that performs inbound processing for a node must have its own inbox. The default node name is WFMAIL.

> **Note:** The node name for each node must be unique. However, multiple mailers can share the same node.

If a particular notification message has the special #WFM_NODENAME message attribute defined, however, an outbound notification mailer will include the #WFM_NODENAME attribute value when sending the message, instead of the Mailer Node Name mailer parameter value.

- **Email Parser** - The Java class used to parse an incoming notification response e-mail formatted according to the templated response method and to create an XML document for the response. The notification mailer uses this parser when the Direct Response parameter is deselected. The default standard e-mail parser provided by Oracle Workflow is named oracle.apps.fnd.wf.mailer.TemplatedEmailParser. Usually you do not need to change this value.

  If you are not implementing inbound e-mail processing for this mailer, leave the default as a placeholder value.

  > **Note:** You do not need to change the value of the Email Parser parameter if you select the Direct Response parameter. The notification mailer automatically switches to the alternate e-mail parser when the Direct Response parameter is selected.

- **Alternate Email Parser** - The Java class used to parse an incoming notification response e-mail formatted according to the direct response method and to create an XML document for the response. The notification mailer uses this parser when the Direct Response parameter is selected. The default alternate e-mail parser provided by Oracle Workflow is named oracle.apps.fnd.wf.mailer.DirectEmailParser. Usually you do not need to change this value.

  If you are not implementing inbound e-mail processing for this mailer, leave the default as a placeholder value.

> **Note:** You do not need to change the value of the Alternate Email Parser parameter if you deselect the Direct Response parameter. The notification mailer automatically switches to the standard e-mail parser when the Direct Response parameter is deselected.

- **Expunge Inbox on Close** - Select this parameter to purge deleted messages from the inbox folder when the notification mailer closes this folder. If you do not select this parameter, copies of messages that were moved to the discard or processed folders remain in the inbox, in a deleted state, until you manually expunge them using your e-mail application.

Inbound E-mail Account

- **Inbound Protocol** - Oracle Workflow currently supports the IMAP protocol for inbound e-mail.

- **Inbound Server Name** - The name of the inbound mail server. Note that you must specify the actual host name for the server. Do not use `localhost` as the setting for this parameter. You can optionally specify the port number to use on that server. If you do not specify a port number, the notification mailer uses port 143 by default. Specify the server in the following format: `<server_name>[:<port_number>]`

  For example: `myimapserver.mycompany.com:143`

  If you are not implementing inbound e-mail processing for this mailer, enter a placeholder value.

- **Username** - The user name of the mail account that the notification mailer uses to receive e-mail messages.

  If you are not implementing inbound e-mail processing for this mailer, enter a placeholder value.

- **Password** - The password for the mail account specified in the Username parameter. The password value is masked as asterisks in the display and is stored in encrypted form.

  If you are not implementing inbound e-mail processing for this mailer, enter a placeholder value.

- **Inbox Folder** - The name of the folder from which the notification mailer receives inbound messages. This value is case-insensitive. The default value is `INBOX`. The inbox must be separate from the processed and discard folders. Each notification mailer that performs inbound processing should have its own separate inbox.

  > **Note:** Usually, you use a dedicated mail account for notification

mailer processing. If you want to use a mail account for the notification mailer that you also use for other purposes, you should create a folder in that account where you will place inbound messages destined for the notification mailer and specify that folder in the Inbox Folder parameter. Otherwise, the notification mailer will attempt to process all messages in the regular inbox and discard any messages that are not notification responses. If you do specify a separate folder to use as the notification mailer inbox folder, however, you must move messages from the regular inbox to that separate folder yourself. Depending on your mail program, you may be able to create a filter in the mail account to move such messages automatically. Use your e-mail client to create the separate folder. A notification mailer may not be able to access folders that were created using command line tools outside the e-mail client.

If you are not implementing inbound e-mail processing for this mailer, leave the default as a placeholder value.

- **Inbound Connection Timeout** - The maximum amount of time, in seconds, that the notification mailer will wait to establish a connection to the inbound server before timing out. The default inbound connection timeout period for a notification mailer is 120 seconds.

- **Inbound Message Fetch Size** - The maximum number of messages that the notification mailer can fetch from the inbox at one time. The default inbound message fetch size is 100 messages.

- **Maximum Ignore List Size** - The maximum number of notification IDs that the notification mailer can store in its ignore list, indicating that this notification mailer will make no further attempts to process them. For example, if the mailer encountered a connection error while processing a notification, that notification ID is temporarily added to the ignore list, and is then removed from the list the next time the inbox folder is successfully closed. The default maximum ignore list size is 1000. Usually you do not need to change this value.

    Note: If the notification mailer finds additional messages to be ignored in the inbox when the ignore list is already full, the notification mailer removes the oldest notification IDs from the list and adds the new notification IDs instead.

- **Inbound SSL Enabled** - Select this parameter to enable the notification mailer to use SSL for connections to the IMAP server. Deselect this parameter to use non-SSL connections.

> **Note:** When you enable SSL, the notification mailer connects to the IMAP server through port 993 by default. You can optionally specify a different port number along with the IMAP server name in the Inbound Server Name parameter.

Before you can use SSL, you must also complete additional setup steps. See: Connecting to Mail Servers Through SSL, page 2-56.

Outbound E-mail Account

- **Outbound Protocol** - Oracle Workflow currently supports the SMTP protocol for outbound e-mail.

- **Outbound Server Name** - The name of the outbound mail server. Note that you must specify the actual host name for the server. Do not use `localhost` as the setting for this parameter. You can optionally specify the port number to use on that server. If you do not specify a port number, the notification mailer uses port 25 by default. Specify the server in the following format: `<server_name>[:<port_number>]`

  For example: `mysmtpserver.mycompany.com:25`

  If you are not implementing outbound e-mail processing for this mailer, enter a placeholder value.

- **Test Address** - This parameter has been replaced by the override e-mail address, which is available through the Component Details page for a notification mailer. See: Reviewing Service Component Details, page 7-13.

- **Outbound Connection Timeout** - The maximum amount of time, in seconds, that the notification mailer will wait to establish a connection to the outbound server before timing out. The default outbound connection timeout period for a notification mailer is 120 seconds.

- **Outbound SSL Enabled** - Select this parameter to enable the notification mailer to use Secure Sockets Layer (SSL) for connections to the SMTP server. Deselect this parameter to use non-SSL connections.

  > **Note:** When you enable SSL, the notification mailer connects to the SMTP server through port 465 by default. You can optionally specify a different port number along with the SMTP server name in the Outbound Server Name parameter.

Before you can use SSL, you must also complete additional setup steps. See: Connecting to Mail Servers Through SSL, page 2-56.

- **Processed Folder** - The name of the mail folder where the notification mailer places successfully processed notification messages. This value is case-insensitive. The processed folder must be separate from the inbox and the discard folder.

  The default value for this parameter is `PROCESS`. If you enabled inbound processing in the Basic Configuration page and the mail account you specified did not already include a folder named `PROCESS`, Oracle Workflow automatically created a folder with this name in that account when you completed the basic notification mailer configuration.

  You can optionally specify the name of a different folder in this parameter. In this case, ensure that you use your e-mail client to create the folder. A notification mailer may not be able to access folders that were created using command line tools outside the e-mail client.

  > **Note:** The notification mailer does not perform any further operations on messages in the processed folder. You can review, back up, or delete these messages through your e-mail application if necessary.

  If you are not implementing inbound e-mail processing for this mailer, leave the default as a placeholder value.

- **Discard Folder** - The name of the mail folder where the notification mailer places incoming messages that are not recognized as notification messages. This value is case-insensitive. The discard folder must be separate from the inbox and the processed folder.

  The default value for this parameter is `DISCARD` If you enabled inbound processing in the Basic Configuration page and the mail account you specified did not already include a folder named `DISCARD`, Oracle Workflow automatically created a folder with this name in that account when you completed the basic notification mailer configuration.

  You can optionally specify the name of a different folder in this parameter. In this case, ensure that you use your e-mail client to create the folder. A notification mailer may not be able to access folders that were created using command line tools outside the e-mail client.

  > **Note:** The notification mailer does not perform any further operations on messages in the discard folder. You can review, back up, or delete these messages through your e-mail application if necessary.

  If you are not implementing inbound e-mail processing for this mailer, leave the

default as a placeholder value.

- **Allow Forwarded Response** - Indicate whether to allow a user to respond by e-mail to an e-mail notification that has been forwarded from another role. This parameter is selected by default.

  - If Allow Forwarded Response is selected, the notification mailer never checks the "From" e-mail address of the notification response and always allows the response to be processed.

  - If Allow Forwarded Response is deselected, the notification mailer will check whether the "From" e-mail address of the notification response exactly matches the e-mail address of the recorded recipient role or the e-mail address of a user in that role. If the two e-mail addresses match exactly, meaning the notification was not forwarded or was forwarded according to a valid routing rule, the notification mailer treats the response as a valid response. If the two e-mail addresses do not match exactly, meaning the notification was simply forwarded using the e-mail Forward command, the notification mailer does not process the response and treats it as unsolicited mail.

    > **Note:** Note that there are limitations when you deselect Allow Forwarded Response. For example, suppose a notification is sent to a distribution list mail alias that does not have a user/role relationship in the Oracle Workflow directory service. If any user from the distribution list responds to the notification, the notification mailer will always treat that notification response as unsolicited mail, because the "From" e-mail address, which is an individual user's e-mail address, will never match the distribution list mail alias.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

> **Note:** When you click the Next button, the configuration wizard validates the parameters you entered. If the inbound thread count is set to 1, the configuration wizard also verifies that it can connect to the e-mail account on the specified inbound mail server with the specified user name and password, and that the folders specified in the Processed Folder and Discard Folder parameters exist in that e-mail account. If the parameters are successfully validated, and the notification mailer is currently running, then Oracle Workflow Manager immediately refreshes the notification mailer with the new

parameters.

Message Generation

This page lets you define message generation parameters for the notification mailer. Some parameters are already set to required values and cannot be modified. You must set parameters marked with an asterisk (*) to appropriate values for your environment before you can run the notification mailer. A refresh icon identifies attributes that can be refreshed dynamically while the service component is running. If the notification mailer is currently running, parameters marked with a refresh icon will be refreshed immediately when you select the Next button or the Finish button.

Send

- **From** - A value that appears in the From field of the message header of a notification e-mail. You can specify the From parameter value either as a display name only, or as a full RFC822-compliant address.

  - If you specify a display name only, the notification mailer adds the e-mail address from the Reply-to Address parameter to create a full RFC822-compliant address for the From message header. The full address is created in the following format: **"***Display Name***"** *<reply_to_address>*

  - If you specify a full RFC822-compliant address, the notification mailer uses only that From parameter value in the From message header, and does not include the Reply-to Address value.

  If a particular notification message has the special #WFM_FROM message attribute defined, however, the notification mailer will use the #WFM_FROM attribute value in the From field for that message, instead of the From parameter value.

  The default From parameter value for the seeded notification mailer service component is `Workflow Mailer`. For other notification mailers, if you selected the Inbound Processing parameter in the Basic Configuration page, Oracle Workflow by default sets the From parameter to the name portion of the reply-to address specified in the Basic Configuration page. For example, if the reply-to address is `Workflow@mycompany.com`, Oracle Workflow sets the From parameter to `Workflow`.

  If you deselected the Inbound Processing parameter in the Basic Configuration page, Oracle Workflow by default sets the From parameter to `nobody@` *<server_name>*, where *<server_name>* is the name of the outbound SMTP mail server specified in the Basic Configuration page.

  If you are not implementing outbound e-mail processing for this mailer, leave the default as a placeholder value.

- **Reply-to Address** - The address of the e-mail account that receives incoming

messages, to which notification responses should be sent. This value must be a full RFC822-compliant e-mail address.

If a particular notification message has the special #WFM_REPLYTO message attribute defined, however, the notification mailer will use the #WFM_REPLYTO attribute value as the reply address for that message, instead of the Reply-to Address parameter value.

> **Note:** If the From parameter value is specified as a display name only, then the notification mailer also uses the reply-to e-mail address together with that display name to create a full RFC822-compliant address for the From field of the message header.

If you deselected the Inbound Processing parameter in the Basic Configuration page, Oracle Workflow by default sets the Reply-to Address parameter to `nobody@ <server_name>`, where `<server_name>` is the name of the outbound SMTP mail server specified in the Basic Configuration page. If you are not implementing inbound e-mail processing for this mailer, leave the default as a placeholder value.

- **HTML Agent** - The base URL that identifies the HTML agent that handles HTML notification responses. This URL is required to support e-mail notifications with HTML attachments. Usually the HTML agent specified here can match the value of the Applications Servlet Agent profile option; however, you can optionally specify a different HTML agent for a particular notification mailer. The HTML agent should be specified in the following format:

`http://<server_name:port>/OA_HTML/`

where `<server_name:port>` represents the server and TCP/IP port number on which your servlet agent accepts requests.

> **Note:** The notification mailer can also still handle an HTML agent value in the previous format:
>
> `http://<server_name:port>/pls/wf`

If a particular notification message has the special #WFM_HTMLAGENT message attribute defined, however, the notification mailer will use the #WFM_HTMLAGENT attribute value as the HTML agent for that message, instead of the HTML Agent mailer parameter value.

- **Message Formatter** - Oracle Workflow uses the oracle.apps.fnd.wf.mailer.NotificationFormatter Java class to generate notification messages.

- **Framework User** - The numerical user ID for the user through which a notification mailer accesses Oracle Application Framework content for inclusion in e-mail

notifications. The Framework user must have workflow administrator privileges in order to access the content for every user's notifications.

The default value for this parameter is 0, which is the user ID for the SYSADMIN user. This setting lets the notification mailer access Oracle Application Framework content through the SYSADMIN user, which is also the default workflow administrator role. If you change the Workflow System Administrator preference, check the Framework User parameter to ensure that the user accessed by the notification mailer has workflow administrator privileges. Set the Framework User parameter to a user that is a member of the Workflow System Administrator role, or, if you set the Workflow System Administrator preference to a responsibility, set the Framework User parameter to a user that has that responsibility. See: Setting Global User Preferences, page 2-9.

- **Framework Responsibility** - The numerical responsibility ID for the responsibility through which a notification mailer accesses Oracle Application Framework content for inclusion in e-mail notifications. The user specified in the Framework User parameter must have this responsibility assigned. The default value for this parameter is 20420, which is the responsibility ID for the System Administrator responsibility.

- **Framework Application ID** - The numerical application ID for the application through which a notification mailer accesses Oracle Application Framework content for inclusion in e-mail notifications. The responsibility specified in the Framework Responsibility parameter must be assigned to this application. The default value for this parameter is 1, which is the application ID for the System Administration application.

- **Framework URL Timeout** - The maximum amount of time, in seconds, that the notification mailer will wait to access a URL for Oracle Application Framework content before timing out. The default Framework URL timeout period for a notification mailer is 30 seconds.

- **Attach Images to Outbound Emails** - Select this parameter to attach any images referenced in HTML content included in a message, such as Oracle Application Framework content, to outbound notification e-mail messages. Deselect this parameter to display the image references as off-page URLs instead of attaching the images.

- **Attach Stylesheet to Outbound Email** - Select this parameter to attach any stylesheet referenced in HTML content included in a message, such as Oracle Application Framework content, to outbound notification e-mail messages. Deselect this parameter to display the stylesheet reference as a URL instead of attaching the stylesheet.

    Note: E-mail clients vary in their support for stylesheet references

within HTML content in the body of an e-mail. Some e-mail clients do not support references to a stylesheet that is attached to the e-mail, while others do not support any form of stylesheet references within HTML content at all. Consequently, attaching a stylesheet may not have the same effect in all e-mail clients.

- **Autoclose FYI** - Indicate whether this notification mailer automatically closes notifications that do not require a response, such as FYI (For Your Information) notifications, after sending the notifications by e-mail. This parameter is selected by default. If Autoclose FYI is deselected, all FYI notifications will remain open in the Worklist until users manually close these notifications.

- **Direct Response** - By default, notification mailers require a response format for plain text notifications called the templated response method. Select this parameter to use the direct response method instead.

  - With the templated response method, a notification mailer sends plain text notifications requiring a templated response to users with a notification preference of `MAILTEXT` or `MAILATTH`. Users must reply using a template of response prompts and enter their response values between the quotes following each prompt.

  - With the direct response method, a notification mailer sends plain text notifications requiring a direct response to users with a notification preference of `MAILTEXT` or `MAILATTH`. Users must enter their response values directly as the first lines of a reply.

    Note: Responses that are generated automatically from an HTML-formatted notification or attachment must always use a response template, regardless of which response method you select.

  See: Workflow Open Mail (Templated) Message, page 2-86, Workflow Open Mail (Direct) Message, page 2-90, To Respond to a Plain Text E-mail Notification Using Templated Response, *Oracle Workflow User's Guide*, To Respond to a Plain Text E-mail Notification Using Direct Response, *Oracle Workflow User's Guide*, and Example 'Respond' Message Attributes, *Oracle Workflow Developer's Guide*.

- **Reset NLS** - Indicate whether the notification mailer should convert the NLS codeset for a notification message according to the notification recipient's preferences before composing the message. This parameter is deselected by default. If Reset NLS is selected, the notification mailer will convert the message to the codeset listed in the WF_LANGUAGES table for the language and territory specified in the recipient's user preferences. If no preferred territory is specified, the notification mailer will use the codeset associated with the first entry it encounters

for the user's preferred language. If neither a language nor a territory is specified in the user preferences, the notification mailer will use the codeset seeded in WF_LANGUAGES for the language AMERICAN and territory AMERICA. This parameter is relevant when there are several languages installed in the database and the character set of the user's e-mail client is not the same as the one specified for the database. For example, when a UTF8 database is used, the character set of e-mail clients used in Western Europe is generally 'Western (ISO-8859-1)'. In this case, selecting the Reset NLS parameter means that users who specify a Western European language such as French or German in their user preferences will receive any e-mail notification messages in the correct character set for the e-mail client.

If a particular notification message has the special #WFM_RESET_NLS message attribute defined, however, the notification mailer will use the #WFM_RESET_NLS attribute value to determine whether or not to encode the e-mail to the preferred codeset for that message, instead of using the Reset NLS parameter value.

- **Inline Attachments** - Select this parameter to set the Content-Disposition MIME header to `inline` for attachments to notification messages, including the Notification Detail Link, HTML Message Body, Notification References containing attached URLs, and attached PL/SQL documents. Deselect this parameter to set the Content-Disposition MIME header to `attachment` for these attachments. For example, if your e-mail application cannot display HTML content such as the Notification Detail Link inline, deselect this parameter to display this link as an attachment instead. Note, however, that some e-mail clients may not support the Content-Disposition header, or may support it in varying ways. Consequently, the Inline Attachment setting may not always have the desired effect, depending on the e-mail clients with which users read their e-mail messages.

- **Send Warning for Unsolicited E-mail** - Select this parameter to allow the notification mailer to send back a warning message the first time it receives an unsolicited e-mail message from a particular e-mail address. Deselect this parameter to prevent the notification mailer from sending warning messages.

- **Send E-mails for Canceled Notifications** - Select this parameter to allow the notification mailer to send cancellation messages to users when previously sent notifications are canceled. Deselect this parameter to prevent the notification mailer from sending cancellation messages.

  If you set up multiple notification mailers in the same Oracle Applications instance, you must set this parameter to the same setting for all the notification mailers.

Templates

This region lets you specify the message templates you want to use to generate e-mail notifications. The template for a particular type of e-mail notification determines the basic format of the notification, including what header information to include, and whether and where to include details such as the message due date and priority.

Oracle Workflow provides a set of standard templates in the System: Mailer item type, which are used by default. It is not recommended to modify the standard templates. However, you can customize the message templates used to send your e-mail notifications by creating your own custom message templates in the System: Mailer item type using the Workflow Builder, and assigning these templates to a particular notification mailer service component in this region. The templates assigned to a mailer override the default System: Mailer templates.

Additionally, you can create your own custom message templates in a custom item type using the Workflow Builder, and assign these templates to a particular notification in a workflow process by defining special message attributes. In this case the templates assigned to the notification override both the templates assigned to a mailer and the default System: Mailer templates.

If you are not implementing outbound e-mail processing for this mailer, leave the default templates as placeholder values.

- **Attached URLs** - The notification mailer uses this template to create the Notification References attachment for HTML-formatted notification messages that include URL attributes with Attach Content checked. The template must includes a list with links to each URL.

- **Outbound Closed Notification** - The notification mailer uses this template to inform the recipient that a previously sent notification is now closed.

- **Outbound Cancelled Notification** - The notification mailer uses this template to inform the recipient that a previously sent notification is canceled. You can optionally use the Send E-mails for Canceled Notifications parameter to specify whether or not the notification mailer should send Outbound Cancelled Notification messages.

- **Invalid Response Notification** - The notification mailer uses this template to inform a user that the user responded incorrectly to a notification. For example, if a response message from a user contains a valid notification ID (NID) line matching it with a notification, but does not contain any response value or contains an invalid response value, the notification mailer sends an Invalid Response Notification message to the user. This template must describe how to respond to the notification correctly.

- **Open Notification** - If you are using the default response method, which is templated response, the notification mailer uses this template to send open notifications that require a response. This message template must provide a response template for the recipient as well as instructions on how to use the response template.

> **Note:** In addition to the default Workflow Open Mail (Templated) message template, Oracle Workflow also provides a predefined

template called Workflow Open Mail (Outlook Express). This template is provided to accommodate e-mail applications such as Microsoft Outlook Express or other e-mail clients that cannot process the response links included in the HTML bodies of the Workflow Open Mail (Templated) and Workflow Open Mail (Direct) templates. If you use one of these e-mail clients, you can select the Workflow Open Mail (Outlook Express) message template to have HTML e-mail notifications include a link to the Notification Details Web page which lets users respond to the notification there.

If you are configuring this notification mailer for outbound message processing only and you are not implementing any corresponding inbound e-mail response processing, then you should set the Open Notification parameter to a message template that does not request a response by e-mail, but instead directs recipients to respond from the Notification Details Web page. For example, you can select the Workflow View From UI message template provided by Oracle Workflow, or create your own custom message template.

If you selected the Inbound Processing parameter in the Basic Configuration page, the Open Notification parameter is set to the Workflow Open Mail (Templated) message template by default. If you deselected the Inbound Processing parameter, the Open Notification parameter is set to the Workflow Open Mail (Outlook Express) message template by default.

> **Note:** The plain text version of the Workflow Open Mail (Outlook Express) message requests a response by e-mail. If you disable inbound processing, ensure that your users do not have a notification preference of MAILTEXT or MAILATTH. Alternatively, if you disable inbound processing and you want users to receive plain text notifications, specify a message template that directs recipients to respond from the Notification Details Web page.

- **Open Notification (Direct Response Parsing)** - If you select the Direct Response parameter, the notification mailer uses this template to send open notifications that require a response. The response instructions in the plain text message body must describe how to reply using the direct response method. This message is used for notifications sent to performers with a notification preference of MAILTEXT or MAILATTH. The response instructions in the HTML-formatted message body must describe how to reply using the automatically generated response template. This message is used for notifications sent to performers with a notification preference of MAILHTML or MAILHTM2, and is also attached to notifications sent to performers with a notification preference of MAILATTH.

> **Note:** Responses that are generated automatically from an HTML-formatted notification or attachment must always use a response template, regardless of which response method you select.

> **Note:** If you are configuring this notification mailer for outbound message processing only and you are not implementing any corresponding inbound e-mail response processing, then you should set the Open Notification (Direct Response Parsing) parameter to a message template that does not request a response by e-mail, but instead directs recipients to respond from the Notification Details Web page. For example, you can select the Workflow View From UI message template provided by Oracle Workflow, or create your own custom message template.

See: Workflow Open Mail (Templated) Message, page 2-86, Workflow Open Mail (Direct) Message, page 2-90, To Respond to a Plain Text E-mail Notification Using Templated Response, *Oracle Workflow User's Guide*, To Respond to a Plain Text E-mail Notification Using Direct Response, *Oracle Workflow User's Guide*, and Example 'Respond' Message Attributes, *Oracle Workflow Developer's Guide*.

- **Open FYI Notification** - The notification mailer uses this template to send notifications that do not require a response. The template must indicate that the notification is for your information (FYI) and does not require a response.

- **Outbound Summary Notification** - This template is no longer used.

- **Outbound Warning Notification** - The notification mailer uses this template to send a message to a user the first time it receives unsolicited mail from that user. For example, if a message from a user does not contain a notification ID (NID) line matching it with a notification, or contains an incorrectly formatted NID line, the notification mailer sends an Outbound Warning Notification message to the user. You can optionally use the Send Warning for Unsolicited E-mail parameter to specify whether or not the notification mailer should send Outbound Warning Notification messages.

- **Open Notification (More Information Request)** - The notification mailer uses this template to send a request for more information about a notification from one user to another user.

  > **Note:** If you use an e-mail application such as Microsoft Outlook Express that cannot process the response link included in the default Workflow Open Mail (More Information Request) message template, you can select an alternative template named Workflow

More Information Request (Outlook Express) instead. In particular, if you set the Open Notification parameter to use the Workflow Open Mail (Outlook Express) message, then you should also set the Open Notification (More Information Request) parameter to use the Workflow More Information Request (Outlook Express) message.

- **Outbound HTML Summary Notification** - The notification mailer uses this template to send a summary of currently open workflow notifications to users and roles that have their notification preference set to SUMMARY or SUMHTML in the Oracle Workflow directory service.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

> **Note:** When you click the Next or Finish button, the configuration wizard validates the parameters you entered. If the parameters are successfully validated, and the notification mailer is currently running, then Oracle Workflow Manager immediately refreshes the notification mailer with the new parameters.

Scheduling Events

This page lets you schedule events to control the running of the service component. The events are raised at the scheduled time by DBMS jobs. For a notification mailer service component, you can schedule the following events:

- Start

- Refresh

- Suspend

- Resume

- Stop

- Launch Summary Notifications

For each event, the list displays the event name, date and time when the event is first scheduled to be raised, the interval in minutes at which the event is reraised, and, for a

Refresh event, any parameters to be refreshed. You can specify the following refreshable parameters, using the parameters' internal names, when you refresh the notification mailer.

- `PROCESSOR_IN_THREAD_COUNT` - Inbound Thread Count

- `PROCESSOR_OUT_THREAD_COUNT` - Outbound Thread Count

- `COMPONENT_LOG_LEVEL` - Log Level, specified as a numerical value

    - `1` - Statement

    - `2` - Procedure

    - `3` - Event

    - `4` - Exception

    - `5` - Error

    - `6` - Unexpected

- `EXPUNGE_ON_CLOSE` - Expunge Inbox on Close

- `ALLOW_FORWARDED_RESPONSE` - Allow Forwarded Response

- `FROM` - From

- `REPLYTO` - Reply-to Address

- `HTMLAGENT` - HTML Agent

- `ATTACH_IMAGES` - Attach Images to Outbound E-mails

- `ATTACH_STYLESHEET` - Attach Stylesheet to Outbound E-mail

- `AUTOCLOSE_FYI` - Autoclose FYI

- `RESET_NLS` - Reset NLS

- `INLINE_ATTACHMENT` - Inline Attachments

- `SEND_UNSOLICITED_WARNING` - Send Warning for Unsolicited E-mail

- `ATTACHED_URLS` - Attached URLs

- `CLOSED` - Outbound Closed Notification

- `CANCELED` - Outbound Cancelled Notification

- OPEN_INVALID - Invalid Response Notification

- OPEN_MAIL - Open Notification

- OPEN_MAIL_DIRECT - Open Notification (Direct Response Parsing)

- OPEN_MAIL_FYI - Open FYI Notification

- SUMMARY - Outbound Summary Notification

- WARNING - Outbound Warning Notification

- OPEN_MORE_INFO - Open Notification (More Information Request)

- SUMHTML - Outbound HTML Summary Notification

To schedule events:

- If no events are currently scheduled, click the Add a Row button to add a new row to the list of events and enter the information for the event.

    - Select the event for the command you want to schedule.

    - Select the date when you want the event to be raised first.

    - Select the hour and minute to specify the time on the specified date when you want the event to be raised first. The hour values are in a twenty-four hour format. For example, select 00 for midnight, or 23 for 11 PM.

    - If you want to raise the event periodically, enter the time interval in minutes at which you want to raise the event. If you do not specify a repeating interval, the event is raised only once.

    - If you choose the refresh event, you can optionally enter any parameters you want to include with the event in order to refresh the notification mailer configuration parameters with those values when the event is raised. Specify the parameter names and values in the following format, separating the parameters with a colon (:):
    internal_parameter_name=parameter_value

    For example: PROCESSOR_OUT_THREAD_COUNT=3

- To schedule another event, click the Add Another Row button and enter the information for the event.

- To remove an event, select the event and click the Remove button.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

> **Note:** The configuration wizard verifies that an event is specified for every row in the list when you click the Next or Finish button. If you do not want to schedule another event, remove any empty rows before proceeding.

Tags

This page lets you enter patterns of text found in unusual messages and the status you want to assign to an inbound message if it contains any of those patterns. For example, unusual messages include bounced or returned messages and auto-reply messages such as those sent by vacation daemons, mass mailing lists, and so on. Since different mail systems vary in how they identify bounced, undeliverable, or otherwise invalid messages, you can use notification mailer tags to specify how your mail system identifies those stray messages and how you want the notification mailer to handle those messages should it come across them.

Oracle Workflow provides several predefined tags for text commonly found in undeliverable or auto-reply messages. For each tag, the list displays the pattern, which is the string of text to look for in the From line, Subject line, or body of the message, and the action, which is the mail status to assign to the message if that pattern is found. The notification mailer handles messages according to these mail status values, as follows:

- UNDELVRD - Moves the message to the discard folder and updates the notification's mail status to FAILED. Additionally, the notification preference of the recipient of the notification is updated to DISABLED. No error process is initiated for this notification activity. However, after correcting the issues that prevented the e-mail from being sent, you can reset the user's notification preference and then run the Resend Failed Workflow Notifications program to re-enqueue failed notifications on the notification mailer's outbound queue. See: Handling Mailer Errors, page 2-73.

- Unavailable - Moves the message to the discard folder and continues waiting for a reply to the notification since the notification's status is still OPEN, but its mail status is updated to UNAVAIL. This status is purely informative, as no further processing occurs with this notification.

- Ignore - Moves the message to the discard folder and continues waiting for a valid reply to the open notification. The notification's status is still OPEN and its mail status is still SENT.

You can define additional tags for other patterns you want the notification mailer to handle automatically.

- To add a new tag, click the Add Another Row button, enter the text pattern in the Pattern column, and select the status you want to assign to messages containing that pattern in the Action column.

- To remove a tag, select the tag and click the Remove button. You can only remove custom tags that you defined. You cannot remove predefined tags provided by Oracle Workflow.

   Note: It is important that you uniquely identify bounced messages and auto-replies by defining tags to distinguish them from normal responses. If you do not identify bounced and auto-reply messages, the notification mailer can mistake these as invalid responses, send an Invalid Response Notification message, and continue to wait for a reply. In both cases a perpetual loop would occur where the notification mailer continues sending out an 'Invalid' message and the 'Invalid' message bounces back or is auto-replied each time.

   Note: Only a message response that contains a notification ID can be handled through the FAILED and UNAVAIL mail statuses. If the notification mailer receives a message response that does not contain a notification ID, it moves the message response to the discard folder. If the Send Warning for Unsolicited E-mail parameter is selected, then for the first such message from a particular e-mail address, the notification mailer also sends an Outbound Warning Notification message to the sender to warn that it received unsolicited mail.

   Note: If a message response matches more than one pattern in the list of tags, the message is tagged with the status of the first tag it matches. That is, the notification mailer performs a top to bottom comparison against the tag list. Due to this behavior, you should prioritize your patterns listing the UNDELVRD tags first, followed by the Unavailable and then Ignore tags.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

Test

This page lets you test the configuration for a notification mailer that performs

outbound e-mail processing by sending a sample notification message. Select the recipient role to which the message should be sent, and click the Send Test Message button. Then check the e-mail account for the recipient role to verify that the test message was received. The test message does not require a response, so you can close it after reviewing it. However, you can optionally respond with a comment to acknowledge the message.

To send a test message successfully, you must select a recipient role that either has a valid e-mail address defined, or that has members with valid e-mail addresses defined. The recipient role must also have a notification preference that includes individual e-mail notifications.

If you set an override e-mail address for the notification mailer, the Test page displays that address. In this case the test message is sent to the override address rather than the e-mail address of the recipient role. However, you must still select a recipient role to enable the notification mailer to send the test message. See: Reviewing Service Component Details, page 7-13.

To exit the advanced configuration wizard, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To proceed to the next step of the configuration wizard, click the Next button.

To proceed to the last step of the configuration wizard, click the Finish button.

Review

This page lets you review the configuration parameter values that you set, the events that you scheduled, and the tags that you defined for this notification mailer service component.

- If you want to change any of these settings, return to the appropriate step in the configuration wizard to make your changes. To return to the previous step, click the Back button.

- To save these settings and finish the configuration, click the Finish button.

Agent Listeners

The Oracle Workflow Business Event System requires agent listeners to be scheduled to receive inbound event messages. An agent listener monitors a Business Event System agent for incoming messages and dequeues messages using the agent's queue handler. You should run agent listeners for your local inbound agents. Run PL/SQL agent listeners to process event subscriptions with a PL/SQL rule function in the database, and run Java agent listeners to process event subscriptions with a Java rule function in the middle tier.

When an event message is dequeued, the Event Manager begins subscription processing for the event. The Event Manager searches for and executes any active

subscriptions by the local system to that event with a source type of External, and also any active subscriptions by the local system to the Any event with a source type of External. The agent listener exits after all event messages on the agent's queue have been dequeued.

The PL/SQL agent listener program is defined as a service component type in the Generic Service Component Framework. This framework helps to simplify and automate the management of background Java services.

Oracle Workflow provides several seeded agent listener service components to process messages on standard agents.

- Workflow Deferred Agent Listener - Handles messages on WF_DEFERRED to support deferred subscription processing. This service component is started automatically by its container.

- Workflow Deferred Notification Agent Listener - Handles notification messages on WF_DEFERRED to support outbound notification processing. This service component is started automatically by its container.

- Workflow Error Agent Listener - Handles messages on WF_ERROR to support error handling for the Business Event System. This service component is started automatically by its container.

- Workflow Inbound Notifications Agent Listener - Handles messages on WF_NOTIFICATION_IN to support inbound e-mail notification processing. This service component is started automatically by its container.

- ECX Inbound Agent Listener - Handles message on ECX_INBOUND to support Oracle XML Gateway processing. This service component must be started manually. For more information, see the *Oracle XML Gateway User's Guide*.

- ECX Transaction Agent Listener - Handles message on ECX_TRANSACTION to support Oracle XML Gateway processing. This service component must be started manually. For more information, see the *Oracle XML Gateway User's Guide*.

You cannot delete the seeded agent listeners or edit their names, assigned agents, correlation ID values, or containers. However, if necessary you can update other configuration parameters, schedule control events, or manually choose control commands to start, stop, suspend, resume, or refresh the agent listeners.

You can also optionally create additional agent listener service components. For example, you can configure agent listeners for other inbound agents that you want to use for event message propagation, such as the standard WF_IN and WF_JMS_IN agents, or any custom agents. You can also configure an agent listener that only processes messages on a particular agent that are instances of a specific event.

If you create custom agent listener service components, you can either assign them to the seeded container for agent listeners, named Workflow Agent Listener Service, or, based on the volume to be handled by the seeded container, you can also choose to

create your own custom containers.

Before the seeded agent listener service components can run, the Workflow Agent Listener Service container which manages them must be first be started. You should ensure that this container is running. If you create your own custom containers for custom service components, ensure that those containers are running as well. Use the Service Instances page to start each container as a service instance in Generic Service Management (GSM). When the Workflow Agent Listener Service container is running, it automatically starts the Workflow Deferred Agent Listener, Workflow Deferred Notification Agent Listener, Workflow Error Agent Listener, and Workflow Inbound Notifications Agent Listener.

Agent Listener Configuration Wizard

The agent listener configuration wizard lets you configure an agent listener service component by defining general and detail attributes and scheduling control events. You can use the configuration wizard to configure a new agent listener service component, or to edit the configuration of an existing agent listener service component.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > (B) Create > (B) Continue

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > (B) Edit

Define

This page lets you define general attributes for the service component. Some attributes are already set to required values and cannot be modified. You must set attributes marked with an asterisk (*) to appropriate values for your environment before you can run the service component.

- **ID** - When you edit a previously created service component, the configuration wizard displays the identifier for the service component.

- **Status** - When you edit a previously created service component, the configuration wizard displays the status of the service component.

- **Name** - The name of the service component. This name must be unique.

- **Startup Mode** - Select Automatic, Manual, or On-Demand as the startup mode for the service component.

- **Container Type** - The container type to which this service component belongs, which is always Oracle Applications Generic Service Management (Oracle Applications GSM).

- **Inbound Agent** - The Business Event System agent that you want to monitor for inbound event messages.

- **Outbound Agent** - Leave this field blank. Agent listener service components do not use an outbound agent.

- **Correlation ID** - Optionally specify the Oracle Advanced Queuing (AQ) correlation ID of the event messages that you want the agent listener to process. The AQ correlation ID for an event message in the Business Event System is usually specified in the following format:

  ```
  <event name>
  ```

  Consequently, by specifying a correlation ID in this attribute, you can dedicate the agent listener to listen only for messages that are instances of the specified event. You can also specify a partial value to listen for messages that are instances of any event whose name begins with the specified value.

  For example, the seeded Workflow Deferred Notification Agent Listener has an AQ correlation ID of `oracle.apps.wf.notification.%`, meaning that this agent listener handles only notification event messages on the WF_DEFERRED agent. However, the Workflow Deferred Agent Listener, Workflow Error Agent Listener, and Workflow Inbound Notifications Agent Listener do not have any correlation ID specified so that they can process all event messages on their respective agents.

  > **Note:** The AQ correlation ID is different than the correlation ID contained within the WF_EVENT_T event message structure.

To cancel the configuration without saving any changes, click the Cancel button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

Details

This page lets you define detail attributes for the service component. You must set attributes marked with an asterisk (*) to appropriate values for your environment before you can run the service component. A refresh icon identifies attributes that can be refreshed dynamically while the service component is running.

- **ID** - When you edit a previously created service component, the configuration wizard displays the identifier for the service component.

- **Status** - When you edit a previously created service component, the configuration wizard displays the status of the service component.

- **Name** - The configuration wizard displays the name defined for the service component.

- **Container** - The container to which the service component will belong. Oracle Workflow provides a container called Workflow Agent Listener Service for agent listener service components.

- **Maximum Idle Time** - If you selected the On-Demand startup mode for the service component, enter the maximum time in minutes that the service component can remain idle before it is stopped. An on-demand component that is stopped in this way will be restarted by its container when it is needed again to process new messages.

- **Max Error Count** - The number of consecutive errors the service component can encounter before its container stops it and changes its status to Stopped with Error. If an error is resolved and processing can continue, the error count is reset. The default value for the maximum error count is 10.

- **Inbound Thread Count** - Set the inbound processing thread count to 1 (one) to enable inbound message processing with this agent listener. Select 0 (zero) to disable this agent listener. The default value is 1.

- **Outbound Thread Count** - Leave this parameter set to the default value of 0 (zero). Agent listener service components do not perform outbound message processing.

- **Log Level** - Select the level of detail for the information you want to record in the service component container log. The recommended log level, which is also the default value, is Error. Usually the log level only needs to be changed if you want to record additional detailed information for debugging purposes. You can choose the following levels:

  - 1 - Statement

  - 2 - Procedure

  - 3 - Event

  - 4 - Exception

  - 5 - Error

  - 6 - Unexpected

- **Processor Read Wait Timeout** - Specify the amount of time in seconds that the service component's processing thread continues to wait, after reading the last message from its assigned queue, before timing out. If another message is received before this time expires, that message is processed and the timeout period begins again. If the timeout period expires and no more messages have been received, the service component stops reading and its sleep time begins. The default read timeout period for an agent listener is 0 (zero) seconds.

- **Processor Min Loop Sleep** - Specify the minimum sleep time in seconds during which the service component waits, after its read timeout period expires, before it checks its queue for messages again. The default minimum sleep time for an agent

listener is 120 seconds.

- **Processor Max Loop Sleep** - Specify the maximum sleep time in seconds if you want to increase the sleep time between read attempts when no messages are received. If you specify a maximum sleep time that is greater than the minimum sleep time, then the service component initially waits for the minimum sleep time after it finishes reading messages from its queue. If no messages are read in subsequent attempts, then the sleep time between read attempts gradually increases until the maximum sleep time is reached. Increasing the sleep time can help enhance performance if messages are received infrequently. You can also specify 0 (zero) for this parameter to indicate that the sleep time should not be increased. In this case, the service component always waits for the minimum sleep time between read attempts. The default value for an agent listener is 0 (zero).

- **Processor Error Loop Sleep** - Specify the sleep time in seconds during which the service component waits, after an error occurs, before it attempts to begin processing again. The default error sleep time for an agent listener is 60 seconds.

- **Processor Close on Read Timeout** - Select this parameter to specify that the service component should close its connections after its read timeout period expires, when its sleep time begins. Deselect this parameter to specify that the connections should remain open until the processing thread stops.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

Scheduling Events

This page lets you schedule events to control the running of the service component. The events are raised at the scheduled time by DBMS jobs. For an agent listener service component, you can schedule the following events:

- Start

- Refresh

- Suspend

- Resume

- Stop

For each event, the list displays the event name, date and time when the event is first

scheduled to be raised, the interval in minutes at which the event is reraised, and, for a refresh event, any parameters to be refreshed. You can specify the following refreshable parameters, using the parameters' internal names, when you refresh the agent listener.

- `PROCESSOR_IN_THREAD_COUNT` - Inbound Thread Count

- `COMPONENT_LOG_LEVEL` - Log Level, specified as a numerical value

  - 1 - Statement

  - 2 - Procedure

  - 3 - Event

  - 4 - Exception

  - 5 - Error

  - 6 - Unexpected

To schedule events:

- If no events are currently scheduled, click the Add a Row button to add a new row to the list of events and enter the information for the event.

  - Select the event for the command you want to schedule. Oracle Workflow provides events to let you start, stop, refresh, suspend, or resume the service component.

  - Select the date when you want the event to be raised first.

  - Select the hour and minute to specify the time on the specified date when you want the event to be raised first. The hour values are in a twenty-four hour format. For example, select 00 for midnight, or 23 for 11 PM.

  - If you want to raise the event periodically, enter the time interval in minutes at which you want to raise the event. If you do not specify a repeating interval, the event is raised only once.

  - If you choose the refresh event, you can optionally enter any parameters you want to include with the event in order to refresh the agent listener configuration parameters with those values when the event is raised. Specify the parameter names and values in the following format, separating the parameters with a colon (:):
    `internal_parameter_name=parameter_value`

    For example: `PROCESSOR_IN_THREAD_COUNT=1`

- To schedule another event, click the Add Another Row button and enter the

information for the event.

- To remove an event, select the event and click the Remove button.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

> **Note:** The configuration wizard verifies that an event is specified for every row in the list when you click the Next or Finish button. If you do not want to schedule another event, you should remove any empty rows before proceeding.

Review

This page lets you review the configuration parameter values that you set and the events that you scheduled for this service component.

- If you want to change any of these settings, return to the appropriate step in the configuration wizard to make your changes. To return to the previous step, click the Back button.

- To save these settings and finish the configuration, click the Finish button.

Java Agent Listeners

The Oracle Workflow Business Event System requires agent listeners to be scheduled to receive inbound event messages. An agent listener monitors a Business Event System agent for incoming messages and dequeues messages using the agent's queue handler. You should run agent listeners for your local inbound agents. Run PL/SQL agent listeners to process event subscriptions with a PL/SQL rule function in the database, and run Java agent listeners to process event subscriptions with a Java rule function in the middle tier.

When an event message is dequeued, the Event Manager begins subscription processing for the event. The Event Manager searches for and executes any active subscriptions by the local system to that event with a source type of External, and also any active subscriptions by the local system to the Any event with a source type of External. The agent listener exits after all event messages on the agent's queue have been dequeued.

The Java agent listener program is defined as a service component type in the Generic

Service Component Framework. This framework helps to simplify and automate the management of background Java services.

Oracle Workflow provides several seeded Java agent listener service components to process messages on standard agents.

- Workflow Java Deferred Agent Listener - Handles messages on WF_JAVA_DEFERRED to support deferred subscription processing in the middle tier. This service component is started automatically by its container.

- Workflow Java Error Agent Listener - Handles messages on WF_JAVA_ERROR to support error handling for the Business Event System in the middle tier. This service component is started automatically by its container.

- Web Services IN Agent - Handles messages on WF_WS_JMS_IN to support inbound Web service message processing. This service component must be started manually.

You can optionally update the configuration of the Workflow WebServices In listener or delete this service component if necessary. You cannot delete the Workflow Java Deferred Agent Listener and Workflow Java Error Agent Listener or edit their names, assigned agents, correlation ID values, or containers. However, if necessary you can update other configuration parameters, schedule control events, or manually choose control commands to start, stop, suspend, resume, or refresh these Java agent listeners.

You can also optionally create additional Java agent listener service components. For example, you can configure Java agent listeners for other inbound agents that you want to use for event message propagation in the middle tier, such as custom agents. You can also configure a Java agent listener that only processes messages on a particular agent that are instances of a specific event.

If you create custom Java agent listener service components, you can either assign them to the seeded container for agent listeners, named Workflow Agent Listener Service, or, based on the volume to be handled by the seeded container, you can also choose to create your own custom containers.

Before the seeded Java agent listener service components can run, the Workflow Agent Listener Service container which manages them must be first be started. You should ensure that this container is running. If you create your own custom containers for custom service components, ensure that those containers are running as well. Use the Service Instances page to start each container as a service instance in Generic Service Management (GSM). When the Workflow Agent Listener Service container is running, it automatically starts the Workflow Java Deferred Agent Listener and Workflow Java Error Agent Listener.

Java Agent Listener Configuration Wizard

The Java agent listener configuration wizard lets you configure a Java agent listener service component by defining general and detail attributes and scheduling control

events. You can use the configuration wizard to configure a new Java agent listener service component, or to edit the configuration of an existing Java agent listener service component.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > (B) Create > (B) Continue

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > (B) Edit

Define

This page lets you define general attributes for the service component. Some attributes are already set to required values and cannot be modified. You must set attributes marked with an asterisk (*) to appropriate values for your environment before you can run the service component.

- **ID** - When you edit a previously created service component, the configuration wizard displays the identifier for the service component.

- **Status** - When you edit a previously created service component, the configuration wizard displays the status of the service component.

- **Name** - The name of the service component. This name must be unique.

- **Startup Mode** - Select Automatic, Manual, or On-Demand as the startup mode for the service component.

- **Container Type** - The container type to which this service component belongs, which is always Oracle Applications Generic Service Management (Oracle Applications GSM).

- **Inbound Agent** - The Business Event System agent that you want to monitor for inbound event messages.

- **Outbound Agent** - Leave this field blank. Java agent listener service components do not use an outbound agent.

- **Correlation ID** - Optionally specify the Oracle Advanced Queuing (AQ) correlation ID of the event messages that you want the Java agent listener to process. The AQ correlation ID for an event message in the Business Event System is usually specified in the following format:

```
<event name>
```

Consequently, by specifying a correlation ID in this attribute, you can dedicate the Java agent listener to listen only for messages that are instances of the specified event. You can also specify a partial value to listen for messages that are instances of any event whose name begins with the specified value.

The seeded Java agent listeners do not have any correlation ID specified so that they can process all event messages on their respective agents.

> **Note:** The AQ correlation ID is different than the correlation ID contained within the WF_EVENT_T event message structure.

To cancel the configuration without saving any changes, click the Cancel button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

Details

This page lets you define detail attributes for the service component. You must set attributes marked with an asterisk (*) to appropriate values for your environment before you can run the service component. A refresh icon identifies attributes that can be refreshed dynamically while the service component is running.

- **ID** - When you edit a previously created service component, the configuration wizard displays the identifier for the service component.

- **Status** - When you edit a previously created service component, the configuration wizard displays the status of the service component.

- **Name** - The configuration wizard displays the name defined for the service component.

- **Container** - The container to which the service component will belong. Oracle Workflow provides a container called Workflow Agent Listener Service for Java agent listener service components.

- **Maximum Idle Time** - If you selected the On-Demand startup mode for the service component, enter the maximum time in minutes that the service component can remain idle before it is stopped. An on-demand component that is stopped in this way will be restarted by its container when it is needed again to process new messages.

- **Max Error Count** - The number of consecutive errors the service component can encounter before its container stops it and changes its status to Stopped with Error. If an error is resolved and processing can continue, the error count is reset. The default value for the maximum error count is 10.

- **Inbound Thread Count** - Set the inbound processing thread count to 1 (one) to enable inbound message processing with this Java agent listener. Select 0 (zero) to disable this Java agent listener. The default value is 1.

- **Outbound Thread Count** - Leave this parameter set to the default value of 0 (zero).

Java agent listener service components do not perform outbound message processing.

- **Log Level** - Select the level of detail for the information you want to record in the service component container log. The recommended log level, which is also the default value, is Error. Usually the log level only needs to be changed if you want to record additional detailed information for debugging purposes. You can choose the following levels:

  - 1 - Statement

  - 2 - Procedure

  - 3 - Event

  - 4 - Exception

  - 5 - Error

  - 6 - Unexpected

- **Processor Read Wait Timeout** - Specify the amount of time in seconds that the service component's processing thread continues to wait, after reading the last message from its assigned queue, before timing out. If another message is received before this time expires, that message is processed and the timeout period begins again. If the timeout period expires and no more messages have been received, the service component stops reading and its sleep time begins. The default read timeout period for a Java agent listener is 10 seconds.

- **Processor Min Loop Sleep** - Specify the minimum sleep time in seconds during which the service component waits, after its read timeout period expires, before it checks its queue for messages again. The default minimum sleep time for a Java agent listener is 5 seconds.

- **Processor Max Loop Sleep** - Specify the maximum sleep time in seconds if you want to increase the sleep time between read attempts when no messages are received. If you specify a maximum sleep time that is greater than the minimum sleep time, then the service component initially waits for the minimum sleep time after it finishes reading messages from its queue. If no messages are read in subsequent attempts, then the sleep time between read attempts gradually increases until the maximum sleep time is reached. Increasing the sleep time can help enhance performance if messages are received infrequently. You can also specify 0 (zero) for this parameter to indicate that the sleep time should not be increased. In this case, the service component always waits for the minimum sleep time between read attempts. The default maximum sleep time for a Java agent listener is 60 seconds.

- **Processor Error Loop Sleep** - Specify the sleep time in seconds during which the service component waits, after an error occurs, before it attempts to begin processing again. The default error sleep time for a Java agent listener is 60 seconds.

- **Processor Close on Read Timeout** - Select this parameter to specify that the service component should close its connections after its read timeout period expires, when its sleep time begins. Deselect this parameter to specify that the connections should remain open until the processing thread stops.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

Scheduling Events

This page lets you schedule events to control the running of the service component. The events are raised at the scheduled time by DBMS jobs. For a Java agent listener service component, you can schedule the following events:

- Start

- Refresh

- Suspend

- Resume

- Stop

For each event, the list displays the event name, date and time when the event is first scheduled to be raised, the interval in minutes at which the event is reraised, and, for a refresh event, any parameters to be refreshed. You can specify the following refreshable parameters, using the parameters' internal names, when you refresh the Java agent listener.

- `PROCESSOR_IN_THREAD_COUNT` - Inbound Thread Count

- `COMPONENT_LOG_LEVEL` - Log Level, specified as a numerical value
    - `1` - Statement

    - `2` - Procedure

    - `3` - Event

- 4 - Exception

- 5 - Error

- 6 - Unexpected

To schedule events:

- If no events are currently scheduled, click the Add a Row button to add a new row to the list of events and enter the information for the event.

  - Select the event for the command you want to schedule. Oracle Workflow provides events to let you start, stop, refresh, suspend, or resume the service component.

  - Select the date when you want the event to be raised first.

  - Select the hour and minute to specify the time on the specified date when you want the event to be raised first. The hour values are in a twenty-four hour format. For example, select 00 for midnight, or 23 for 11 PM.

  - If you want to raise the event periodically, enter the time interval in minutes at which you want to raise the event. If you do not specify a repeating interval, the event is raised only once.

  - If you choose the refresh event, you can optionally enter any parameters you want to include with the event in order to refresh the Java agent listener configuration parameters with those values when the event is raised. Specify the parameter names and values in the following format, separating the parameters with a colon (:):
    `internal_parameter_name=parameter_value`

    For example: `PROCESSOR_IN_THREAD_COUNT=1`

- To schedule another event, click the Add Another Row button and enter the information for the event.

- To remove an event, select the event and click the Remove button.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

> **Note:** The configuration wizard verifies that an event is specified for every row in the list when you click the Next or Finish button. If you do not want to schedule another event, you should remove any empty rows before proceeding.

Review

This page lets you review the configuration parameter values that you set and the events that you scheduled for this service component.

• If you want to change any of these settings, return to the appropriate step in the configuration wizard to make your changes. To return to the previous step, click the Back button.

• To save these settings and finish the configuration, click the Finish button.

Web Services Outbound

You can use Web services in Oracle Workflow to initiate outbound Web service requests and to accept inbound Web service requests.

When Web service messages are dequeued by the Oracle E-Business Suite, they are transmitted by the Web service outbound component.

The Web services outbound program is defined as a service component type in the Generic Service Component Framework. This framework helps to simplify and automate the management of background Java services.

Oracle Workflow provides a seeded Web services outbound component named Web Services OUT Agent to process messages on the standard WF_WS_JMS_OUT queue, which is a Business Event System agent. This service component must be started manually. You can optionally update its configuration if necessary.

You can also optionally create additional Web services outbound components. For example, you can configure a Web services outbound component that only processes messages on a particular agent or queue.

If you create custom Web services outbound components, you can either assign them to the seeded container for Web services outbound components, named Workflow Document Web Services Service, or, based on the volume to be handled by the seeded container, you can also choose to create your own custom containers.

Before the seeded Web services outbound component can run, the Workflow Document Web Services Service container which manages it must be first be started. You should ensure that this container is running. If you create your own custom containers for custom service components, ensure that those containers are running as well. Use the Service Instances page to start each container as a service instance in Generic Service Management (GSM).

> **Note:** Inbound Web service messages are processed by a seeded service component of type Java agent listener, named Workflow Web Services In.

Web Services Outbound Configuration Wizard

The Web services outbound configuration wizard lets you configure a Web services outbound service component by defining general and detail attributes and scheduling control events. You can use the configuration wizard to configure a new Web services outbound service component, or to edit the configuration of an existing Web services outbound service component.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > (B) Create > (B) Continue

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Service Components status icon > (B) Edit

Define

This page lets you define general attributes for the service component. Some attributes are already set to required values and cannot be modified. You must set attributes marked with an asterisk (*) to appropriate values for your environment before you can run the service component.

- **ID** - When you edit a previously created service component, the configuration wizard displays the identifier for the service component.

- **Status** - When you edit a previously created service component, the configuration wizard displays the status of the service component.

- **Name** - The name of the service component. This name must be unique.

- **Startup Mode** - Select Automatic, Manual, or On-Demand as the startup mode for the service component.

- **Container Type** - The container type to which this service component belongs, which is always Oracle Applications Generic Service Management (Oracle Applications GSM).

- **Inbound Agent** - Leave this field blank. Web services outbound components do not use an inbound agent.

- **Outbound Agent** - The agent/queue that you want to monitor for outbound Web services messages.

To cancel the configuration without saving any changes, click the Cancel button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

Details

This page lets you define detail attributes for the service component. You must set attributes marked with an asterisk (*) to appropriate values for your environment before you can run the service component. A refresh icon identifies attributes that can be refreshed dynamically while the service component is running.

- **ID** - When you edit a previously created service component, the configuration wizard displays the identifier for the service component.

- **Status** - When you edit a previously created service component, the configuration wizard displays the status of the service component.

- **Name** - The configuration wizard displays the name defined for the service component.

- **Container** - The container to which the service component will belong. Oracle Workflow provides a container called Workflow Document Web Services Service for Web services outbound components.

- **Maximum Idle Time** - If you selected the On-Demand startup mode for the service component, enter the maximum time in minutes that the service component can remain idle before it is stopped. An on-demand component that is stopped in this way will be restarted by its container when it is needed again to process new messages.

- **Max Error Count** - The number of consecutive errors the service component can encounter before its container stops it and changes its status to Stopped with Error. If an error is resolved and processing can continue, the error count is reset. The default value for the maximum error count is 10.

- **Inbound Thread Count** - Leave this parameter set to the default value of 0 (zero). Web services outbound components do not perform inbound message processing.

- **Outbound Thread Count** - Specify the number of outbound processing threads you want to execute simultaneously with this Web services outbound component, depending on the volume of outbound messages you need to send. Specify 0 (zero) to disable this Web services outbound component. The default value is 1 (one).

- **Log Level** - Select the level of detail for the information you want to record in the service component container log. The recommended log level, which is also the default value, is Error. Usually the log level only needs to be changed if you want to record additional detailed information for debugging purposes. You can choose the following levels:

- • 1 - Statement

- • 2 - Procedure

- • 3 - Event

- • 4 - Exception

- • 5 - Error

- • 6 - Unexpected

- **Processor Read Wait Timeout** - Specify the amount of time in seconds that the service component's processing threads continue to wait, after reading the last message from the assigned queue, before timing out. If another message is received before this time expires, that message is processed and the timeout period begins again. If the timeout period expires and no more messages have been received, the service component stops reading and its sleep time begins. The default read timeout period for a Web services outbound component is 10 seconds.

- **Processor Min Loop Sleep** - Specify the minimum sleep time in seconds during which the service component waits, after its read timeout period expires, before it checks its queue for messages again. The default minimum sleep time for a Web services outbound component is 5 seconds.

- **Processor Max Loop Sleep** - Specify the maximum sleep time in seconds if you want to increase the sleep time between read attempts when no messages are received. If you specify a maximum sleep time that is greater than the minimum sleep time, then the service component initially waits for the minimum sleep time after it finishes reading messages from its queue. If no messages are read in subsequent attempts, then the sleep time between read attempts gradually increases until the maximum sleep time is reached. Increasing the sleep time can help enhance performance if messages are received infrequently. You can also specify 0 (zero) for this parameter to indicate that the sleep time should not be increased. In this case, the service component always waits for the minimum sleep time between read attempts. The default maximum sleep time for a Web services outbound component is 60 seconds.

- **Processor Error Loop Sleep** - Specify the sleep time in seconds during which the service component waits, after an error occurs, before it attempts to begin processing again. The default error sleep time for a Web services outbound component is 60 seconds.

- **Processor Close on Read Timeout** - Select this parameter to specify that the service component should close its connections after its read timeout period expires, when its sleep time begins. Deselect this parameter to specify that the connections should

remain open until the processing thread stops.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

Scheduling Events

This page lets you schedule events to control the running of the service component. The events are raised at the scheduled time by DBMS jobs. For a Web services outbound component, you can schedule the following events:

- Start

- Refresh

- Suspend

- Resume

- Stop

For each event, the list displays the event name, date and time when the event is first scheduled to be raised, the interval in minutes at which the event is reraised, and, for a refresh event, any parameters to be refreshed. You can specify the following refreshable parameters, using the parameters' internal names, when you refresh the Web services outbound component.

- `PROCESSOR_OUT_THREAD_COUNT` - Outbound Thread Count

- `COMPONENT_LOG_LEVEL` - Log Level, specified as a numerical value

  - `1` - Statement

  - `2` - Procedure

  - `3` - Event

  - `4` - Exception

  - `5` - Error

  - `6` - Unexpected

To schedule events:

- If no events are currently scheduled, click the Add a Row button to add a new row to the list of events and enter the information for the event.

  - Select the event for the command you want to schedule. Oracle Workflow provides events to let you start, stop, refresh, suspend, or resume the service component.

  - Select the date when you want the event to be raised first.

  - Select the hour and minute to specify the time on the specified date when you want the event to be raised first. The hour values are in a twenty-four hour format. For example, select 00 for midnight, or 23 for 11 PM.

  - If you want to raise the event periodically, enter the time interval in minutes at which you want to raise the event. If you do not specify a repeating interval, the event is raised only once.

  - If you choose the refresh event, you can optionally enter any parameters you want to include with the event in order to refresh the Web services outbound configuration parameters with those values when the event is raised. Specify the parameter names and values in the following format, separating the parameters with a colon (:):
    ```
    internal_parameter_name=parameter_value
    ```

    For example: `PROCESSOR_OUT_THREAD_COUNT=3`

- To schedule another event, click the Add Another Row button and enter the information for the event.

- To remove an event, select the event and click the Remove button.

To cancel any changes on this page, click the Cancel button.

To return to the previous step of the configuration wizard, click the Back button.

To save these settings and proceed to the next step of the configuration wizard, click the Next button.

To save these settings and proceed to the last step of the configuration wizard, click the Finish button.

> **Note:** The configuration wizard verifies that an event is specified for every row in the list when you click the Next or Finish button. If you do not want to schedule another event, you should remove any empty rows before proceeding.

Review

This page lets you review the configuration parameter values that you set and the

events that you scheduled for this service component.

- If you want to change any of these settings, return to the appropriate step in the configuration wizard to make your changes. To return to the previous step, click the Back button.

- To save these settings and finish the configuration, click the Finish button.

Background Engines

Background engine processes serve three purposes in Oracle Workflow: to handle activities deferred by the Workflow Engine, to handle timed out notification activities, and to handle stuck processes.

When the Workflow Engine initiates and performs a process, it completes all necessary activities before continuing to the next eligible activity. In some cases, an activity can require a large amount of processing resource or time to complete. Oracle Workflow lets you manage the load on the Workflow Engine by setting up supplemental engines to run these costly activities as background tasks. In these cases, the costly activity is deferred by the Workflow Engine and run later by a background engine. The main Workflow Engine can then continue to the next available activity, which may occur on some other parallel branch of the process.

A background engine must also be set up to handle timed out notification activities. When the Workflow Engine comes across a notification activity that requires a response, it calls the Notification System to send the notification to the appropriate performer, and then sets the notification activity to a status of 'NOTIFIED' until the performer completes the notification activity. Meanwhile, a background engine set up to handle timed out activities periodically checks for 'NOTIFIED' activities and whether these activities have time out values specified. If a 'NOTIFIED' activity does have a time out value, and the current date and time exceeds that time out value, the background engine marks that activity as timed out and calls the Workflow Engine. The Workflow Engine then resumes by trying to execute a <timeout> transition activity.

Additionally, a background engine must be set up to handle stuck processes. A process is identified as stuck when it has a status of ACTIVE, but cannot progress any further. For example, a process could become stuck in the following situations:

- A thread within a process leads to an activity that is not defined as an End activity but has no other activity modeled after it, and no other activity is active.

- A process with only one thread loops back, but the pivot activity of the loop has the On Revisit property set to Ignore.

- An activity returns a result for which no eligible transition exists. For instance, if the function for a function activity returns an unexpected result value, and no default transition is modeled after that activity, the process cannot continue.

The background engine sets the status of a stuck process to ERROR:#STUCK and executes the error process defined for it.

You can define and start up as many background engines as you like to check for deferred and timed out activities.

Background engines can be restricted to handle activities associated with specific item types, and within specific cost ranges. A background engine runs until it completes all eligible activities at the time it was initiated. Generally, you should set the background engine up to run periodically.

Ensure that you have at least one background engine that can check for timed out activities, one that can process deferred activities, and one that can handle stuck processes. At a minimum, you need to set up one background engine that can handle both timed out and deferred activities as well as stuck processes. Generally, you should run a separate background engine to check for stuck processes at less frequent intervals than the background engine that you run for deferred activities, normally not more often than once a day. Run the background engine to check for stuck processes when the load on the system is low.

Running Background Engines

You run a background engine by submitting the Workflow Background Process concurrent program (FNDWFBG). When you start a new background engine, you can restrict the engine to handle activities associated with specific item types, and within specific cost ranges. You can submit the Workflow Background Process concurrent program several times to schedule different background engines to run at different times.

- To submit a request for the Workflow Background Process concurrent program through Oracle Self-Service Web Applications, choose Background Engines from the Submit Request For pull-down menu in the Workflow System status page and click the Go button.

- To view Workflow Background Process concurrent requests, click the Background Engines status icon in the Workflow System status page.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go

Parameters

When you submit the Workflow Background Process concurrent program, specify the following parameters.

- **Item Type** - Specify an item type to restrict this engine to activities associated with that item type. If you do not specify an item type, the engine processes any activity regardless of its item type.

- **Minimum Threshold** - Specify the minimum cost that an activity must have for this

background engine to execute it, in hundredths of a second.

- **Maximum Threshold** - Specify the maximum cost that an activity can have for this background engine to execute it, in hundredths of a second. By using Minimum Threshold and Maximum Threshold you can create multiple background engines to handle very specific types of activities. The default values for these arguments are null so that the background engine runs activities regardless of cost.

- **Process Deferred** - Specify whether this background engine checks for deferred activities. Setting this parameter to Yes allows the engine to check for deferred activities.

- **Process Timeout** - Specify whether this background engine checks for activities that have timed out. Setting this parameter to Yes allows the engine to check for timed out activities.

- **Process Stuck** - Specify whether this background engine checks for stuck processes. Setting this parameter to Yes allows the engine to check for stuck processes.

    > **Note:** Make sure you have a least one background engine that can check for timed out activities, one that can process deferred activities, and one that can handle stuck processes. At a minimum, you need to set up one background engine that can handle both timed out and deferred activities as well as stuck processes.

Viewing Concurrent Requests

When you view the Workflow Background Process concurrent requests, the Search Results page shows standard request detail information for these requests. For each request, the list displays the request ID, program short name, description, application short name, phase, status, requester, duration, wait time, and submission date. Click a column heading to sort the list by that column.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Background Engines status icon

- To show the details for a request if they are hidden, click the Show link in the Details column. Oracle Applications Manager displays details about the request depending on the status of the request. You can also perform actions, such as placing a hold on a request, canceling a request, viewing diagnostic information, viewing manager details, viewing logs, or viewing request output, by clicking the corresponding button. The actions that are available depend on the status of the request.

- To hide the details for a request if they are shown, click the Hide link in the Details column.

- To search for concurrent requests with different criteria, click the New Search button or click one of the Quick Search links.

- To modify the search criteria from this search, click the Modify Search button.

- To add the information from this page to your support cart, click the Add to Support Cart button.

Purging Workflow Data

The Oracle Applications Manager console helps you easily maintain the Oracle Workflow and Oracle XML Gateway database tables. Oracle Workflow and Oracle XML Gateway access several tables that can grow quite large with obsolete workflow information that is stored for all completed workflow processes, as well as obsolete information for XML transactions. The size of these tables and indexes can adversely affect performance. These tables should be purged on a regular basis, using the Purge Obsolete Workflow Runtime Data concurrent program.

This program purges obsolete runtime information associated with work items, including status information and any associated notifications and Oracle XML Gateway transactions. By default, it also purges obsolete design information, such as activities that are no longer in use and expired ad hoc users and roles, and obsolete runtime information not associated with work items, such as notifications or Oracle XML Gateway transactions that were not handled through a workflow process. You can optionally choose to purge only core runtime information associated with work items for performance gain during periods of high activity, and purge all obsolete information as part of your routine maintenance during periods of low activity.

To preserve electronic signature evidence for future reference, this program by default does not delete any notifications that required signatures or their associated signature information. If you do not need to maintain signature evidence, you can choose to delete signature-related information as well.

> **Note:** You can also use the Purge Obsolete ECX Data concurrent program to purge Oracle XML Gateway transactions according to Oracle XML Gateway-specific parameters. See: Purge Obsolete ECX Data Concurrent Program, *Oracle XML Gateway User's Guide*.

Workflow Purge

The Workflow Purge page shows summary information about the next scheduled and last completed purge requests and about completed work items.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Purge status icon

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Related

Links > Throughput > Work Items

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Requests Summary

This region displays summary information about the next scheduled and last completed Purge Obsolete Workflow Runtime Data concurrent requests.

- To show information in this region if it is hidden, click the Show link.

- To hide information in this region if it is shown, click the Hide link.

Next Scheduled

For the next scheduled Purge Obsolete Workflow Runtime Data concurrent request, Oracle Workflow Manager displays the request ID, requestor, status, requested start time, wait time, and parameters.

Last Completed

For the last completed Purge Obsolete Workflow Runtime Data concurrent request, Oracle Workflow Manager displays the request ID, requestor, status, completed time, duration, and parameters.

To view the log file for the request, click the Request Log link.

Completed Work Items

This region displays the distribution of completed work items across different item types.

- To show information in this region if it is hidden, click the Show link

- To hide information in this region if it is shown, click the Hide link.

- This region displays the date and time when the work item statistics were last updated. To refresh this information, click the refresh icon. See: Gathering Oracle

Workflow Statistics, page 7-2.

For each work item type in the Completed Work Items list, Oracle Workflow Manager displays the work item type name, the persistence type, the retention period in days, the number of completed work items of that type, and the number of items of that type that are available for purging. Click any column heading to sort the list by that column.

- To filter the item types displayed in the list, select an item type property and an operator from the Filter pull-down menus, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  - Work item type display name

  - Work item type internal name

  - Persistence type

  - Retention period

  - Number of completed work items of this type

  - Number of items of this type available for purging

- To view details for work items of a particular item type, either click the item type link in the Work Item Type column, or select the item type and click the View Details button.

Submitting the Purge Program

You perform purging by submitting the Purge Obsolete Workflow Runtime Data concurrent program (FNDWFPR). You can enter restrictions to specify the data that you want to purge.

- To submit a request for the Purge Obsolete Workflow Runtime Data concurrent program through Oracle Self-Service Web Applications, either click the Purge button in the Completed Work Items region of the Workflow Purge page, or choose Purge from the Submit Request For pull-down menu in the Workflow System status page and click the Go button.

- To view Purge Obsolete Workflow Runtime Data concurrent requests, click the View Purge Requests button in the Completed Work Items region of the Workflow Purge page.

Parameters

When you submit the Purge Obsolete Workflow Runtime Data concurrent program, specify the following parameters.

- **Item Type** - Specify the item type to purge. Leave this field blank to purge the runtime data for all item types.

- **Item Key** - Specify the item key to purge. The item key is a unique identifier for an item within an item type. Leave this field blank to purge the runtime data for all items of the specified item type.

- **Age** - Specify the minimum age of data to purge, in days, if you are purging items with a Temporary persistence type. The default is 0 days.

- **Persistence Type** - Specify the persistence type of the data you want to purge, either Permanent or Temporary. The default is Temporary.

- **Core Workflow Only** - Enter 'Y' to purge only obsolete runtime data associated with work items, or 'N' to purge all obsolete runtime data as well obsolete design data. The default is 'N'.

- **Commit Frequency** - Enter the number of records to purge before the program commits data. To reduce rollback size and improve performance, set this parameter to commit data after a smaller number of records. The default is 500 records.

  > **Note:** After performing a commit, the program resumes purging work items with the next subsequent begin date. In some cases, if additional items have the same begin date as the last item that was purged before a commit, the program may not purge all eligible items. To purge these remaining work items, simply rerun the program.

- **Signed Notifications** - Enter 'N' to preserve signature evidence, including notifications that required electronic signatures and their associated signature information. Enter 'Y' to purge signature-related information. The default is 'N'.

Viewing Concurrent Requests

When you view the Purge Obsolete Workflow Runtime Data concurrent requests, the Search Results page shows standard request detail information for these requests. For each request, the list displays the request ID, program short name, description, application short name, phase, status, requestor, duration, wait time, and submission date. Click a column heading to sort the list by that column.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Purge status icon > (B) View Purge Requests

- To show the details for a request if they are hidden, click the Show link in the Details column. Oracle Applications Manager displays details about the request depending on the status of the request. You can also perform actions, such as

placing a hold on a request, canceling a request, viewing diagnostic information, viewing manager details, viewing logs, or viewing request output, by clicking the corresponding button. The actions that are available depend on the status of the request.

- To hide the details for a request if they are shown, click the Hide link in the Details column.

- To search for concurrent requests with different criteria, click the New Search button or click one of the Quick Search links.

- To modify the search criteria from this search, click the Modify Search button.

- To add the information from this page to your support cart, click the Add to Support Cart button.

Completed Work Item Details

This page shows details about completed work items of a particular item type.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Purge status icon > (B) View Details

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Completed Work Items Stage Summary

This region displays the distribution of completed work items that ended at various activity stages within the workflow process. For each activity stage, the list displays the activity internal name and result, and the number of completed work items that ended at that stage. Click any column heading to sort the list by that column.

- By default, the list shows completed work items that ended within the last 30 days. To view completed work items that ended within a different period, enter a number of days in the Filter: End Date Within Last _ Days option and click the Go button.

- To view details about the work items that ended at a particular activity stage, either click the activity stage link in the Work Item Activity Stage column, or select the activity stage and click the View Details button.

Completed Work Item Activity Details

This page shows details about completed work items that ended at a particular activity stage within a particular item type.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Purge status icon > (B) View Details > (B) View Details

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Oracle Workflow Manager displays a list of all completed work items of the selected item type that ended at the selected activity stage. By default, the list shows completed work items that ended within the last 30 days. For each work item, the list displays the internal name of the activity at which the work item ended, the activity start date, end date, user assigned to perform the activity, and item key. Click any column heading to sort the list by that column.

- To filter the work items displayed in the list, select an activity property from the Filter pull-down menu, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  - Internal name of the activity at which the work item ended

  - Start date within a specified number of days

  - End date within a specified number of days

  - User assigned to perform the activity

  - Item key of the work item

- To launch the Workflow Monitor for a work item, select the work item and click the Launch Workflow Monitor button.

> **Note:** If you perform an action in the Workflow Monitor that changes the status of the work item, then you must refresh your Oracle Workflow Manager web page in order to see the updated information.

Workflow Control Queue Cleanup

Oracle Workflow contains a standard Business Event System agent named WF_CONTROL, which is associated with a standard queue that is also named WF_CONTROL. This queue has a payload type of JMS Text message. The WF_CONTROL agent is used for internal processing only, and is not meant for customer use. You should not place custom event messages on this queue.

The Generic Service Component Framework uses WF_CONTROL to handle control events for containers and service components, such as notification mailer or agent listener service components. WF_CONTROL is also used for other Oracle Applications internal processing.

You do not need to schedule propagation for the WF_CONTROL agent, because the middle tier processes that use WF_CONTROL dequeue messages directly from its queue. However, the subscribers to the WF_CONTROL queue need to be cleaned up periodically. A concurrent program named Workflow Control Queue Cleanup is automatically scheduled to perform this cleanup for you.

When a middle tier process for Oracle Applications starts up, it creates a JMS subscriber to the queue. Then, when an event message is placed on the queue, a copy of the event message is created for each subscriber to the queue. If a middle tier process dies, however, the corresponding subscriber remains in the database. For more efficient processing, you should ensure that WF_CONTROL is periodically cleaned up by removing the subscribers for any middle tier processes that are no longer active. The Workflow Control Queue Cleanup concurrent program sends an event named oracle.apps.wf.bes.control.ping to check the status of each subscriber to the WF_CONTROL queue. If the corresponding middle tier process is still alive, it sends back a response. The next time the cleanup program runs, it checks whether responses have been received for each ping event sent during the previous run. If no response was received from a particular subscriber, that subscriber is removed.

The recommended frequency for performing cleanup is every twelve hours. In order to allow enough time for subscribers to respond to the ping event, the minimum wait time between two cleanup runs is thirty minutes. If you run the procedure again less than thirty minutes after the previous run, it will not perform any processing.

Running Workflow Control Queue Cleanup

You perform Workflow control queue cleanup by submitting the Workflow Control Queue Cleanup concurrent program (FNDWFBES_CONTROL_QUEUE_CLEANUP). This program does not require any parameters. This concurrent program is scheduled

to run every twelve hours by default, which is the recommended frequency for performing cleanup. You can optionally submit this program with a different schedule if you want to perform cleanup at a different frequency.

- To submit a request for the Workflow Control Queue Cleanup concurrent program through Oracle Self-Service Web Applications, choose Control Queue Cleanup from the Submit Request For pull-down menu in the Workflow System status page and click the Go button.

- To view Workflow Control Queue Cleanup concurrent requests, click the Control Queue Cleanup status icon in the Workflow System status page.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go

Viewing Concurrent Requests

When you view the Workflow Control Queue Cleanup concurrent requests, the Search Results page shows standard request detail information for these requests. For each request, the list displays the request ID, program short name, description, application short name, phase, status, requester, duration, wait time, and submission date. Click a column heading to sort the list by that column.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Control Queue Cleanup status icon

- To show the details for a request if they are hidden, click the Show link in the Details column. Oracle Applications Manager displays details about the request depending on the status of the request. You can also perform actions, such as placing a hold on a request, canceling a request, viewing diagnostic information, viewing manager details, viewing logs, or viewing request output, by clicking the corresponding button. The actions that are available depend on the status of the request.

- To hide the details for a request if they are shown, click the Hide link in the Details column.

- To search for concurrent requests with different criteria, click the New Search button or click one of the Quick Search links.

- To modify the search criteria from this search, click the Modify Search button.

- To add the information from this page to your support cart, click the Add to Support Cart button.

Active Work Items

The Active Work Items page shows the distribution of active work items across

different item types. All work items that do not have an end date are counted as Active work items, including deferred, suspended, and errored work items as well as running work items.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Active

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

The page displays the date and time when the work item statistics were last updated. To refresh this information, click the refresh icon. See: Gathering Oracle Workflow Statistics, page 7-2.

For each work item type, the Active Work Items page displays the work item type name and the number of active work items of that type. Click any column heading to sort the list by that column.

- To filter the item types displayed in the list, select an item type property and an operator from the Filter pull-down menus, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  - Work item type display name

  - Work item type internal name

  - Number of active work items of this type

To view details about active work item activities within a particular item type, either click the item type link in the Work Item Type column, or select the item type and click the View Details button.

Active Work Item Activities

This page shows details about active work item activities within a particular item type. Active work item activities include only activities with a status of Active, Waiting, or Notified.

Note: Only activities with a status of Active, Waiting, or Notified are included in this page. Activities with a status of Deferred, Suspended, or Error are not included in this page, although the work items to which they belong are counted as Active work items. You can use the View pull-down menu to view details for activities with a status of Deferred, Suspended, or Error.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Active > (B) View Details

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Active Work Items Stage Summary

This region displays the distribution of active work items that are currently at various activity stages within the workflow process, if the activity has a status of Active, Waiting, or Notified. For each activity stage, the list displays the activity internal name and the number of active work items at that stage. Click any column heading to sort the list by that column.

- By default, the list shows active work items that started within the last 30 days. To view active work items that started within a different period, enter a number of days in the Filter: Start Date Within Last _ Days option and click the Go button.

- To view details about the work items at a particular activity stage, either click the activity stage link in the Work Item Activity Stage column, or select the activity stage and click the View Details button.

Active Work Item Activity Details

This page shows details about active work item activities of a particular activity stage within a particular item type. Active work item activities include only activities with a status of Active, Waiting, or Notified.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go >

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Oracle Workflow Manager displays a list of all active activities of the selected stage for work items of the selected item type. Active work item activities include only activities with a status of Active, Waiting, or Notified. By default, the list shows active work items that started within the last 30 days. For each activity, the list displays the activity internal name, start date, due date, user assigned to perform the activity, and item key of the work item. Click any column heading to sort the list by that column.

- To filter the work items displayed in the list, select an activity property from the Filter pull-down menu, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

    - Internal name of the active activity

    - Start date within a specified number of days

    - Due date within a specified number of days

    - User assigned to perform the activity

    - Item key of the work item

- To abort all work items in the list, click the Abort All button. If you have filtered the list, only the work items currently displayed in the list are aborted.

- To suspend all activities in the list, click the Suspend All button. If you have filtered the list, only the work items currently displayed in the list are suspended.

- To abort a single work item, select the activity you want and click the Abort button.

- To suspend a single activity, select the activity you want and click the Suspend button.

- To launch the Workflow Monitor for a work item, select the activity you want and

click the Launch Workflow Monitor button.

> **Note:** If you perform an action in the Workflow Monitor that changes the status of the work item, such as aborting the work item, then you must refresh your Oracle Workflow Manager web page in order to see the updated information.

Deferred Work Items

The Deferred Work Items page shows the distribution of deferred work items across different item types. An abnormal number of activities with a deferred status may indicate that there are not enough background engines available.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Deferred

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

The page displays the date and time when the work item statistics were last updated. To refresh this information, click the refresh icon. See: Gathering Oracle Workflow Statistics, page 7-2.

For each work item type, the Deferred Work Items page displays the work item type name and the number of deferred work items of that type. Click any column heading to sort the list by that column.

- To filter the item types displayed in the list, select an item type property and an operator from the Filter pull-down menus, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  - Work item type display name

  - Work item type internal name

  - Number of deferred work items of this type

- To view details for work items of a particular item type, either click the item type link in the Work Item Type column, or select the item type and click the View Details button.

Deferred Work Item Details

This page shows details about deferred work items of a particular item type.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Deferred > (B) View Details

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Deferred Work Items Stage Summary

This region displays the distribution of deferred work items that are currently at various activity stages within the workflow process. For each activity stage, the list displays the activity internal name and the number of deferred work items at that stage. Click any column heading to sort the list by that column.

- By default, the list shows active work items that started within the last 30 days. To view deferred work items that started within a different period, enter a number of days in the Filter: Start Date Within Last _ Days option and click the Go button.

- To view details about the work items at a particular activity stage, either click the activity stage link in the Work Item Activity Stage column, or select the activity stage and click the View Details button.

Deferred Work Item Activity Details

This page shows details about deferred work items that are currently at a particular activity stage within a particular item type.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Deferred > (B) View Details > (B) View Details

To view work items with a different status, choose the status you want from the View

pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Oracle Workflow Manager displays a list of all deferred activities of the selected stage for work items of the selected item type. By default, the list shows deferred work items that started within the last 30 days. For each activity, the list displays the activity internal name, start date, due date, user assigned to perform the activity, and item key of the work item. Click any column heading to sort the list by that column.

- To filter the work items displayed in the list, select an activity property from the Filter pull-down menu, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  - Internal name of the deferred activity

  - Start date within a specified number of days

  - Due date within a specified number of days

  - User assigned to perform the activity

  - Item key of the work item

- To abort all work items in the list, click the Abort All button. If you have filtered the list, only the work items currently displayed in the list are aborted.

- To suspend all activities in the list, click the Suspend All button. If you have filtered the list, only the work items currently displayed in the list are suspended.

- To abort a single work item, select the activity you want and click the Abort button.

- To suspend a single activity, select the activity you want and click the Suspend button.

- To launch the Workflow Monitor for a work item, select the activity you want and click the Launch Workflow Monitor button.

    Note: If you perform an action in the Workflow Monitor that changes

the status of the work item, such as aborting the work item, then you must refresh your Oracle Workflow Manager web page in order to see the updated information.

Suspended Work Items

The Suspended Work Items page shows the distribution of suspended work items across different item types.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Suspended

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

The page displays the date and time when the work item statistics were last updated. To refresh this information, click the refresh icon. See: Gathering Oracle Workflow Statistics, page 7-2.

For each work item type, the Suspended Work Items page displays the work item type name and the number of suspended work items of that type. Click any column heading to sort the list by that column.

- To filter the item types displayed in the list, select an item type property and an operator from the Filter pull-down menus, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  - Work item type display name

  - Work item type internal name

  - Number of suspended work items of this type

- To view details for an item type, either click the item type link in the Work Item Type column, or select the item type and click the View Details button.

Suspended Work Item Details

This page shows details about all suspended work items of a particular item type.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Suspended > (B) View Details

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Suspended Work Items Stage Summary

This region displays the distribution of suspended work items that are currently at various activity stages within the workflow process. For each activity stage, the list displays the activity internal name and the number of suspended work items at that stage. Click any column heading to sort the list by that column.

- To view suspended work items that started within a specific period, enter a number of days in the Filter: Start Date Within Last _ Days option and click the Go button.

- To view details about the work items at a particular activity stage, either click the activity stage link in the Work Item Activity Stage column, or select the activity stage and click the View Details button.

Suspended Work Item Activity Details

This page shows details about all suspended work items at a particular activity stage within a particular item type.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Suspended > (B) View Details > (B) View Details

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Oracle Workflow Manager displays a list of all suspended activities of the selected stage for work items of the selected item type. For each activity, the list displays the activity internal name, start date, due date, user assigned to perform the activity, and item key of the work item. Click any column heading to sort the list by that column.

- To filter the work items displayed in the list, select an activity property from the Filter pull-down menu, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  - Internal name of the suspended activity

  - Start date within a specified number of days

  - Due date within a specified number of days

  - User assigned to perform the activity

  - Item key of the work item

- To abort all work items in the list, click the Abort All button. If you have filtered the list, only the work items currently displayed in the list are aborted.

- To resume all activities in the list, click the Resume All button. If you have filtered the list, only the work items currently displayed in the list are resumed.

- To abort a single work item, select the activity you want and click the Abort button.

- To resume a single activity, select the activity you want and click the Resume button.

- To launch the Workflow Monitor for a work item, select the activity you want and click the Launch Workflow Monitor button.

    Note: If you perform an action in the Workflow Monitor that changes the status of the work item, such as aborting the work item, then you must refresh your Oracle Workflow Manager web page in order to see the updated information.

Errored Work Items

The Errored Work Items page shows the distribution of errored work items across different item types.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Error

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

• Completed Work Items

• Active Work Items

• Deferred Work Items

• Suspended Work Items

• Errored Work Items

The page displays the date and time when the work item statistics were last updated. To refresh this information, click the refresh icon. See: Gathering Oracle Workflow Statistics, page 7-2.

For each work item type, the Errored Work Items page displays the work item type name and the number of errored work items of that type. Click any column heading to sort the list by that column.

• To filter the item types displayed in the list, select an item type property and an operator from the Filter pull-down menus, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

  • Work item type display name

  • Work item type internal name

  • Number of errored work items of this type

• To view details for an item type, either click the item type link in the Work Item Type column, or select the item type and click the View Details button.

Errored Work Item Details

This page shows details about all errored work items of a particular item type.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Error > (B) View Details

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Errored Work Items Stage Summary

This region displays the distribution of errored work items that are currently at various activity stages within the workflow process. For each activity stage, the list displays the activity internal name and the number of errored work items at that stage. Click any column heading to sort the list by that column.

- To view errored work items that started within a specific period, enter a number of days in the Filter: Start Date Within Last _ Days option and click the Go button.

- To view details about the work items at a particular activity stage, either click the activity stage link in the Work Item Activity Stage column, or select the activity stage and click the View Details button.

Errored Work Item Activity Details

This page shows details about all errored work items at a particular activity stage within a particular item type.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Work Items > Error > (B) View Details > (B) View Details

To view work items with a different status, choose the status you want from the View pull-down menu and click the Go button. You can view items with the following statuses:

- Completed Work Items

- Active Work Items

- Deferred Work Items

- Suspended Work Items

- Errored Work Items

Oracle Workflow Manager displays a list of all errored activities of the selected stage for work items of the selected item type. For each activity, the list displays the activity internal name, start date, due date, user assigned to perform the activity, and item key of the work item. Click any column heading to sort the list by that column.

- To filter the work items displayed in the list, select an activity property from the Filter pull-down menu, enter a filter value in the text field, and click the Go button. You can filter by the following properties:

    - Internal name of the errored activity

    - Start date within a specified number of days

    - Due date within a specified number of days

    - User assigned to perform the activity

    - Item key of the work item

- To abort all work items in the list, click the Abort All button. If you have filtered the list, only the work items currently displayed in the list are aborted.

- To retry all activities in the list, click the Retry All button. If you have filtered the list, only the work items currently displayed in the list are retried.

- To abort a single work item, select the activity you want and click the Abort button.

- To retry a single activity, select the activity you want and click the Retry button.

- To launch the Workflow Monitor for a work item, select the activity you want and click the Launch Workflow Monitor button.

    Note: If you perform an action in the Workflow Monitor that changes the status of the work item, such as aborting the work item, then you must refresh your Oracle Workflow Manager web page in order to see the updated information.

Agents

The Agent Activity page shows the distribution of event messages with different statuses on different Business Event System agents in your instance of Oracle Workflow.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Agent Activity

The page displays the date and time when the agent activity statistics were last

updated. To refresh this information, click the refresh icon. See: Gathering Oracle Workflow Statistics, page 7-2.

For each agent, the list displays the agent name as well as the number of event messages on that agent with the following statuses: Ready, Waiting, Processed, Expired, and Undeliverable. Click any column heading to sort the list by that column.

- To view queue details for an agent, click the agent link in the Agent column.

- To view details about the messages being held on an agent, select the agent and click the Search Agent Entry Details button.

> **Note:** The Agent Activity page displays event messages on the WF_ERROR agent according to their explicitly assigned status on the WF_ERROR queue, unlike the Agent Activity graph in the Workflow System Status page which summarizes all messages on the WF_ERROR agent in an Error status.

If an inbound agent has an abnormally large number of messages with a status of Ready, you may need to check the status of the agent listener processing message for that agent, or create a new agent listener service component for that agent. Similarly, if an outbound agent has an abnormally large number of messages with a status of Ready, you may need to check the status of the propagation schedule for that agent's queue, or schedule propagation if necessary.

Agent Queue Details

The Agent Details page displays the following details for the queue associated with an agent:

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Agent Activity > agent link

- Owner - The owner of the queue.

- Name - The name of the queue.

- Queue Table - The name of the table in which the queue data resides.

- Queue ID - The object number of the queue.

- Queue Type - The type of the queue.

- Maximum Retries - The maximum number of attempts that is allowed when dequeuing a message from the queue.

- Retry Delay - The time interval between retry attempts, when dequeuing a message from the queue.

- Enqueue Enabled - Whether the queue is enabled for enqueuing.

- Dequeue Enabled - Whether the queue is enabled for dequeuing.

- Retention - The time interval during which processed messages are retained in the queue.

- User Comments - Descriptive comments about the queue.

After reviewing the agent queue details, choose the OK button to return to the Agent Activity page.

Message Details

The Search Queue page lets you search for messages being held on a particular agent and review details about those messages. This page displays different message details depending on the payload type of the agent's queue.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Workflow Metrics > Agent Activity > (B) Search Agent Entry Details

WF_EVENT_T and SYS.AQ$_JMS_TEXT_MESSAGE

This page lets you review messages on queues with a payload type of WF_EVENT_T, such as the standard WF_ERROR or WF_DEFERRED queues, or SYS.AQ$_JMS_TEXT_MESSAGE, such as the standard WF_CONTROL queue.

Enter filter criteria to locate the messages you want to review and click the Go button. You can filter by the following message properties:

- Internal event name

- Event key

- Correlation ID used to associate a message with other related messages

- Enqueue date either within the last seven days or prior to the last seven days

- Dequeue date either within the last seven days, prior to the last seven days, or on any date

- Status

Oracle Workflow Manager displays the event messages on the queue for the selected agent that match your filter criteria. For each message, the list displays the event name, event key, correlation ID, event parameters, From System that sent the message, To System that received the message, date the message was sent, error message, error stack, and the message status.

The list also includes any messages on the exception queue associated with the selected

queue. Messages are transferred from a user queue to the associated exception queue if Oracle Advanced Queuing cannot retrieve or process them for some reason. For more information, see: Oracle Streams AQ Exception Handling, *Oracle Streams Advanced Queuing User's Guide and Reference*.

> **Note:** Each queue table contains one default exception queue that is shared by all the user queues in that queue table. When you search for messages on a particular queue, the search result list includes all messages on the associated exception queue as well, regardless of the user queue from which they originated. Consequently, if you create more than one user queue in the same queue table, the search result list may display exception messages that originated from other queues than the queue you selected.

- To review the event data for a message as an XML document, choose the message details icon in the View XML column.

  > **Note:** The message details icon is disabled if the event data for a message is empty.

- To add the information from this page to your support cart, click the Add to Support Cart button.

SYSTEM.ECXMSG

This page lets you review messages on queues with a payload type of SYSTEM.ECXMSG, including the standard Oracle XML Gateway ECX_INBOUND and ECX_OUTBOUND queues.

Enter filter criteria to locate the messages you want to review and click the Go button. You can filter by the following message properties:

- Transaction type

- Document number

- Party site ID

- Correlation ID used to associate a message with other related messages

- Enqueue date either within the last seven days or prior to the last seven days

- Dequeue date either within the last seven days, prior to the last seven days, or on any date

- Status

Oracle Workflow Manager displays the messages on the queue for the selected agent that match your filter criteria. For each message, the list displays the message type, message standard, transaction type and subtype, document number, party ID, party site ID, party type, protocol type, protocol address, first, second, third, fourth, and fifth attributes, and the message status.

- To review the XML document for a message, choose the message details icon in the View XML column.

> **Note:** The message details icon is disabled if the XML document for a message is empty.

- To add the information from this page to your support cart, click the Add to Support Cart button.

SYSTEM.ECX_INENGOBJ

This page lets you review messages on queues with a payload type of SYSTEM.ECX_INENGOBJ, including the standard Oracle XML Gateway ECX_IN_OAG_Q queue.

Enter filter criteria to locate the messages you want to review and click the Go button. You can filter by the following message properties:

- Message ID

- Correlation ID used to associate a message with other related messages

- Enqueue date either within the last seven days or prior to the last seven days

- Dequeue date either within the last seven days, prior to the last seven days, or on any date

- Status

Oracle Workflow Manager displays the messages on the queue for the selected agent that match your filter criteria. For each message, the list displays the message ID, debug mode, and the message status.

To add the information from this page to your support cart, click the Add to Support Cart button.

Queue Propagation

You should schedule propagation for your local outbound agents to send event messages to their destinations. You can schedule Oracle Advanced Queueing (AQ) propagation for agents that use the SQLNET protocol by the following methods:

- Use the Distributed Database Management feature to manage AQ through Oracle Enterprise Manager. See: Oracle Enterprise Manager Support, *Oracle Streams Advanced Queuing User's Guide and Reference*.

- Run the DBMS_AQADM.Schedule_Propagation API in SQL*Plus. See: DBMS_AQADM, *Oracle Database PL/SQL Packages and Types Reference*.

If you want to use the standard WF_OUT and WF_JMS_OUT agents or custom agents for event message propagation, ensure that you schedule propagation for those agents. You do not need to schedule propagation for the WF_CONTROL or WF_NOTIFICATION_OUT agents, however, because the middle tier processes that use WF_CONTROL dequeue messages directly from its queue, and a notification mailer sends messages placed on the WF_NOTIFICATION_OUT queue.

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Related Links > Configuration > Queue Propagation

Queue Propagation

Use the Queue Propagation page to review the database initialization parameters required for queue propagation, as well as the existing propagation schedules for Business Event System agents in your instance of Oracle Workflow.

Database Initialization Parameters for Queue Propagation

For each parameter, this list shows the parameter name, actual parameter value, recommended value, and description. If the actual value does not match the recommended value, the recommended value is marked with a warning indicator icon.

The JOB_QUEUE_PROCESSES parameter defines the number of SNP job queue processes for your instance. Oracle Workflow requires job queue processes to handle propagation of Business Event System event messages by AQ queues. The recommended number of processes for Oracle Workflow is ten or more.

> **Note:** In Oracle Database 10*g*, you do not need to set the AQ_TM_PROCESSES parameter.

Queue Schedules

For each propagation schedule, the list displays the outbound queue, destination database link, job queue process executing the schedule, whether the schedule is enabled or disabled, and the error date and error message of the last unsuccessful execution. Click any column heading to sort the list by that column.

If no process is allocated to execute the schedule, you may need to increase the JOB_QUEUE_PROCESSES database initialization parameter to ensure that processes are available for propagation.

To view details for a propagation schedule, either click the queue link in the Queue column, or select the schedule and click the View Details button.

Queue Propagation Details

The Queue Propagation Details page displays the following details for a propagation schedule:

Navigation: Applications Dashboard > (pull-down menu) Workflow Manager > (B) Go > Related Links > Configuration > Queue Propagation > (B) View Details

- Destination - The destination database link.

- Process Name - The name of the job queue process executing this schedule.

- Enabled - Y if this schedule is enabled or N if the schedule is disabled. The schedule will not be executed if it is disabled.

- Last Error Date - The date of the last unsuccessful execution.

- Last Error Time - The time of the last unsuccessful execution.

- Last Error Message - The error message of the last unsuccessful execution.

- Schema - The schema that owns the queue.

- Session ID - The session ID (SID, SERIAL#) of the job executing this schedule; NULL if not currently executing.

- Propagation Window - The duration in seconds of the propagation window.

- Maximum Bytes - The maximum number of bytes propagated during a propagation window.

- Failures - The number of times that execution of the schedule failed. If the number of failures reaches 16, the schedule will be disabled.

- Latency - The latency time in seconds that specifies how long to wait, after all messages have been propagated, before rechecking the queue for new messages to the destination. The latency represents the maximum wait time during the propagation window for a message to be propagated after it is enqueued.

- Next Run Date - The date at which the next propagation window of this schedule will be started.

- Next Run Time - The time at which the next propagation window of this schedule will be started, in HH:MI:SS format.

- Current Start Date - The date at which the current propagation window of this schedule was started.

- Current Start Time - The time at which the current propagation window of this schedule was started, in HH:MI:SS format.

- Instance - The cluster database instance number executing the schedule.

- Start Date - The date when propagation should be started, in the default date format.

- Start Time - The time when propagation should be started, in HH:MI:SS format.

- Last Run Date - The date of the last successful execution.

- Last Run Time - The time of the last successful execution, in HH:MI:SS format.

- Total Time - The total time, in seconds, spent by the system in executing this schedule.

- Total Number - The total number of messages propagated in this schedule.

- Total Bytes - The total number of bytes propagated in this schedule .

- Maximum Number - The maximum number of messages propagated during a propagation window.

- Average Number - The average number of messages propagated during a propagation window.

- Average Size - The average size of a propagated message, in bytes.

- Average Time - The average time, in seconds, to propagate a message.

8

Oracle Workflow Loaders

This chapter describes access protection for workflow object definitions and how to load those definitions between a database and a flat file.

This chapter covers the following topics:

- Overview of Oracle Workflow Access Protection
- Access Protection for Business Event System Data
- Using the Workflow Definitions Loader
- Using the Workflow XML Loader

Overview of Oracle Workflow Access Protection

Access protection is a feature that prevents workflow seed data created by a 'seed data provider' from being modified by a 'seed data consumer'. Here, a 'seed data provider' is any organization that creates 'seed data' for other organizations ('seed data consumers') to use in defining and customizing a workflow process. In Oracle Workflow, seed data refers to either of the following:

- Workflow object definitions that can and should be customized to meet a certain consumer's needs.

- Workflow object definitions protected against customization because they represent standards that may also be upgraded in the future by the provider.

For example, the Oracle Workflow development team is a provider of seed data called the Standard item type. The Standard item type contains standard activities that can be dropped into any custom workflow process. The development team at your organization's headquarters may create a custom workflow process definition that references activities from the Standard item type. This makes the headquarters team a consumer of the Standard item type seed data.

Now suppose the headquarters team wants to deploy the custom workflow definition that it created to teams at other regional offices. The headquarters team, as seed data

providers, may want to do the following:

- Identify certain workflow objects in its custom workflow definition as corporate standards that the regional teams should adhere to and not modify.

- Designate certain objects in its deployed process as customizable for the regional offices to alter to their offices' needs.

The headquarters team can satisfy both requirement using the access protection feature in Oracle Workflow. Access protection lets seed data providers protect certain data as 'read-only', while allowing other data to be customized. Also during a seed data upgrade, access protection lets the seed data provider overwrite any existing protected seed data with new versions of that seed data, while preserving any customizations made to customizable seed data.

Oracle Workflow assigns a protection and customization level to every workflow object definition stored in the database and requires every user of Oracle Workflow to operate at a certain access level. The combination of protection, customization, and access levels makes up the access protection feature and determines whether a user can modify a given workflow object. The level, in all three cases, is a numeric value ranging from 0 to 1000 that indicates the relationship between different organizations as providers and consumers of seed data.

The following range of levels are presumed by Oracle Workflow:

0-9	Oracle Workflow
10-19	Oracle Application Object Library
20-99	Oracle Applications development
100-999	Customer organization. You can determine how you want this range to be interpreted. For example, 100 can represent headquarters, while 101 can represent a regional office, and so on.
1000	Public

Access Level

Each user of Oracle Workflow operates the system at a certain access level according to the range of levels listed above. A "user of Oracle Workflow" in this case, represents someone who is operating Oracle Workflow Builder, or the Workflow Definitions Loader program, which loads workflow process definitions from a file into a database. As a seed data provider, you should always operate Oracle Workflow Builder at the same consistent access level because the level you work at affects the protection level of the seed data you create.

You can view your access level as follows:

- In Oracle Workflow Builder, select About Workflow from the Help menu.

- If you are going to run the Workflow Definitions Loader program to download workflow process definitions from the database to a file, check the value for the environment variable WF_ACCESS_LEVEL on your workflow server. See: Using the Workflow Definitions Loader, page 8-8.

> **Note:** The Workflow Definitions Loader program references the access level stored in the environment variable called WF_ACCESS_LEVEL, which you must define when you install Oracle Workflow on your server. If you do not define this environment variable, the Workflow Definitions Loader simply assumes a default access level of 1.
>
> When you install Oracle Applications, you need to define this variable in an environment file. The default environment file is APPLSYS.env. If you do not define this environment variable, the Workflow Definitions Loader simply assumes a default access level of 1. Refer to your Oracle Applications installation manual for more information about environment files.

Protection Level

Whenever you create a workflow object in Oracle Workflow Builder, you have the option of protecting the object at a certain level. An object's protection level helps control whether other users can modify the object based on their access levels, by allowing only users with an access level equal to or lower than the object's protection level to modify the object.

> **Note:** The range of access levels allowed to modify the object may be further restricted by the object's customization level.

To set the protection level of an object, display the Access tab of the object's property page and either check or clear the Lock at this Access Level check box. The protection level that you set for an object is dependent on the setting of the Lock at this Access Level check box and on your current access level.

- If you check the Lock at this Access Level check box, the protection level for the object is set to your current access level. Users with an access level higher than your current access level will not be able to modify the object. These users will see a small lock on the workflow object's icon, indicating that the object can be used but not modified. For users with an access level equal to or lower than your current access level, the customization level for the object will determine whether they can modify the object.

- If you do not check the Lock at this Access Level check box, the protection level for the object is set to 1000. In this case all users who are not restricted by the customization level can modify the object.

Customization Level

Every workflow object, in addition to having a protection level, also records a customization level when you modify the object and save it to a database or file. An object's customization level helps control whether other users can modify the object based on their access levels, by allowing only users with an access level equal to or higher than the object's customization level to modify the object.

> **Note:** The range of access levels allowed to modify the object may be further restricted by the object's protection level.

Setting the customization level ensures that a customizable object that has been customized never gets overwritten during a seed data upgrade, because the upgrade always occurs with the Workflow Definitions Loader operating at an access level below the customized object's customization level.

To set the customization level of an object, display the Access tab of the object's property page and either check or clear the Preserve Customizations check box. The customization level that you set for an object is dependent on the setting of the Preserve Customizations check box and on your current access level.

- If you check the Preserve Customizations check box, the customization level for the object is set to your current access level. Users with an access level lower than your current access level will not be able to modify the object. These users will see a small lock on the workflow object's icon, indicating that the object can be used but not modified. For users with an access level equal to or lower than your current access level, the protection level for the object will determine whether they can modify the object.

- If you do not check the Preserve Customizations check box, the customization level for the object is set to 0. In this case all users who are not restricted by the protection level can modify the object.

Combining Protection and Customization Levels to Control Access

You control access to an object by the combined settings of the protection level and the customization level. You can set the Preserve Customizations and Lock at this Access Level check boxes for an object in one of four ways to specify the type of access you want to allow:

- **Allow access to everyone** - By default, all users are allowed access to an object if the Preserve Customizations and the Lock at this Access Level check box are both not

checked. That is, the protection level is 1000 and the customization level is 0.

- **Limit access to users with access levels equal to your own or higher** - If you check the Preserve Customizations check box but do not check the Lock at this Access Level check box, you designate the object as being customizable by anyone with an access level equal to or higher than your current access level. However, users with a lower access level will not be able to modify the object. That is, the protection level is 1000 and the customization level is your current access level. You should only mark objects as customizable in this way if you are sure that you will not be providing upgraded versions of this object in the future that would overwrite other users' customizations to it.

- **Limit access to users with access levels equal to your own or lower** - If you check the Lock at this Access Level check box but do not check the Preserve Customizations check box, you protect the object and ensure that the object can only be modified by users with an access level equal to or lower than your current access level. Users with a higher access level will not be able to modify the object. That is, the protection level is your current access level and the customization level is 0. Protect any objects that you want to define as standard components that will not change unless you provide a global upgrade. For this reason, it is important that you always operate at the same consistent access level.

- **Limit access to users with access levels equal to your own** - If you check both the Lock at this Level and Preserve Customizations check boxes, you ensure that the object cannot be modified by anyone other than users operating at your current access level. That is, the protection level and customization level are both set to your current access level.

The following table summarizes which access levels can access an object under different settings of the Preserve Customizations and Lock at this Access Level options.

Object Access Under Combined Customization and Protection Levels

Preserve Customizations	Lock at this Access Level	Access Level Applied to Object
Cleared	Cleared	Object may be updated by any access level.
Checked	Cleared	Object may only be updated by users with access levels equal to or higher than your current access level.

Preserve Customizations	Lock at this Access Level	Access Level Applied to Object
Cleared	Checked	Object may only be updated by users with access levels equal to or lower than your current access level.
Checked	Checked	Object cannot be updated by any access level except for your current access level.

The protection and access levels in Oracle Workflow are present to remind you that certain workflow objects should not be modified or should only be modified by someone accessing the tool at an authorized access level. This feature is not intended as a means of securing or source controlling your workflow objects.

> **Important:** Most workflow objects provided by Oracle Workflow have a protection level of 0, which means the objects can only be modified by the Oracle Workflow team, operating at an access level of 0. If you attempt to alter your access level to 0 and modify the data anyway, your customizations will not be supported, especially if Oracle Workflow provides an upgrade to the seed data that may overwrite the modifications you make to the originally protected data.

See: To Set the Access Level for an Object, *Oracle Workflow Developer's Guide*.

Setting Up a Default Access Level

When you install Oracle Workflow Builder on a Windows PC, Oracle Universal Installer assigns a default access level that is global to the PC and the operating system you are installing on. After installing Oracle Workflow Builder, you can have individual users on the PC change their access level to a new setting which overrides the default access level set for the PC. If a user does not define an access level, Oracle Workflow Builder assumes the value of the default access level for the PC. The access levels are stored in the Microsoft Windows registry.

If you are deploying Oracle Workflow Builder and workflow seed data to users in other parts of your organization, and you wish to discourage those users from modifying the seed data that you provide, you can have them operate Oracle Workflow Builder at an access level that is higher than the data's protection level. For example if you, as a seed data provider, are operating at an access level of 100 and the seed data you create is protected at a level of 100, then you should require the access level for your users or seed data consumers to be 101 or higher.

You can set a user's access level in Oracle Workflow Builder by having them choose `About Oracle Workflow Builder...` from the Help menu. In the About Oracle Workflow Builder window, change the Access Level field to a number higher than your seed data protection level, then choose OK.

You can also set the access level directly in the Microsoft Windows registry by using a registry editor such as `regedit` to edit the decimal value under `HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\Workflow\Level`.

For the Workflow Definitions Loader program, you set the default access level that the program operates at for downloading process definitions to a file, by defining an environment variable called WF_ACCESS_LEVEL and setting its value using the appropriate operating system command.

> **Caution:** Although you can modify your access level, Oracle Workflow does not support any customizations to seed data originally protected at a level 99 or lower. We STRONGLY RECOMMEND that you not change your access level to an unauthorized level for modifying protected data.

Access Protection for Business Event System Data

Oracle Workflow also provides access protection for business event and event subscription definitions, in order to protect Oracle seed data and preserve your customizations in an upgrade. In the Business Event System, access protection is based on a customization level assigned to each event and subscription. The customization level determines who can update the event or subscription definition.

An event or subscription can have one of the following customization levels:

- Core - This level is used only for events and subscriptions seeded by Oracle. At this level, you cannot make any changes to the event or subscription definition. However, any property in the definition can be changed by an Oracle seed data upgrade.

- Limit - This level is used only for events and subscriptions seeded by Oracle. At this level, you can update the event or subscription status to Enabled or Disabled, but you cannot make any other changes to the object definition. Conversely, an Oracle seed data upgrade cannot change the status setting, but can change any other property in the definition.

- User - This level is automatically set for events and subscriptions that you define. At this level, you can change any property in the event or subscription definition. However, an Oracle seed data upgrade cannot make any changes in the definition.

During an Oracle Applications seed data upgrade, the Workflow XML Loader loads Business Event System object definitions in normal upload mode, preserving any

customized data according to these customization levels. See: Using the Workflow XML Loader, page 8-11.

Related Topics

Events, *Oracle Workflow Developer's Guide*

Event Subscriptions, *Oracle Workflow Developer's Guide*

Using the Workflow Definitions Loader

Rather than use the File Save or File Open menu options in Oracle Workflow Builder, you can also run a program called Workflow Definitions Loader to save or load process definitions from a database or flat file.

Before you upgrade your database, you can use the Workflow Definitions Loader to preserve and back up your process definitions to a flat file. After the database upgrade is complete, you can use the Loader program again to upload the definitions back into your database if you need to reload your saved definitions for any reason. You can also use the Loader program to upgrade your database with a newer version of a process definition or to transfer process definitions to other databases.

When you upload or upgrade a process definition, the Workflow Definitions Loader automatically validates the process definition to ensure that it conforms to specific process design rules. It performs the same validation as the Oracle Workflow Builder Verify feature. See: To Validate a Process Definition, *Oracle Workflow Developer's Guide*.

You run the Workflow Definitions Loader as a concurrent program.

> **Note:** When you upload or upgrade a workflow definition onto an existing definition in a database, it is possible that an object in the upload/upgrade definition has a Display Name that is already in use by a different object in the target database. If this occurs, the Workflow Definitions Loader automatically resolves the display name conflict by adding a '@' character to the beginning of conflicting display names in the target database. The upload/upgrade definition is then applied as is and a warning message is generated.

To run the Workflow Definitions Loader:

1. Navigate to the Submit Requests form in Oracle Applications to submit the Workflow Definitions Loader concurrent program. When you install and set up Oracle Applications and Oracle Workflow, your system administrator needs to add this concurrent program to a request security group for the responsibility from which you want to run this program. See: Overview of Concurrent Programs and Requests, *Oracle Applications System Administrator's Guide*.

2. Submit the Workflow Definitions Loader concurrent program as a request. See: Running Reports and Programs, *Oracle Applications User's Guide*.

3. In the Parameters window, enter values for the following parameters:

Mode

- Specify "Download" to download a process definition from the database to a flat file.

- Specify "Upgrade" to apply a seed data upgrade to a database from an input file. The Workflow Definitions Loader assumes the access level of the file's creator (seed data provider) and overwrites any objects protected at a level equal to or above the upgrade file's access level. The Loader program preserves any customizations made to customizable seed data in the database.

- Specify "Upload" to load a process definition from a flat file into the database. The upload mode is useful to someone who is developing a workflow process. It allows the developer to save definitions to the database without concern that accidental customizations to existing objects might prevent the upload of some process definition elements. The Workflow Definitions Loader uses the access level defined by the input file to upload the process definitions from the file and therefore will overwrite objects in the database that are protected at a level equal to or higher than that file's access level.

- Specify "Force" to force an upload of the process definitions from an input file to a database regardless of an object's protection level You should be certain that the process definition in the file is correct as it overwrites the entire process stored in the database. The Force mode is useful for fixing data integrity problems in a database with a known, reliable file backup.

File

Specify the full path and name of the file that you want to download a process definition to, or upgrade or upload a process definition from.

Item Type

If you set Mode to "Download", use the List button to choose the item type for the process definition you

want to download.

> **Note:** When you submit the Workflow
> Definitions Loader from the Submit
> Requests form to download process
> definitions to a file, you can only
> specify to download one item type at a
> time. If you wish to download multiple
> or all item types simultaneously, you
> should submit the Workflow
> Definitions Loader concurrent
> program from the command line. See
> Step 6, page 8-10 below for details.

4. Choose OK to close the Parameters window.

5. When you finish modifying the print and run options for this request, choose
   Submit to submit the request.

6. Rather than use the Submit Requests form, you can also run the Workflow
   Definitions Loader concurrent program from the command line by entering the
   following commands:

   • To upgrade:

     ```
     WFLOAD apps/pwd 0 Y UPGRADE file.wft
     ```

   • To upload:

     ```
     WFLOAD apps/pwd 0 Y UPLOAD file.wft
     ```

   • To force:

     ```
     WFLOAD apps/pwd 0 Y FORCE file.wft
     ```

   • To download:

     ```
     WFLOAD apps/pwd 0 Y DOWNLOADfile.wftITEMTYPE1 [ITEMTYPE2
     ...ITEMTYPEN]
     ```

   Replace *apps/pwd* with username and password to the APPS schema, replace
   file.wft with the file specification of a workflow process definition file, and
   replace *ITEMTYPE1, ITEMTYPE2, ... ITEMTYPEN* with the one or more item
   type(s) you want to download. You can also download all item types
   simultaneously by replacing *ITEMTYPE1* with '*' (make sure you enclose the
   asterisk in single quotes).

   A file specification is specified as:

   ```
   @<application_short_name>:[<dir>/.../]file.ext
   ```

or

```
<native path>
```

Related Topics

Overview of Oracle Workflow Access Protection, page 8-1

Using the Workflow XML Loader

The Workflow XML Loader is a command line utility that lets you upload and download XML definitions for Business Event System objects between a database and a flat file. When you download Business Event System object definitions from a database, Oracle Workflow saves the definitions as an XML file. When you upload object definitions to a database, Oracle Workflow loads the definitions from the source XML file into the Business Event System tables in the database, creating new definitions or updating existing definitions as necessary.

The XML definitions for Business Event System objects are structured according to the following document type definitions (DTDs):

- Events - WF_EVENTS DTD, *Oracle Workflow API Reference*

- Event group members - WF_EVENT_GROUPS DTD, *Oracle Workflow API Reference*

- Event subscriptions - WF_EVENT_SUBSCRIPTIONS DTD, *Oracle Workflow API Reference*

- Systems - WF_SYSTEMS DTD, *Oracle Workflow API Reference*

- Agents - WF_AGENTS DTD, *Oracle Workflow API Reference*

- Agent group members - WF_AGENT_GROUPS DTD, *Oracle Workflow API Reference*

XML files uploaded or downloaded by the Workflow XML Loader should have the extension .wfx to identify them as Workflow object XML definitions.

You can download Business Event System object definitions in either normal download mode or exact download mode.

- Normal download mode lets you save a generic copy of object definitions from one system that you can use to create similar definitions in other systems. In this mode, the Workflow XML Loader replaces certain system-specific data within the object definitions with tokens. Choose normal download mode, for example, when you want to save Business Event System object definitions from a development system as seed data that can be uploaded to a production system.

- Exact download mode lets you save object definitions as they are specified in the database. In this mode, the Workflow XML Loader does not convert most

system-specific data to tokens; instead, all values are copied to the XML file. Choose exact download mode, for example, when you want to save Business Event System object definitions from one production system so that you can replicate them to another production system that communicates with the first.

In normal download mode, the Workflow XML Loader uses the following tokens to replace system-specific data within Business Event System object definitions. The tokens are prefixed by #.

- `#NEW` - Replaces the global unique identifier for an agent within an agent definition, or for an event subscription within a subscription definition.

- `#LOCAL` - Replaces the global unique identifier for the local system wherever it appears within an agent or subscription definition.

- `#OWNER` - Replaces the name of the schema that owns a queue when the schema appears as part of the queue name and agent address within an agent definition.

- `#SID` - Replaces the database system identifier (SID) when it appears as part of the agent address within an agent definition.

- `#WF_IN` - Replaces the global unique identifier for the WF_IN agent on the local system when it appears as the Source Agent, Out Agent, or To Agent within an event subscription definition.

- `#WF_OUT` - Replaces the global unique identifier for the WF_OUT agent on the local system when it appears as the Source Agent, Out Agent, or To Agent within an event subscription definition.

- `#WF_ERROR` - Replaces the global unique identifier for the WF_ERROR agent on the local system when it appears as the Source Agent, Out Agent, or To Agent within an event subscription definition.

By converting these system-specific values to tokens, the loader produces template definitions that you can use to create similar objects in other systems. When you upload object definitions that contain tokens to a database, Oracle Workflow replaces the tokens with the appropriate values for that system.

> **Note:** Because the Workflow XML Loader replaces the global unique identifier for a subscription with a token in normal download mode, each subscription must be identified instead by a unique combination of subscribing system, source type, triggering event, source agent, owner name, and owner tag.

However, to ensure the portability of the downloaded Business Event System object definitions, the Workflow XML Loader always identifies events by their event names, regardless of the download mode. It never copies the global unique identifier for an

event. Instead, the Workflow XML Loader uses the token `#NEW` to replace the global unique identifier for an event within an event definition. Additionally, the Workflow XML Loader uses event names to represent event groups and event group members within event group member definitions, and to represent triggering events within event subscription definitions. When an event is uploaded to another database, a new global unique ID is generated for the event in that database, while the event name within the event definition is used to match the event with any related objects.

You can upload Business Event System object definitions in either normal upload mode or force upload mode.

- In normal upload mode, the Workflow XML Loader loads the object definitions from the source XML file into the Business Event System tables in the database, but does not update any event or subscription definition with a customization level of User. Oracle Workflow uses this mode to preserve your customizations during upgrades.

- In force upload mode, the Workflow XML Loader loads the object definitions from the source XML file into the Business Event System tables in the database and overwrites any existing definitions, even for events or subscriptions with a customization level of User. Use this mode when you want to update the definitions for your own custom events and subscriptions.

See: Access Protection for Business Event System Data, page 8-7 and Managing Business Events, *Oracle Workflow Developer's Guide*.

To Download Business Event System XML Definitions from a Database:

To download Business Event System object definitions from a database to a flat XML file, run the Workflow XML Loader by running Java against `oracle.apps.fnd.wf.WFXLoad`. You must specify your CLASSPATH pointing to the Java Runtime Environment, the directory containing the Workflow JAR files, the Oracle JDBC implementation, and the following Workflow JAR files:

- `wfjava.jar` - Workflow Java utilities

- `wfapi.jar` - Workflow Java APIs

- `fndctx.jar` - Application Object Library services

> **Note:** The Workflow JAR files are normally located in the `$JAVA_TOP/oracle/apps/fnd/jar` directory.

For example, on UNIX, use the following command to run the Workflow XML Loader:

```
java -classpath "$<JREPATH>/rt.jar:$<Workflow_JAR_file_directory>:
$<Workflow_JAR_file_directory>/wfjava.jar:
$<Workflow_JAR_file_directory>/wfapi.jar:
$<Workflow_JAR_file_directory>/fndctx.jar:
$<ORACLE_HOME>/jdbc/lib/classes111.zip:"
oracle.apps.fnd.wf.WFXLoad -d[e] <user> <password>
<connect_string> <protocol> <lang> <output_file> <object> <key>
OWNER_TAG <owner_tag>
```

On Windows, use the following command:

```
java -classpath ";<JREPATH>\rt.jar;<Workflow_JAR_file_directory>;
<Workflow_JAR_file_directory>\wfjava.jar;
<Workflow_JAR_file_directory>\wfapi.jar;
<Workflow_JAR_file_directory>\fndctx.jar;
<ORACLE_HOME>\jdbc\lib\classes111.zip;"
oracle.apps.fnd.wf.WFXLoad -d[e] <user> <password>
<connect_string> <protocol> <lang> <output_file> <object> <key>
OWNER_TAG <owner_tag>
```

When running the Workflow XML Loader, use either the -d option or the -de option to specify the download mode that you want.

- -d - Normal download mode. The loader converts system-specific data within the object definitions to tokens prefixed with #, where appropriate.

- -de - Exact download mode. The loader copies the object definitions as they are specified in the database and does not convert the system-specific data to tokens.

Additionally, replace the variables in the download command with your parameters as follows:

- *<user>* - The user name of your database account.

- *<password>* - The password for your database account.

- *<connect_string>* - The connect string for the database. The format of the connect string depends on the JDBC driver type.

  - For a JDBC OCI8 driver, the connect string should be the database name as specified in its TNSNAMES entry, in the following format:

    <database_name>

  - For a JDBC THIN driver, you can use two different types of connect string. For the first type, the connect string should include the host name, port number, and database system identifier (SID) in the following format:

    <host_name>:*<port_number>*:*<database_SID>*

    For the second type, the connect string should include an Oracle Net name-value pair with the host name, protocol, port number, and SID in the following format:

    ```
    (description=(address=(host=<host_name>)(protocol= <protocol>)
    (port=<port_number>))(connect_data=(sid=<database_SID>)))
    ```

- *<protocol>* - The JDBC driver type you want to use to connect to the database. The JDBC driver type can be either `oci8` or `thin`.

- *<lang>* - The abbreviation for the language of the XML file. This parameter is case insensitive. Use the standard language abbreviations for the Oracle Database, such as `US` for American or `JA` for Japanese. For a list of the standard language abbreviations, see: Locale Data, *Oracle National Language Support Guide.*

- *<output_file>* - The name and full path of the output file to which you want to save the definitions. The output file should have the extension `.wfx`.

- *<object>* - The type of object definitions you want to download.

  - `EVENT_GROUPS` - Event group definitions

    > **Note:** This object type includes only the definitions of the event groups themselves, without any information about their member events.

  - `EVENTS` - Event definitions for individual events and definitions for those events' memberships in any event groups

  - `SUBSCRIPTIONS` - Event subscription definitions

  - `AGENTS` - Agent, agent group member, and system definitions

    > **Note:** The Workflow XML Loader only downloads event subscription, agent, agent group member, and system definitions that belong to the local system.

- *<key>* - An optional key to restrict the definitions that are downloaded. If you specify a key, the loader retrieves definitions only for those objects whose internal names include that key. The key value is case sensitive and cannot contain any spaces. To retrieve all object definitions of the specified type, you can omit this parameter.

- `OWNER_TAG` *<owner_tag>* - An optional program ID code for the program or application that owns the object definitions you want to download. This value is case sensitive and cannot contain any spaces. If you pass this parameter, you must include the `OWNER_TAG` marker before the owner tag value to identify the parameter.

  > **Note:** To report more extensive debugging information in the program output, you can run the Workflow XML Loader in debug mode by

including the `DebugMode` argument just before the -d or -de option. For example, use the following command:

```
java -classpath <classpath> oracle.apps.fnd.wf.WFXLoad
DebugMode -d[e] <user> <password> <connect_string>
<protocol> <lang> <output_file> <object> <key>
OWNER_TAG <owner_tag>
```

To Upload Business Event System XML Definitions to a Database:

To upload Business Event System object definitions from an XML file to a database, run the Workflow XML Loader by running Java against `oracle.apps.fnd.wf.WFXLoad`. You must specify your CLASSPATH pointing to the Java Runtime Environment, the directory containing the Workflow JAR files, the Oracle JDBC implementation, and the following Workflow JAR files:

*   `wfjava.jar` - Workflow Java utilities

*   `wfapi.jar` - Workflow Java APIs

*   `fndctx.jar` - Application Object Library services

    Note: The Workflow JAR files are normally located in the `$JAVA_TOP/oracle/apps/fnd/jar` directory.

For example, on UNIX, use the following command to run the Workflow XML Loader:

```
java -classpath "$<JREPATH>/rt.jar:$<Workflow_JAR_file_directory>:
$<Workflow_JAR_file_directory>/wfjava.jar:
$<Workflow_JAR_file_directory>/wfapi.jar:
$<Workflow_JAR_file_directory>/fndctx.jar:
$<ORACLE_HOME>/jdbc/lib/classes111.zip:"
oracle.apps.fnd.wf.WFXLoad -u[f] <user> <password>
<connect_string> <protocol> <lang> <source_file>
```

On Windows, use the following command:

```
java -classpath ";<JREPATH>\rt.jar;<Workflow_JAR_file_directory>;
<Workflow_JAR_file_directory>\wfjava.jar;
<Workflow_JAR_file_directory>\wfapi.jar;
<Workflow_JAR_file_directory>\fndctx.jar;
<ORACLE_HOME>\jdbc\lib\classes111.zip;"
oracle.apps.fnd.wf.WFXLoad -u[f] <user> <password>
<connect_string> <protocol> <lang> <source_file>
```

When running the Workflow XML Loader, use either the -u option or the -uf option to specify the upload mode that you want.

*   -u - Normal upload mode. The Workflow XML Loader loads the object definitions from the source XML file into the Business Event System tables in the database, but does not make any updates to events or subscriptions with a customization level of User.

- `-uf` - Force upload mode. The Workflow XML Loader loads the object definitions from the source XML file into the Business Event System tables in the database and overwrites any existing definitions, even for events or subscriptions with a customization level of User.

Additionally, replace the variables with your parameters as follows:

- `<user>` - The user name of your database account.

- `<password>` - The password for your database account.

- `<connect_string>` - The connect string for the database. The format of the connect string depends on the JDBC driver type.

  - For a JDBC OCI8 driver, the connect string should be the database name as specified in its TNSNAMES entry, in the following format:

    `<database_name>`

  - For a JDBC THIN driver, the connect string should include the host name, port number, and database system identifier (SID) in the following format:

    `<host_name>`:`<port_number>`:`<database_SID>`

- `<protocol>` - The JDBC driver type you want to use to connect to the database. The JDBC driver type can be either `oci8` or `thin`.

- `<lang>` - The abbreviation for the language of the XML file. This parameter is case insensitive. Use the standard language abbreviations for the Oracle Database, such as `US` for American or `JA` for Japanese. For a list of the standard language abbreviations, see: Locale Data, *Oracle National Language Support Guide.*

- `<source_file>` - The name and full path of the source file from which you want to upload definitions. The source file should have the extension `.wfx`.

Each subscription definition that you want to upload must include a phase number, owner name, and owner tag. The Workflow XML Loader cannot upload a source file containing subscription definitions that are missing this information.

If the source file contains subscription definitions downloaded in exact mode, the subscriptions are loaded into the Business Event System tables according to their global unique identifiers. However, if the subscription definitions were downloaded in normal mode, using tokens, each subscription is identified instead by a unique combination of subscribing system, source type, triggering event, source agent, owner name, and owner tag.

> **Note:** To report more extensive debugging information in the program output, you can run the Workflow XML Loader in debug mode by including the DebugMode argument just before the `-u` option. For

example, use the following command:

```
java -classpath <classpath> oracle.apps.fnd.wf.WFXLoad
DebugMode -u[f] <user> <password> <connect_string>
<protocol> <lang> <source_file>
```

9

Workflow Administration Scripts

This chapter describes the SQL scripts that workflow administrators can run against an Oracle Workflow server installation.

This chapter covers the following topics:

- Workflow Administration SQL Scripts

Workflow Administration SQL Scripts

You can use any of the following administrative scripts to help set up and maintain various features in Oracle Workflow. The scripts are located in the `$FND_TOP/sql` directory.

- Update translation tables - WFNLADD.sql, page 9-5.

- Enable/disable a language - wfnlena.sql, page 9-10

- Run a workflow process - wfrun.sql, page 9-13.

- Show activities deferred for the next background engine execution - wfbkgchk.sql, page 9-6.

- Display a status report for an item

  - wfstatus.sql, page 9-14.

  - wfstat.sql, page 9-14.

- Show a notification's status - wfntfsh.sql, page 9-11.

- Display debugging information for a notification - wfmlrdbg.sql, page 9-10.

- Reset the protection level for objects - wfprot.sql, page 9-11.

- Handle errored activities - wfretry.sql, page 9-11.

- Check for version and process definition errors

  - wfverchk.sql, page 9-14.

  - wfverupd.sql, page 9-14.

  - wfstdchk.sql, page 9-14.

- Check for invalid hanging foreign keys - wfrefchk.sql, page 9-11.

- Check the directory service data model - wfdirchk.sql, page 9-8.

- Change the internal name of a workflow object

  > **Note:** Generally, you cannot update the internal name of a
  > workflow object in Oracle Workflow Builder. However, if you load
  > your process definition to a database, you can use one of these
  > scripts to update a workflow object's internal name if no runtime
  > data exists for the object. You should only use these scripts to
  > correct errors in an object's internal name during design time. Do
  > not use these scripts to rename objects that are involved in running
  > instances of processes.

  - wfchact.sql, page 9-6.

  - wfchacta.sql, page 9-7.

  - wfchita.sql, page 9-7.

  - wfchitt.sql, page 9-7.

  - wfchluc.sql, page 9-7.

  - wfchlut.sql, page 9-8.

  - wfchmsg.sql, page 9-8.

  - wfchmsga.sql, page 9-8.

- Remove data from Oracle Workflow tables

  - wfrmall.sql, page 9-12.

  - wfrmitms.sql, page 9-12.

- wfrmitt.sql, page 9-13.

- wfrmtype.sql, page 9-13.

- wfrmita.sql, page 9-12.

> **Note:** A concurrent program called "Purge Obsolete Workflow Runtime Data" is also available. See: FNDWFPR, page 9-3.

- Display the version of the Oracle Workflow server - wfver.sql, page 9-14.

- Stop the Java Function Activity Agent - wfjvstop.sql, page 9-10.

- Enqueue an event message on a queue using an override agent - wfevtenq.sql, page 9-9.

- Run a listener to monitor an agent for inbound event messages - wfagtlst.sql, page 9-5.

- Display debugging information for an event - wfbesdbg.sql, page 9-5.

FNDWFPR

Use the standard concurrent program FNDWFPR "Purge Obsolete Workflow Runtime Data" to purge old data from the Oracle Workflow tables regularly.

This program purges obsolete runtime information associated with work items, including status information and any associated notifications and Oracle XML Gateway transactions. By default, it also purges obsolete design information, such as activities that are no longer in use and expired ad hoc users and roles, and obsolete runtime information not associated with work items, such as notifications or Oracle XML Gateway transactions that were not handled through a workflow process. You can optionally choose to purge only core runtime information associated with work items for performance gain during periods of high activity, and purge all obsolete information as part of your routine maintenance during periods of low activity.

To preserve electronic signature evidence for future reference, this program by default does not delete any notifications that required signatures or their associated signature information. If you do not need to maintain signature evidence, you can choose to delete signature-related information as well.

Navigate to the Submit Requests form in Oracle Applications to submit the Purge Obsolete Workflow Runtime Data concurrent program. When you install and set up Oracle Applications and Oracle Workflow, your system administrator needs to add this concurrent program to a request security group for the responsibility that you want to run this program from. See: Overview of Concurrent Programs and Requests, *Oracle Applications System Administrator's Guide* and Running Reports and Programs, *Oracle*

Applications User's Guide.

You can supply the following parameters for the Purge Obsolete Workflow Runtime Data concurrent program:

- Item Type - The item type to purge. Leaving this field blank defaults to purging the runtime data for all item types.

- Item Key - The item key to purge. Leaving this field blank defaults to purging the runtime data for all item keys.

- Age - Minimum age of data to purge, in days. The default is `0`.

- Persistence Type - The persistence type to be purged, either `Temporary` or `Permanent`. The default is `Temporary`

- Core Workflow Only - Enter '`Y`' to purge only obsolete runtime data associated with work items, or '`N`' to purge all obsolete runtime data as well obsolete design data. The default is '`N`'.

- Commit Frequency - The number of records to purge before the program commits data. To reduce rollback size and improve performance, set this parameter to commit data after a smaller number of records. The default is 500 records.

    > **Note:** After performing a commit, the program resumes purging work items with the next subsequent begin date. In some cases, if additional items have the same begin date as the last item that was purged before a commit, the program may not purge all eligible items. To purge these remaining work items, simply rerun the program.

- Signed Notifications - Enter '`N`' to preserve signature evidence, including notifications that required electronic signatures and their associated signature information. Enter '`Y`' to purge signature-related information. The default is '`N`'.

    > **Note:** You can use Oracle Workflow Manager to submit and manage the Purge Obsolete Workflow Runtime Data concurrent program. See: Purging Workflow Data, page 7-81.

    > Additionally, you can use the Purge Obsolete ECX Data concurrent program to purge Oracle XML Gateway transactions according to Oracle XML Gateway-specific parameters. See: Purge Obsolete ECX Data Concurrent Program, *Oracle XML Gateway User's Guide*.

WFNLADD.sql

If you enable a new language in your Oracle installation, use `WFNLADD.sql` to add the missing rows for that language to the Oracle Workflow translation tables. See: Setting Up Additional Languages, page 2-36 and wfnlena.sql, page 9-10.

Use the script as follows:

```
sqlplus <user/pwd> @WFNLADD
```

> **Note:** The `wfnlena.sql` and `WFNLADD.sql` scripts prepare the database for loading workflow definitions in another language. However, these scripts by themselves do not enable displaying Oracle Workflow Web pages in other languages. See: Setting Up Additional Languages, page 2-36.

Wfagtlst.sql

Use `wfagtlst.sql` to run a PL/SQL agent listener to monitor an agent for inbound event messages. When a message is received, the Event Manager searches for and executes any enabled subscriptions by the local system to the event with a source type of External, and also any enabled subscriptions by the local system to the Any event with a source type of External.

Use the script as follows:

```
sqlplus <user/pwd> @wfagtlst <agent_name>
```

Replace `<agent_name>` with the internal name of the agent that you want to monitor for inbound event messages.

> **Note:** You should use this script primarily for debugging purposes.

Related Topics

Listen, *Oracle Workflow API Reference*

Wfbesdbg.sql

Use `wfbesdbg.sql` to display debugging information for an event that you can use to check the status of the event or investigate errors. This script creates an output file named `wfevtdbg<event_key>.html`, where `<event_key>` is the event key for a particular instance of the event. The output file displays details about the Business Event System, including the statuses of the local system and agent listeners, details about the definitions of the specified event and any subscriptions to that event, and details about the specified instance of the event if it appears on one of these queues:

- WF_DEFERRED

- WF_JAVA_DEFERRED

- WF_ERROR

- WF_JAVA_ERROR

Use the script as follows:

```
sqlplus <user/pwd> @wfbesdbg <event_name> <event_key>
```

Replace *<event_name>* with the name of the event and *<event_key>* with the event key that identifies the particular instance of the event. The event name and event key parameters are case-sensitive.

> **Note:** You can also obtain this information by running the Event Diagnostic Test in Oracle Diagnostics. See: Event Diagnostic Test, page E-7.

Wfbkgchk.sql

Use `wfbkgchk.sql` to get a list of all activities waiting to be processed by the background engine the next time it runs.

Use the script as follows:

```
sqlplus <user/pwd> @wfbkgchk <WF_schema>
```

Replace *<WF_schema>* with the Oracle Workflow schema, usually `applsys`.

Related Topics

Background, *Oracle Workflow API Reference*

Setting up Background Workflow Engines, page 2-39

Wfchact.sql

Use `wfchact.sql` to change the internal name of an activity and update all references to the activity. See: Change the internal name of a workflow object, page 9-2.

Use the script as follows:

```
sqlplus <user/pwd> @wfchact <act_type> <old_act> <new_act>
```

Replace *<act_type>* with the item type that the activity you wish to update is associated with, replace *<old_act>* with the current internal name of the activity, and replace *<new_act>* with the new internal name of the activity.

Wfchacta.sql

Use `wfchacta.sql` to change the internal name of an activity attribute and update all references to the activity attribute. See: Change the internal name of a workflow object, page 9-2.

Use the script as follows:

```
sqlplus <user/pwd> @wfchacta <act_type> <old_acta> <new_acta>
```

Replace `<act_type>` with the item type that the activity attribute you wish to update is associated with, replace `<old_acta>` with the current internal name of the activity attribute, and replace `<new_acta>` with the new internal name of the activity attribute.

Wfchita.sql

Use `wfchita.sql` to change the internal name of an item attribute and update all references to the item attribute. See: Change the internal name of a workflow object, page 9-2.

Use the script as follows:

```
sqlplus <user/pwd> @wfchita <item_type> <old_attr> <new_attr>
```

Replace `<item_type>` with the item type that the item attribute you wish to update is associated with, replace `<old_attr>` with the current internal name of the item attribute, and replace `<new_attr>` with the new internal name of the item attribute.

Wfchitt.sql

Use `wfchitt.sql` to change the internal name of an item type and update all references to the item type. See: Change the internal name of a workflow object, page 9-2.

Use the script as follows:

```
sqlplus <user/pwd> @wfchitt <old_type> <new_type>
```

Replace `<old_type>` with the current internal name of the item type, and replace `<new_type>` with the new internal name of the item type.

Wfchluc.sql

Use `wfchluc.sql` to change the internal name of a lookup code and update all references to the lookup code. See: Change the internal name of a workflow object, page 9-2.

Use the script as follows:

```
sqlplus <user/pwd> @wfchluc <lookup_type> <old_luc> <new_luc>
```

Replace `<lookup_type>` with the lookup type of the lookup code you wish to update, replace `<old_luc>` with the current internal name of the lookup code, and replace

<new_luc> with the new internal name of the lookup code.

Wfchlut.sql

Use `wfchlut.sql` to change the internal name of a lookup type and update all references to the lookup type. See: Change the internal name of a workflow object, page 9-2.

Use the script as follows:

`sqlplus <user/pwd> @wfchlut <old_lut> <new_lut>`

Replace *<old_lut>* with the current internal name of the lookup type, and replace *<new_lut>* with the new internal name of the lookup type.

Wfchmsg.sql

Use `wfchmsg.sql` to change the internal name of a message and update all references to the message. See: Change the internal name of a workflow object, page 9-2.

Use the script as follows:

`sqlplus <user/pwd> @wfchmsg <msg_type><old_msg> <new_msg>`

Replace *<msg_type>* with the item type of the message you wish to update, replace *<old_msg>* with the current internal name of the message, and replace *<new_msg>* with the new internal name of the message.

Wfchmsga.sql

Use `wfchmsga.sql` to change the internal name of a message attribute. This script does not update the message subject or message body references to the message attribute, however. You must manually update the message attribute references. See: Change the internal name of a workflow object, page 9-2.

Use the script as follows:

`sqlplus <user/pwd> @wfchmsga <msg_type> <msg_name> <old_attr> <new_attr>`

Replace *<msg_type>* with the item type of the message attribute you wish to update, replace *<msg_name>* with the internal name of the message that the message attribute belongs to, replace *<old_attr>* with the current internal name of the message attribute, and replace *<new_attr>* with the new internal name of the message attribute.

Wfdirchk.sql

Use `wfdirchk.sql` to check for the following conditions in your directory service data model:

• Invalid internal names that contain the characters '#', ':', or '/' in WF_USERS.

- Invalid compound names in WF_USERS or WF_ROLES.

- Duplicate names in WF_USERS or WF_ROLES.

- Multiple names in WF_USERS or WF_ROLES linked to the same row in the original repository.

- Missing display names in WF_USERS or WF_ROLES.

- Invalid notification preference or null e-mail address if the notification preference is set in WF_USERS or WF_ROLES to receive e-mail.

- Invalid status in WF_USERS.

- Rows in WF_USERS that do not have a corresponding row in WF_ROLES.

- Invalid internal names in WF_ROLES that contain the characters '#' or '/' or have a length greater than 30 characters.

- Invalid user/role foreign key in WF_USER_ROLES.

- Missing user/role in WF_USER_ROLES. Every user must participate in its own role.

- Duplicate rows in WF_USER_ROLES.

`Wfdirchk.sql` should return no rows to ensure that your directory service data model is correct.

Use the script as follows:

```
sqlplus <user/pwd> @wfdirchk
```

Wfevtenq.sql

Use `wfevtenq.sql` to enqueue an event message on a local queue using an override agent. This script constructs an event message using the event name, event key, event data, From Agent, and To Agent you specify. Then the event message is enqueued on the queue associated with the override agent you specify, which can be different than the From Agent listed inside the event message. If no override agent is specified, the event message is enqueued on the message's From Agent by default.

> **Note:** This script can only enqueue an event message onto a queue for an agent on the local system.

Use the script as follows:

```
sqlplus <user/pwd> @wfevtenq <overrideagent><overridesystem><fromagent>
<fromsystem><toagent><tosystem><eventname><eventkey><message>
```

Replace the variables with your parameters as follows:

- *`<overrideagent>`* - The agent on whose queue you want to enqueue the event message.

- *`<overridesystem>`* - The system where the override agent is located.

- *`<fromagent>`* - The From Agent that you want to list in the event message.

- *`<fromsystem>`* - The system where the From Agent is located.

- *`<toagent>`* - The To Agent that receives the event message.

- *`<tosystem>`* - The system where the To Agent is located.

- *`<eventname>`* - The internal name of the event.

- *`<eventkey>`* - The event key that uniquely identifies the instance of the event.

- *`<message>`* - The event data.

Wfjvstop.sql

Use `wfjvstop.sql` to stop the Java Function Activity Agent by placing a stop message on the 'Outbound' queue.

Use the script as follows:

```
sqlplus <user/pwd> @wfjvstop
```

Wfmlrdbg.sql

Use `wfmlrdbg.sql` to display debugging information for a notification that you can use to check the status of the notification or investigate errors. This script creates an output file named `wfmlrdbg<nid>.html`, where *`<nid>`* is the notification ID (NID).

Use the script as follows:

```
sqlplus <user/pwd> @wfmlrdbg <nid>
```

Replace *`<nid>`* with the notification ID for the notification.

> **Note:** You can also obtain this information by running the Mailer Diagnostic Test in Oracle Diagnostics. See: Mailer Diagnostic Test, page E-8.

Wfnlena.sql

If you define a new language in your Oracle installation, use `wfnlena.sql` to enable or disable that language in Oracle Workflow. See: WFNLADD.sql, page 9-5.

Use the script as follows:

```
sqlplus <user/pwd> @wfnlena <language_code> <enable_flag>
```

Replace *<language_code>* with a valid language code, and replace *<enable_flag>* with Y to enable and N to disable the specified language.

> **Note:** The `wfnlena.sql` and `WFNLADD.sql` scripts prepare the database for loading workflow definitions in another language. However, these scripts by themselves do not enable displaying Oracle Workflow Web pages in other languages. See: Setting Up Additional Languages, page 2-36.

Wfntfsh.sql

Use `wfntfsh.sql` to display status information about a particular notification, given its notification ID.

Use the script as follows:

```
sqlplus <user/pwd> @wfntfsh <notification_id>
```

Wfprot.sql

Use `wfprot.sql` to reset the protection level of all objects associated with a specified item type.

> **Important:** If you reset the protection level for all objects in an item type, then none of those objects in the item type will be customizable by users operating at an access level higher than the new protection level.

Use the script as follows:

```
sqlplus <user/pwd> @wfprot <item_type><protection_level>
```

Replace *<item_type>* with the item type that you want to reset the protection level for, and replace *<protection_level>* with the new protection level.

Wfrefchk.sql

Use `wfrefchk.sql` to check for invalid workflow data that is missing primary key data for a foreign key.

```
sqlplus <user/pwd> @wfrefchk
```

Wfretry.sql

Use `wfretry.sql` to display a list of activities that have encountered an error for a given process instance and then specify whether to skip, retry, or reset any one of those

errored activities.

Use the script as follows:

```
sqlplus <user/pwd> @wfretry <item_type><item_key>
```

Provide an item type and item key to uniquely identify an item or process instance. The script first returns the list of errored activities by label name. The script then prompts you for the label name of an activity that you wish to skip, retry, or reset. If you choose skip, then you must also specify the result that you want the skipped activity to have.

> **Note:** This script calls the WF_ENGINE HandleError API, so you can actually specify the label name of any activity associated with the specified item type and item key to perform a rollback. See: HandleError, *Oracle Workflow API Reference*.

Wfrmall.sql

Use `wfrmall.sql` to delete all data in all Oracle Workflow runtime and design time tables.

Use the script as follows:

```
sqlplus <user/pwd> @wfrmall
```

> **Warning:** This script deletes **ALL** workflow definitions. Do not use this script unless you are absolutely sure you want to remove all workflow data from your runtime and design time tables.
>
> After you run this script, you should also reload the workflow definitions for the Standard, System: Mailer, and System: Error item types stored in the files `wfstd.wft`, `wfmail.wft`, and `wferror.wft`, respectively.

Wfrmita.sql

Use `wfrmita.sql` to delete all workflow data for a specified item type attribute. This script prompts you for the item type and the name of the attribute to delete. Alternatively, you can use Oracle Workflow Builder to delete an item type attribute from a workflow definition stored in a file or a database.

Use the script as follows:

```
sqlplus <user/pwd> @wfrmita
```

Wfrmitms.sql

Use `wfrmitms.sql` to delete status information in Oracle Workflow runtime tables for a particular item. This script prompts you to choose between deleting all data

associated with a specified item type and item key or deleting only data for the completed activities of the specified item type and item key.

Use the script as follows:

```
sqlplus <user/pwd> @wfrmitms <item_type> <item_key>
```

Wfrmitt.sql

Use `wfrmitt.sql` to delete all data in all Oracle Workflow design time and runtime tables for a particular item type. This script prompts you for an item type from a list of valid item types.

Use the script as follows:

```
sqlplus <user/pwd> @wfrmitt
```

> **Warning:** This script deletes **ALL** workflow data for the specified item type.

Wfrmtype.sql

Use `wfrmtype.sql` to delete runtime data associated with a given item type. This script prompts you for an item type to purge from a list of valid item types, then asks you to choose between deleting all runtime data associated with the specified item type or deleting only runtime data for the completed activities and items of the specified item type.

Use the script as follows:

```
sqlplus <user/pwd> @wfrmtype
```

Wfrun.sql

Use `wfrun.sql` to create and start a specified process.

Use the script as follows:

```
sqlplus <user/pwd> @wfrun <item_type> <item_key> <process_name>
```

> **Note:** The item key for a process instance can only contain single-byte characters. It cannot contain a multibyte value.

Wfstat.sql

Use `wfstat.sql` to display a developer status report for a specified item. The output is 132 characters per line.

Use the script as follows:

```
sqlplus <user/pwd> @wfstat <item_type> <item_key>
```

Wfstatus.sql

Use `wfstatus.sql` to display an end user status report for a specified item. The output is 132 characters per line.

Use the script as follows:

```
sqlplus <user/pwd> @wfstatus <item_type> <item_key>
```

Wfstdchk.sql

Use `wfstdchk.sql` to check and report any problems found in the Oracle Workflow data model. For example, this script will report any function activities that reference invalid functions and scan the tables of each workflow process definition object to verify that each row has a valid internal name and display name.

Use the script as follows:

```
sqlplus <user/pwd> @wfstdchk
```

Wfver.sql

Use `wfver.sql` to display the version of the Oracle Workflow server, the status and version of the Oracle Workflow PL/SQL packages, and the version of the Oracle Workflow views installed.

Use the script as follows:

```
sqlplus <user/pwd> @wfver
```

Wfverchk.sql

Use `wfverchk.sql` if you suspect that problems arising in your workflow process are due to multiple versions of an activity being active simultaneously. This script identifies errors in versions of activities that cause multiple versions to appear to be active at once.

Use the script as follows:

```
sqlplus <user/pwd> @wfverchk
```

Wfverupd.sql

Use `wfverupd.sql` to correct problems arising in your workflow process that are due to multiple versions of an activity being active simultaneously. This script identifies and corrects errors in versions of activities that cause multiple versions to appear to be active at once.

Use the script as follows:

```
sqlplus <user/pwd> @wfverupd
```

A

Oracle Workflow Administrator Navigation Paths

This appendix lists the navigation paths to Oracle Workflow administrator Web pages in the seeded Oracle Workflow responsibilities for Oracle Applications.

This appendix covers the following topics:

- Oracle Workflow Administrator Navigation Paths

Oracle Workflow Administrator Navigation Paths

This table shows the navigation paths to Oracle Workflow administrator Web pages in the seeded Oracle Workflow responsibilities for Oracle Applications. Your system administrator may have customized the responsibilities, menus, and navigation paths in your installation.

Oracle Workflow Administrator Navigation Paths

Web Page	Standard Navigation Path
Workflow Configuration, page 2-9	Workflow Administrator Web Applications: Administration > Workflow Preferences *or* Workflow Administrator Web (New): Administration > Workflow Preferences *or* Workflow Administrator Event Manager: Administration > Workflow Preferences
Administrator Home, page 4-1	Workflow Administrator Web Applications: Home *or* Workflow Administrator Web (New): Home *or* Workflow Administrator Event Manager: Home

Web Page	Standard Navigation Path
Administrator Status Monitor, page 5-3	Workflow Administrator Web Applications: Status Monitor *or* Workflow Administrator Web (New): Status Monitor *or* Workflow Administrator Event Manager: Status Monitor
Administrator Notification Search, page 6-1	Workflow Administrator Web Applications: Administration > Notification Search *or* Workflow Administrator Web (New): Administration > Notification Search *or* Workflow Administrator Event Manager: Administration > Notification Search
Administrator Vacation Rules, page 6-4	Workflow Administrator Web Applications: Administration > Vacation Rules *or* Workflow Administrator Web (New): Administration > Vacation Rules *or* Workflow Administrator Event Manager: Administration > Vacation Rules
Signature Evidence Store, page 6-5	Workflow Administrator Web Applications: Administration > Signature Evidence Store*or* Workflow Administrator Web (New): Administration > Signature Evidence Store*or* Workflow Administrator Event Manager: Administration > Signature Evidence Store
Worklist Flexfields Rules, page 6-16	Workflow Administrator Web Applications: Worklist Flexfields Rules *or* Workflow Administrator Web (New): Worklist Flexfields Rules
Worklist Flexfields Rules Simulation, page 6-21	Workflow Administrator Web Applications: Worklist Flexfields Rules Simulation *or* Workflow Administrator Web (New): Worklist Flexfields Rules Simulation
Oracle Workflow Manager component of Oracle Applications Manager, page 7-1	Workflow Administrator Web Applications: Workflow Manager *or* Workflow Administrator Web (New): Workflow Manager *or* Oracle Applications Manager direct login page

Note: The Workflow Administrator Web (New) responsibility is

provided for reference to highlight the Oracle Application Framework-based functionality in Oracle Workflow. The same functionality is also available on the Workflow Administrator Web Applications and Workflow Administrator Event Manager responsibilities.

B

Oracle Workflow Administrator Personalizations

This appendix lists features that you can add to Oracle Workflow administrator Web pages through Oracle Application Framework Personalization.

This appendix covers the following topics:

- Oracle Workflow Administrator Personalizations

Oracle Workflow Administrator Personalizations

Oracle Application Framework Personalization provides the ability to customize Oracle self-service Web application pages. To personalize your pages, you must log in as a user or with a responsibility for which the Personalize Self-Service Defn profile option is set to Yes. You can then choose the Personalize Page global button to create personalizations for an entire page, or choose the Personalize link in a region to create personalizations for that region only.

Oracle Workflow supports several types of personalizations. This appendix lists features that do not appear in the Oracle Workflow administrator Web pages by default, but are available for you to add if you choose.

Many Oracle Workflow Web pages also let you perform other personalizations, such as displaying additional information columns in a list, changing the sort order of a list, or hiding features you do not want to use. To view all the personalizations that are available for a particular page, choose the Personalize button for that page.

For more information about creating personalizations, see: Personalization, *Oracle Application Framework Personalization Guide*.

Electronic Signature Page (Signature Evidence Store)

The Notification ID column lets you show the numerical ID of the notification for which the signature was requested or submitted.

Find Worklist Flexfields Rule Map Conflicts Page

The Message Attribute column lets you show the internal name of each message attribute. The Attribute Display Name column, which shows the message attribute display name, is displayed by default; you can hide it if you add the Message Attribute column instead, or you can display both columns.

Worklist Flexfields Rule Simulation Page

The Attribute Name column lets you show the internal name of each message attribute. The Attribute Display Name column, which shows the message attribute display name, is displayed by default; you can hide it if you add the Attribute Name column instead, or you can display both columns.

C

Oracle Workflow Performance Concepts

This appendix describes concepts and techniques that you can use to enhance performance when running Oracle Workflow.

This appendix covers the following topics:

- Oracle Workflow Performance Concepts

Oracle Workflow Performance Concepts

The performance of Oracle Workflow depends on several different factors. You can enhance performance in your workflow process design through effective use of synchronous, asynchronous, and forced synchronous processes, item attributes, message attributes, subprocesses, and deferred activities. You can also use partitioning and purging techniques to address performance issues associated with large quantities of runtime data. Additionally, you can enhance Business Event System performance by adjusting its maximum cache size and by enabling static function calls for custom PL/SQL functions.

For more information, see: *The Oracle Applications Tuning Handbook* by Andy Tremayne and Steve Mayze (Oracle Press, ISBN 0-07-212549-7) and *Oracle Database Concepts*.

Designing Workflow Processes for Performance

You can enhance the performance of your workflow processes through effective process design.

Synchronous, Asynchronous, and Forced Synchronous Workflows

When designing a workflow process, you must decide whether you want it to be executed as a synchronous, asynchronous, or forced synchronous process. The process design impacts the amount of time it takes for the Workflow Engine to return control to the calling application that initiates the process.

- **Synchronous** - A synchronous process contains only activities that can be executed

immediately, so that the Workflow Engine executes the process without interruption from start to finish. The Workflow Engine does not return control to the calling application until it completes the process. With a synchronous process, you can immediately check for process results that were written to item attributes or directly to the database. However, the user must wait for the process to complete. If the process takes a long time, the application may appear to hang. In this case, you should change the process to an asynchronous process.

- **Asynchronous** - An asynchronous process is a process that the Workflow Engine cannot complete immediately because it contains activities that interrupt the flow. Examples of activities that force an asynchronous process include deferred activities, notifications with responses, blocking activities, and wait activities. Rather than waiting indefinitely when it encounters one of these activities, the Workflow Engine sets the audit tables appropriately and returns control to the calling application. The workflow process is left in an unfinished state until it is started again, usually by the Notification System, Business Event System, or the background engine. With an asynchronous process, the user does not have to wait for the process to complete to continue using the application. However, the results of the process are not available until the process is completed at a later time.

- **Forced synchronous** - A forced synchronous process completes in a single SQL session from start to finish and never inserts into or updates any database tables. As a result, the execution speed of a forced synchronous process is significantly faster than a typical synchronous process. The process results are available immediately upon completion. However, no audit trail is recorded. You may want to use a forced synchronous process if your application needs to generate a specific result quickly and recording an audit trail is not a concern. To create a forced synchronous process, you must set the item key of your process to #SYNCH and follow certain restrictions in designing your process, such as not including any notification activities.

See: Overview of the Workflow Engine, *Oracle Workflow API Reference* and Synchronous, Asynchronous, and Forced Synchronous Processes, *Oracle Workflow API Reference*.

Item Attributes

Item attributes act as global variables that can be referenced or updated by any activity within a workflow process. The number of item attributes directly affects the startup time for work items, because the Workflow Engine by default creates runtime copies of all item attributes when a new work item is created. For this reason, item attributes should be kept to a minimum.

You can optionally enhance performance for a process by specifying that the Workflow Engine should create runtime copies of item attributes only on demand. To do so, define a special activity attribute named #ONDEMANDATTR for the top-level runnable process activity. In this case, the Workflow Engine creates runtime copies of item attributes only when the process sets values for those item attributes. Otherwise, the

Workflow Engine simply references the default values specified for the item attributes in the design-time workflow definition. See: #ONDEMANDATTR Attribute, *Oracle Workflow Developer's Guide*.

Item attributes should be used for:

- Storing working information for the work item.

- Token replacement for messages. For messages where the number of lines may vary, such as in repeating groups, do not create individual item attributes for each line. Instead, use item and message attributes of type Document to combine the lines together.

- Storing primary key values so that functions can look up all necessary values from the database.

- Temporary placeholders to set activity attributes dynamically. For example, the performer of a notification may only be known at runtime, so you can reference an item attribute and seed the desired value just before executing the notification.

Item attributes should reference static values or values that are not in the database so that there are no concerns about keeping the values synchronized. (Primary key values, however, do not change.) Do not implement every column within a table as an item attribute.

See: Item Type Attributes, *Oracle Workflow Developer's Guide* and To Define an Item Type or Activity Attribute, *Oracle Workflow Developer's Guide*.

Item attribute types that can help you reduce the number of attributes you need include the following:

- Document - The attribute value is an embedded or attached document, which enables a complex structure to be rendered inline, or attached to notifications. You can specify the following types of documents:

  - PL/SQL document - A document representing data from the database as a character string, generated from a PL/SQL procedure.

  - PL/SQL CLOB document - A document representing data from the database as a character large object (CLOB), generated from a PL/SQL procedure.

  - PL/SQL BLOB document - A document representing data from the database as a binary large object (BLOB), generated from a PL/SQL procedure.

  - Oracle Application Framework region - A JSP call to an Oracle Application Framework region for inclusion in a notification message.

- Role - The attribute value is the internal name of a role. If a message attribute of type role is included in a notification message, the attribute automatically resolves to the role's display name, eliminating the need for you to maintain separate

attributes for the role's internal and display names. Also, when you view the notification from a Web browser, the role display name is a hypertext link to the e-mail address for that role.

See: Attribute Types, *Oracle Workflow Developer's Guide*.

Whenever multiple item attributes will be created, or multiple item attribute values will be set during workflow processing, use the array versions of the Add Item Attribute and Set Item Attribute workflow engine APIs (AddItemAttributeArray and SetItemAttributeArray, respectively). These APIs will significantly decrease the number of calls to Workflow Engine APIs, which can have a measurable impact on performance during batch processing. See: AddItemAttributeArray, *Oracle Workflow API Reference* and SetItemAttributeArray, *Oracle Workflow API Reference*.

Message Attributes

To enhance performance, message attributes should be kept to a minimum. For messages where the number of lines may vary, such as in repeating groups, do not create individual item and message attributes for each line (LINE_INFO1, LINE_INFO2, etc.). Instead, use item and message attributes of type Document to combine the lines together.

See: Attribute Types, *Oracle Workflow Developer's Guide*, Send and Respond Message Attributes, *Oracle Workflow Developer's Guide*, and To Define a Message Attribute, *Oracle Workflow Developer's Guide*.

Subprocesses

When you design a workflow process, you can group a collection of activities together in a process activity which represents a subprocess within the main process. Using subprocesses judiciously can help make workflow diagrams clearer and easier to read and can simplify workflow monitoring and maintenance. However, subprocesses also result in additional DML operations and additional state information stored in Workflow tables. Consequently, you should avoid unnecessary use of subprocesses when there is no functional benefit.

For example, the following two processes, Process 1 and Process 2, are functionally identical, both performing a function called Function 1. However, they result in different numbers of state rows being stored in Workflow tables.

Process 1 contains a Start activity, a Subprocess activity, and an End activity. The subprocess contains a Start activity, the Function 1 activity, and an End activity. This process stores 7 state rows in Workflow tables.

Process 1

Subprocess
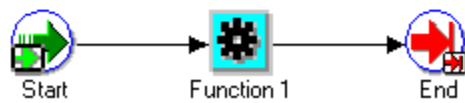
Process 2 simply contains a Start activity, the Function 1 activity, and an End activity. This process stores only 4 state rows in Workflow tables.

Example Process 2

Process 2
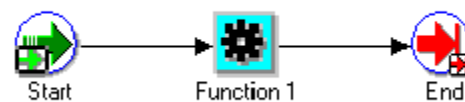
Because more rows are stored in Workflow tables, the kind of design shown in the Process 1 diagram will result in slower workflow throughput and a need to purge Workflow runtime tables more frequently than what should be necessary with the Process 2 design.

> **Note:** This guideline is not meant to imply that subprocesses should not be used at all. Collapsing all subprocesses can make workflow diagrams unreadable and difficult to maintain. This recommendation merely highlights that unnecessary overuse of subprocesses can have a negative performance impact.

See: Process Activity, *Oracle Workflow Developer's Guide* and To Create a Process Activity, *Oracle Workflow Developer's Guide*.

Deferring Activities

The simplest and most effective way to improve the online user's response time is to defer function activities. You may want to defer activities that require a large amount of processing resource or time to complete. Oracle Workflow lets you manage the load on the Workflow Engine and the response time for the user by setting up supplemental engines to run these costly activities as background tasks. In these cases, the costly activity is deferred by the Workflow Engine and run later by a background engine.

When an activity is deferred, the main Workflow Engine can then continue to the next available activity, which may occur on some other parallel branch of the process. If no other activity is available to be executed, the Workflow Engine returns control immediately to the calling application. The user remains unaware that processing is still taking place, rendering a faster execution time.

To defer an activity, set the activity's cost above the default threshold cost at design time. The threshold cost is a PL/SQL package variable with a default value of 50 hundredths of a second. Set a cost above this threshold for all activities that you don't want the user to wait for.

At runtime, the Workflow Engine defers any thread to the background as soon as it encounters an activity with a cost higher than the threshold. Then the background engine later identifies the process as deferred and continues its execution.

In addition to deferred activities, background engines also handle timed out activitites and stuck processes. You can run as many background engines as you want. You must have at least one background engine that can check for timed out activities, one that can process deferred activities, and one that can handle stuck processes. At a minimum, you need to set up one background engine that can handle both timed out and deferred activities as well as stuck processes.

Generally, you should run a separate background engine to check for stuck processes at less frequent intervals than the background engine that you run for deferred activities, normally not more often than once a day. Run the background engine to check for stuck processes when the load on the system is low.

See: Deferred Processing, *Oracle Workflow API Reference*, Activity Cost, *Oracle Workflow Developer's Guide*, Setting Up Background Workflow Engines, page 2-39, and To Set Engine Thresholds, page 2-43.

Managing Runtime Data for Performance

When the Workflow Engine executes any type of workflow other than forced synchronous processes, status information is stored in runtime tables. The amount of data stored in these tables will grow depending on the complexity and number of workflows being executed.

Performance issues associated with large quantities of runtime data can be addressed by:

- Partitioning

- Purging

Partitioning for Performance

Partitioning addresses key issues in supporting very large tables and indexes by letting you decompose them into smaller and more manageable pieces called partitions. SQL queries and DML statements do not need to be modified in order to access partitioned tables. However, once partitions are defined, DDL statements can access and manipulate individual partitions rather than entire tables or indexes. In this way, partitioning can simplify the manageability of large database objects. Also, partitioning is entirely transparent to applications.

You can optionally run a script to partition certain Workflow tables that store runtime status data. This step is highly recommended for performance gain. Before running the script, you should ensure that you have backed up the tables that will be partitioned and that you have allowed sufficient free space and time for the script to run. The script is called `wfupartb.sql` and is located in the `admin/sql` subdirectory under `$FND_TOP`. See: Partitioning Workflow Tables, page 2-8.

Purging for Performance

You can use the Workflow purge APIs to purge obsolete runtime information for completed work items, obsolete runtime information not associated with work items, and obsolete design information. You may want to periodically purge this obsolete data from your system to increase performance. The Workflow purge APIs are defined in the PL/SQL package called WF_PURGE.

The availability of runtime data for purging depends on the persistence type of the item type. The persistence type controls how long a status audit trail is maintained for each instance of the item type.

- If you set an item type's Persistence to Permanent, the runtime status information is maintained indefinitely until you specifically purge the information by calling the procedure *WF_PURGE.TotalPerm( )*.

- If you set an item type's Persistence to Temporary, you must also specify the number of days of persistence ('*n*'). The status audit trail for each instance of a Temporary item type is maintained for at least '*n*' days of persistence after its completion date. After the '*n*' days of persistence, you can then use any of the WF_PURGE APIs to purge the item type's runtime status information. See: WF_PURGE, *Oracle Workflow API Reference*.

- If you set an item type's Persistence to Synchronous, Oracle Workflow expects instances of that item type to be run as forced synchronous processes with an item key of `#SYNCH`. Forced synchronous processes complete in a single SQL session from start to finish and never insert into or update any database tables. Since no

runtime status information is maintained, you do not normally need to perform any purging for a process with the Synchronous persistence type. However, if you run the process with a unique item key in asynchronous mode for testing or debugging purposes, Oracle Workflow does maintain runtime status information for that process instance. You can purge this information by changing the item type's Persistence to Temporary and running any of the WF_PURGE APIs. Then change the item type's Persistence back to Synchronous. See: Synchronous, Asynchronous, and Forced Synchronous Processes, *Oracle Workflow API Reference.*

Additionally, the administration script `wfrmtype.sql` is provided to delete runtime data for a particular item type. This script prompts you for an item type to purge from a list of valid item types, then asks you to choose between deleting all runtime data associated with the specified item type or deleting only runtime data for the completed activities and items of the specified item type. See: Wfrmtype.sql, page 9-13.

Related Topics

WF_PURGE, *Oracle Workflow API Reference*

Persistence Type, *Oracle Workflow Developer's Guide*

Managing Business Event System Performance

The Business Event System caches event, subscription, and agent definitions to enhance performance during subscription processing. The default maximum size of the cache is 50 records. You can optionally increase the maximum cache size to reduce the database queries performed by the Business Event System, or decrease the maximum cache size to reduce the amount of memory used by the cache. See: Changing the Maximum Cache Size for the Business Event System, page 2-157.

If you use custom PL/SQL functions within the Business Event System, including event data generate functions, event subscription rule functions, and queue handler enqueue and dequeue APIs, Oracle Workflow calls those functions using dynamic SQL by default. However, you can enable Oracle Workflow to call your custom functions statically to enhance performance. See: Enabling Static Function Calls for Custom PL/SQL Functions, page 2-157.

D

Oracle Workflow Profile Options

This appendix lists the profile options that you can set to configure Oracle Workflow.

This appendix covers the following topics:

- Oracle Workflow Profile Options

Oracle Workflow Profile Options

You can set the following profile options to configure Oracle Workflow. See: Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide - Maintenance* and Profile Categories, *Oracle Applications System Administrator's Guide - Maintenance*.

Applications SSO Login Types

If your site has implemented single sign-on with Oracle Internet Directory, this profile option determines how users can log in to Oracle Applications.

- SSO - User passwords are maintained only in Oracle Internet Directory. Users must log in to Oracle Applications through single sign-on.

- Local - User passwords are maintained only in Oracle Application Object Library. Users must log in through the local Oracle Applications login page.

- Both - User passwords are maintained both in Oracle Internet Directory and in Oracle Application Object Library. Users can log in to Oracle Applications either through single sign-on or through the local Oracle Applications login page. A user's single sign-on password can be different than that user's Oracle Application Object Library password.

Oracle Workflow supports password-based signatures for notifications based on Oracle Application Object Library passwords. If you have implemented single sign-on, you must set this profile option to either Local or Both for all users who need to enter password-based signatures. You must also ensure that these users have valid passwords defined in Oracle Application Object Library. See: Implementing

Password-based Signatures with Single Sign-On, page 2-140.

Users can view this profile option but cannot update it.

System administrators can view and update this profile option only at the site and user levels.

The internal name for this profile option is APPS_SSO_LOCAL_LOGIN.

This profile option belongs to the Single Sign-On Deployment (FND_SSO_DEPLOYMENT) profile category in the Application Object Library application.

For more information, see: Oracle Single Sign-On Integration, *Oracle Applications System Administrator's Guide - Security*.

WF: BPEL Server

If you define event subscriptions that invoke business process execution language (BPEL) processes, this profile option lets you specify the host name and port of the BPEL server. When this profile option is set, you can use relative URLs rather than absolute URLs in the subscription parameters to specify the location of the Web Services Description Language (WSDL) descriptions for the BPEL processes.

At runtime, Oracle Workflow replaces the `bpel://` prefix in the relative URLs with the host name and port for the BPEL server specified in the WF: BPEL Server profile option. In this way, the profile option lets you quickly change the BPEL server setup for all subscriptions that invoke BPEL processes, such as if you move these subscription definitions from a test instance to a production instance. See: Specifying the BPEL Server, page 2-159 and Invoking a Web Service, *Oracle Workflow Developer's Guide*.

Set the WF: BPEL Server profile option to the host name and port of the BPEL server using the following format:

```
http://<host_name>:<port>/
```

> **Note:** The profile option value must end with a slash (/) in order to form the final WSDL description URL correctly.

Users can view this profile option but cannot update it.

System administrators can view and update this profile option only at site level.

The internal name for this profile option is WF_BPEL_SERVER.

WF: GUEST Access to Notification

This profile option helps control whether users must log in before they can access the Notification Details Web page from a notification. To enable guest access, which does not require an individual login, you must both set this profile option to Enabled and create a grant assigning the "Workflow Guest User permission set" to the GUEST user.

See: Responses through the Notification Detail Link Attachment, page 2-69.

- Enabled - If the required grant has also been created, users can access the Notification Details page as the GUEST user, without an individual login.

- Disabled - Users must always log into Oracle Applications with their individual user names and passwords before they can access the Notification Details page.

Users can view and update this profile option.

System administrators can view and update this profile option only at the site level.

The internal name for this profile option is WF_VALIDATE_NTF_ACCESS.

This profile option belongs to the Workflow (FND_WORKFLOW) profile category and to the Security (FND_SECURITY) profile category in the Application Object Library application.

WF: Notification Reassign Mode

This profile option controls how users can reassign notifications. See: Setting the WF: Notification Reassign Mode Profile Option, page 2-137.

- Delegate - Provides users access to delegate a notification to another user while still retaining ownership of the notification.

- Transfer - Provides users access to transfer complete ownership of a notification to another user.

- Reassign - Provides users access to both the Delegate and Transfer reassign modes. This setting is the default value for this profile option.

Users can view and update this profile option.

System administrators can view and update this profile option at the site, application, responsibility, and user levels.

The internal name for this profile option is FND_NTF_REASSIGN_MODE.

This profile option belongs to the Workflow (FND_WORKFLOW) profile category in the Application Object Library application.

WF: Vacation Rules - Allow All

This profile option controls whether the list of item types for vacation rules includes the "All" option. The "All" option lets users create a rule that applies to notifications associated with any item type. See: Setting Up Notification Handling Options, page 2-138.

- Enabled - The "All" option appears in the list of available item types for vacation rules. This setting is the default value for this profile option.

- Disabled - The "All" option does not appear in the list of available item types. Users must specify the item type to which the vacation rule applies.

After changing the value of this profile option, you must stop and restart Oracle HTTP Server for the change to take effect.

Users can view and update this profile option.

System administrators can view and update this profile option only at site level.

The internal name for this profile option is WF_ROUTE_RULE_ALLOW_ALL.

This profile option belongs to the Workflow (FND_WORKFLOW) profile category in the Application Object Library application.

WF: Workflow Mailer Framework Web Agent

This profile option specifies the host and port of the Web server that notification mailers use to generate the content for Oracle Application Framework regions that are embedded in notifications. If this profile option is not set, notification mailers will use the same Web agent specified in the Application Framework Agent profile option. However, if necessary for load balancing purposes, you can optionally specify a different Web agent for notification mailers to use. See: Setting Up Notification Mailers, page 2-46.

Users can view and update this profile option.

System administrators can view and update this profile option only at site level.

The internal name for this profile option is WF_MAIL_WEB_AGENT.

This profile option belongs to the Workflow (FND_WORKFLOW) profile category in the Application Object Library application.

E

Oracle Workflow Diagnostic Tests

This appendix describes the diagnostic tests that workflow administrators can run to check the setup of Oracle Workflow.

This appendix covers the following topics:

- Oracle Workflow Diagnostic Tests

Oracle Workflow Diagnostic Tests

Oracle Diagnostics provides a mechanism which makes Oracle Applications more supportable and robust by providing predefined tests to check for key setup requirements. Oracle Workflow provides several tests through Oracle Diagnostics that you can use to check the setup of Oracle Workflow and review debugging information.

You can access Oracle Diagnostics through several different user interfaces, including Oracle Applications Manager as well as other administrative consoles. For more information, see: Launching Oracle Diagnostics, *Oracle Applications Supportability Guide*.

The Oracle Workflow tests are available in Oracle Diagnostics under the Application Object Library application.

Workflow Tests

The following tests are available in the Workflow Tests group.

Duplicate User Test

This test checks the Oracle Workflow directory service to verify that there are no roles in the WF_LOCAL_ROLES table with the same internal name, originating system, and originating system ID. This test does not require any input parameters.

If any duplicate roles are found, the test reports the internal name, display name, originating system, and originating system ID of those roles. Use this information to investigate and remove any invalid roles from your directory service. Otherwise, users may encounter errors when accessing their workflow processes and notifications. See:

Setting Up an Oracle Workflow Directory Service, page 2-14.

Notification Preference Validation Test

This test checks the Oracle Workflow directory service to ensure that all roles with a notification preference for receiving e-mail notifications have an e-mail address defined. This test does not require any input parameters.

If any roles with a notification preference for e-mail do not have an e-mail address defined, the test reports the internal name, display name, originating system, originating system ID, and notification preference of those roles. Enter e-mail addresses for these roles in the Oracle Workflow directory service to enable notification mailers to send them e-mail notifications. If any roles should not receive e-mail notifications, change the notification preference for those roles to QUERY ("Do not send me mail"). See: Implementing Notification Mailers, page 2-43, Setting Up an Oracle Workflow Directory Service, page 2-14, Setting Global User Preferences, page 2-9, and Preferences, *Oracle Applications User's Guide*.

Rule Function Validation Test

This test checks the rule functions defined for subscriptions and the generate functions defined for events in the Business Event System. For PL/SQL functions, the test verifies that the package and function exist in the database and are valid. For Java functions, the test verifies that the Java class exists in the classpath. This test does not require any input parameters. See: Event Subscriptions, *Oracle Workflow Developer's Guide* and Events, *Oracle Workflow Developer's Guide*.

If any subscription rule functions or event generate functions are missing or invalid, the test reports the name and problem status for each function. You must make these functions available for the Business Event System to execute the corresponding subscriptions or generate the event data for the corresponding events. For PL/SQL functions, load the appropriate packages and standalone functions to the database, and recompile the schema that owns them. For Java functions, ensure that your classpath includes the JAR files containing the appropriate classes.

GSM Setup Test

This test checks the Generic Service Management (GSM) setup required for Oracle Workflow in Oracle Applications Manager (OAM). Oracle Workflow implements service component containers for its Generic Service Component Framework as GSM service instances. The test verifies that GSM and the Oracle Workflow service instances are enabled. It also checks whether a high number of operating system processes have been created for a service instance in the last few minutes, which may indicate an error. This test does not require any input parameters.

The test reports the status of GSM and of the Oracle Workflow service instances.

- If GSM is disabled, set the Concurrent: GSM Enabled profile option to Y. See: Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide*.

- If an Oracle Workflow service instance is disabled, enable it using OAM. Navigate to the Service Components page in Oracle Workflow Manager, and choose the appropriate service instance link in the Container column. In the Service Instances for Generic Service Component Container page, select the service instance, and select the Edit button. In the Edit page, select the Enabled check box. See: Service Components, page 7-7.

- OAM creates new operating system processes for a service instance each time the service instance is started. If a high number of operating system processes have been created for a particular service instance in the last few minutes, there may be an error that is causing that service instance to become continually deactivated and then restarted by OAM. Use the service instance log to investigate any possible errors. To view the log, navigate to the Service Components page in Oracle Workflow Manager, select a service component belonging to the appropriate service instance, and choose the View Log button. You can also choose the service instance link in the Container column to review the current number of actual and target processes in the Service Instances for Generic Service Component Container page. See: Service Components, page 7-7.

BES Clone Test

This test checks certain standard agents and subscriptions required for internal Business Event System and notification mailer processing to verify that they are enabled and that their definitions include the correct local system. You can use this test after cloning or migrating an Oracle Applications instance to ensure that the local system references have been updated correctly for the new instance. This test does not require any input parameters.

The test reports the name of the local system currently defined in the global workflow preferences, as well as the standard agents and subscriptions to events required for internal processing. For each agent, the report shows the agent name, system name, and status. For each subscription, the report shows the event name, subscription rule function together with the subscribing system, out agent, and status.

- If the local system name is incorrect, use the Workflow Configuration page to update the Local System preference. See: To Set Global Preferences for Oracle Workflow, page 2-9.

- If any agent is assigned to an incorrect system or is disabled, use the Update Agent page to update that agent definition. See: Agents, *Oracle Workflow Developer's Guide*.

- If any subscription has an incorrect subscribing system, has an out agent assigned to an incorrect system, or is disabled, use the Update Event Subscriptions page to update that subscription definition. See: Event Subscriptions, *Oracle Workflow Developer's Guide*.

GSC Control Queue Test

This test verifies that the Workflow control queue, WF_CONTROL, is properly accessible. The Generic Service Component Framework uses this queue to handle control events for containers and service components. This test does not require any input parameters.

The test reports whether it was able to establish a JMS connection to the WF_CONTROL queue and dequeue a message from the queue.

- If the required JMS libraries are not included in the `appsborg2.zip` file referenced in your AF_CLASSPATH, perform the following steps:

  - Verify that JDK (Java Development Kit) 1.3 is installed on the concurrent processing node, patch 2356331 has been applied, and you have performed all post-installation steps described in Oracle*MetaLink* note 384249.1, *Upgrading Java with Oracle E-Business Suite Release 12*. Ensure that the environment variables AF_JRE_TOP, AF_CLASSPATH, and AFJVAPRG are set correctly in `adovars.env`. The AF_CLASSPATH must include the `appsborg2.zip` file.

  - Verify that the `appsborg2.zip` file contains the required JMS classes. Execute the following command:

    ```
    unzip -l $JAVA_TOP/appsborg2.zip | grep -i jms | wc -l
    ```

    This command should return the count 128 if the required JMS classes are present. If it returns 0 (zero), then you need to add the required classes into `appsborg2.zip`. First verify that the `$APPL_TOP/admin/adjborg2.txt` file contains entries for the required classes, `jmscommon.jar` and `aqapi.jar`. Execute the following command:

    ```
    grep jar $APPL_TOP/admin/adjborg2.txt
    ```

    Ensure that the output includes entries for `jmscommon.jar` and `aqapi.jar`. For example:

    ```
    /d01/vis120/oracle/visora/iAS/rdbms/jlib/aqapi.jar
    /d01/vis120/oracle/visora/iAS/rdbms/jlib/jmscommon.jar
    ```

    Then use the AD Administration utility (adadmin) to regenerate product JAR files. Within adadmin, choose the Generate Applications Files menu option and choose the Generate product JAR files task. At the prompt that asks whether you want to force generation of all JAR files, enter Yes. See: Generating Applications Files, *Oracle Applications Maintenance Utilities*.

- If dead subscribers are not being regularly removed from the WF_CONTROL queue, ensure that a recurring concurrent request is scheduled in your Oracle Applications instance for a concurrent program named Workflow Control Queue Cleanup (FNDWFBES_CONTROL_QUEUE_CLEANUP). If you do not have this request scheduled, perform the following steps:

- Terminate the service instances for Oracle Workflow in Oracle Applications Manager. Navigate to the Service Components page in Oracle Workflow Manager, and choose a service instance link in the Container column. In the Service Instances for Generic Service Component Container page, terminate each service instance. See: Service Components, page 7-7.

- Submit a concurrent request for the Workflow Control Queue Cleanup program (FNDWFBES_CONTROL_QUEUE_CLEANUP) for immediate execution. See: Cleaning Up the Workflow Control Queue, page 2-155.

- Submit another request for the Workflow Control Queue Cleanup program to run periodically, so that the WF_CONTROL queue will be cleaned of dead subscribers regularly. The recommended interval for performing cleanup is every six to twelve hours. See: Cleaning Up the Workflow Control Queue, page 2-155.

- After the first concurrent request has completed, restart the service instances for Oracle Workflow. Navigate back to the Service Instances for Generic Service Component Container page in Oracle Applications Manager and restart each service instance. See: Service Components, page 7-7.

  > **Note:** You should not restart the service instances until the initial cleanup performed by the first Workflow Control Queue Cleanup request is finished.

- If the test could not successfully remove a dead subscriber from the WF_CONTROL queue, the queue may need to be dropped and recreated. Contact Oracle Support for instructions.

- If the SYS.DBMS_AQIN package is invalid, Oracle Streams Advanced Queuing may not be properly installed in the database. Contact Oracle Support for instructions.

Workflow Advanced Queue Rule Validation Test

This test checks the standard WF_ERROR and WF_DEFERRED queues to verify that only one subscriber rule is defined for each queue. These queues are reserved for internal Oracle Workflow processing. You should not add any custom subscribers to these queues. This test does not require any input parameters. See: Standard Agents, *Oracle Workflow Developer's Guide*.

If either queue has more than one subscriber rule defined, the test reports the queue name and the invalid rules. Remove any custom subscribers that have been added to the queues. See: Managing Subscribers, *Oracle Streams Advanced Queuing User's Guide and Reference*.

Workflow Agents/AQ Status Test

This test checks the Business Event System agents for Oracle Workflow and Oracle XML Gateway, as well as the queues associated with these agents. The test verifies that the agents are enabled within the Business Event System. It also verifies that the queues and the queue tables in which they reside are valid database objects, and that the queues are enabled for enqueuing and dequeuing within Oracle Streams Advanced Queuing (AQ). This test does not require any input parameters. See: Oracle Streams AQ Administrative Interface, *Oracle Streams Advanced Queuing User's Guide and Reference*.

The test reports the status of the agents, queues, and queue tables, and also shows the subscribers and subscriber rules defined for the queues.

* If an agent is disabled, you can edit the agent definition in the Event Manager Web pages to set its status to Enabled. See: To Create or Update an Agent, *Oracle Workflow Developer's Guide*.

* If a queue is not enabled for enqueuing or dequeuing, you can stop the queue and restart it with these operations enabled, using the DBMS_AQADM.Stop_Queue and DBMS_AQADM.Start_Queue APIs. See: Managing Queues, *Oracle Streams Advanced Queuing User's Guide and Reference*.

* If any queues or queue tables are invalid, contact Oracle Support for instructions.

Workflow Objects Validity Test

This test checks the Oracle Workflow and Oracle XML Gateway database objects to verify that all the objects are valid. This test does not require any input parameters.

If any invalid objects are found, the test reports the name and type of those objects. Recompile those objects and ensure that they are valid.

XML Parser Installation Test

This test checks your Oracle Applications installation to verify that the Oracle XML parser is installed and valid. This test does not require any input parameters.

The test reports the version of the XML parser you have installed and shows the status of the XMLDOM PL/SQL package or xmlparser Java source object in each schema. If any XML parser object is invalid, reinstall the XML parser for that schema.

Mailer Component Test

This test checks your notification mailer service components to verify that at least one notification mailer has been configured with all the parameters needed to run it. This test does not require any input parameters.

If your installation does not include any notification mailer components with a completed configuration, the test reports that no notification mailer is configured.

Navigate to the Service Components page in Oracle Workflow Manager to configure a notification mailer service component. You can either complete the configuration of the seeded Workflow Notification Mailer component, or create a new custom notification mailer component. See: Implementing Notification Mailers, page 2-43 and Notification Mailer Configuration Wizard, page 7-30.

> **Note:** This test does not report whether a configured notification mailer is currently running. You can use the Service Components page in Oracle Workflow Manager to check the current status of your notification mailer service components.

Mailer Component Parameter Test

This test checks your notification mailer service components to validate their configuration parameters. The test checks only those notification mailers for which all the mandatory configuration parameters have been defined. This test does not require any input parameters.

If any notification mailer has invalid configuration parameters, the test reports the errors. Navigate to the Service Components page in Oracle Workflow Manager to edit the configuration of a notification mailer. For some parameters, you may also need to check the corresponding setup outside the notification mailer.

- Ensure that your SMTP server is accessible and able to accept connections, and that the reply-to e-mail address defined for the notification mailer has a valid account on the SMTP server.

- Ensure that your IMAP server is accessible and able to accept connections, that the inbound username defined for the notification mailer has a valid account on the IMAP server, and that you can connect to the IMAP server from an e-mail client using the inbound username and password.

- Ensure that the inbox, processed folder, and discard folder defined for the notification mailer exist within the inbound e-mail account, and that these are three separate folders.

- Ensure that the HTML agent defined for the notification mailer is accessible and that the Web listener is processing requests.

- Ensure that all messages defined as message templates for the notification mailer are loaded in your database, together with the item types to which the messages belong.

See: Implementing Notification Mailers, page 2-43 and Notification Mailer Configuration Wizard, page 7-30.

Event Diagnostic Test

This test reports details about the Business Event System, including the statuses of the local system and agent listeners, details about the definitions of the specified event and any subscriptions to that event, and details about the specified instance of the event if it appears on one of these queues:

- WF_DEFERRED

- WF_JAVA_DEFERRED

- WF_ERROR

- WF_JAVA_ERROR

You can use this information to check the status of a particular event or to investigate errors. Use the Advanced tab within Oracle Diagnostics to enter the event name and event key for the event that you want to review and run the test. The Event Name and Event Key fields are case-sensitive. See: Setting Up the Business Event System, page 2-145.

> **Note:** You can also obtain this information through SQL*Plus using the `wfbesdbg.sql` script. See: Wfbesdbg.sql, page 9-5.

Event Raise Test

This test checks the basic operation of the Business Event System by raising a test event from Java and from PL/SQL and executing synchronous and asynchronous subscriptions to that event. This test does not require any input parameters. See: Setting Up the Business Event System, page 2-145 and Event Subscriptions, *Oracle Workflow Developer's Guide*.

The test reports whether the event was raised successfully and whether the synchronous subscriptions were executed successfully. You can optionally use the test event name and event key provided in the report to run the Event Diagnostic Test in order to verify the execution of the asynchronous subscriptions or perform further diagnosis for any failures. See: Event Diagnostic Test, page E-7.

Mailer Tests

The following test is available in the Mailer Tests group.

Mailer Diagnostic Test

This test reports details about a notification and about the notification mailer that sent the notification. You can use this information to check the status of a particular notification or to investigate errors. Use the Advanced tab within Oracle Diagnostics to

enter the notification ID (NID) of the notification that you want to review and run the test. See: Implementing Notification Mailers, page 2-43.

> **Note:** You can also obtain this information through SQL*Plus using the `wfmlrdbg.sql` script. See: Wfmlrdbg.sql, page 9-10.

Glossary

Access Level

A numeric value ranging from 0 to 1000. Every workflow user operates at a specific access level. The access level defines whether the user can modify certain workflow data. You can only modify data that is protected at a level equal to or higher than your access level.

Activity

A unit of work performed during a business process.

Activity Attribute

A parameter that has been externalized for a function activity that controls how the function activity operates. You define an activity attribute by displaying the activity's Attributes properties page in the Activities window. You assign a value to an activity attribute by displaying the activity node's Attribute Values properties page in the Process window.

Agent

A named point of communication within a system.

Agent Listener

A type of service component that processes event messages on inbound agents.

Attribute

See Activity Attribute, Item Type Attribute, or Message Attribute.

Background Engines

A supplemental Workflow Engine that processes deferred or timed out activities or stuck processes.

Business Event

See Event.

Cost

A relative value that you can assign to a function or notification activity to inform the Workflow Engine how much processing is required to complete the activity. Assign a higher cost to longer running, complex activities. The Workflow Engine can be set to operate with a threshold cost. Any activity with a cost above the Workflow Engine threshold cost gets set to 'DEFERRED' and is not processed. A background engine can be set up to poll for and process deferred activities.

Concurrent Manager

An Oracle Applications component that manages the queuing of requests and the operation of concurrent programs.

Concurrent Process

An instance of running a non-interactive, data-dependent function, simultaneously with online operations. Each time you submit a request, a concurrent manager processes your request, starts a concurrent process, and runs a concurrent program.

Concurrent Program

A concurrent program is an executable file that performs a specific task, such as posting a journal entry or generating a report.

Concurrent Queue

A list of concurrent requests awaiting completion by a concurrent manager. Each concurrent manager has a queue of requests waiting in line to be run. If your system administrator sets up your Oracle Application to have simultaneous queuing, your request can wait to run in more than one queue.

Directory Service

A mapping of Oracle Workflow users and roles to a site's directory repository.

Event

An occurrence in an internet or intranet application or program that might be significant to other objects in a system or to external agents.

Event Activity

A business event modelled as an activity so that it can be included in a workflow process.

Event Data

A set of additional details describing an event. The event data can be structured as an XML document. Together, the event name, event key, and event data fully

communicate what occurred in the event.

Event Key

A string that uniquely identifies an instance of an event. Together, the event name, event key, and event data fully communicate what occurred in the event.

Event Message

A standard Workflow structure for communicating business events, defined by the datatype `WF_EVENT_T`. The event message contains the event data as well as several header properties, including the event name, event key, addressing attributes, and error information.

Event Subscription

A registration indicating that a particular event is significant to a system and specifying the processing to perform when the triggering event occurs. Subscription processing can include calling custom code, sending the event message to a workflow process, or sending the event message to an agent.

External Functions

Programs that are executed outside of the Oracle Database.

Function

A PL/SQL stored procedure that can define business rules, perform automated tasks within an application, or retrieve application information. The stored procedure accepts standard arguments and returns a completion result.

Function Activity

An automated unit of work that is defined by a PL/SQL stored procedure.

Generic Service Component Framework

A facility that helps to simplify and automate the management of background Java services.

Item

A specific process, document, or transaction that is managed by a workflow process.

Item Attribute

See Item Type Attribute.

Item Type

A grouping of all items of a particular category that share the same set of item

attributes. Item type is also used as a high level grouping for processes.

Item Type Attribute

A feature associated with a particular item type, also known as an item attribute. An item type attribute is defined as a variable whose value can be looked up and set by the application that maintains the item. An item type attribute and its value are available to all activities in a process.

Lookup Code

An internal name of a value defined in a lookup type.

Lookup Type

A predefined list of values. Each value in a lookup type has an internal and a display name.

Message

The information that is sent by a notification activity. A message must be defined before it can be associated with a notification activity. A message contains a subject, a priority, a body, and possibly one or more message attributes.

Message Attribute

A variable that you define for a particular message to either provide information or prompt for a response when the message is sent in a notification. You can use a predefine item type attribute as a message attribute. Defined as a 'Send' source, a message attribute gets replaced with a runtime value when the message is sent. Defined as a 'Respond' source, a message attribute prompts a user for a response when the message is sent.

Node

An instance of an activity in a process diagram as shown in the Process window.

Notification

An instance of a message delivered to a user.

Notification Activity

A unit of work that requires human intervention. A notification activity sends a message to a user containing the information necessary to complete the work.

Notification Mailer

A type of service component that sends e-mail notifications to users through a mail application, and processes e-mail responses.

Notification Worklist

A Web page that you can access to query and respond to workflow notifications.

Performer

A user or role assigned to perform a human activity (notification). Notification activities that are included in a process must be assigned to a performer.

Process

A set of activities that need to be performed to accomplish a business goal.

Process Definition

A workflow process as defined in Oracle Workflow Builder, which can be saved as a flat file or in a database.

Process Activity

A process modelled as an activity so that it can be referenced by other processes.

Protection Level

A numeric value ranging from 0 to 1000 that represents who the data is protected from for modification. When workflow data is defined, it can either be set to customizable (1000), meaning anyone can modify it, or it can be assigned a protection level that is equal to the access level of the user defining the data. In the latter case, only users operating at an access level equal to or lower than the data's protection level can modify the data.

Result Code

The internal name of a result value, as defined by the result type.

Result Type

The name of the lookup type that contains an activity's possible result values.

Result Value

The value returned by a completed activity.

Role

One or more users grouped by a common responsibility or position.

Service Component Container

An instance of a service or servlet that manages the running of the individual service components that belong to it. The container monitors the status of its components and

handles control events for itself and for its components.

Service Component

An instance of a Java program which has been defined according to the Generic Service Component Framework standards so that it can be managed through this framework.

Subscription

See Event Subscription.

System

A logically isolated software environment such as a host machine or database instance.

Timeout

The amount of time during which a notification activity must be performed before the Workflow Engine transitions to an error process or an alternate activity if one is defined.

Transition

The relationship that defines the completion of one activity and the activation of another activity within a process. In a process diagram, the arrow drawn between two activities represents a transition.

Workflow Definitions Loader

A concurrent program that lets you upload and download workflow definitions between a flat file and a database.

Workflow Engine

The Oracle Workflow component that implements a workflow process definition. The Workflow Engine manages the state of all activities for an item, automatically executes functions and sends notifications, maintains a history of completed activities, and detects error conditions and starts error processes. The Workflow Engine is implemented in server PL/SQL and activated when a call to an engine API is made.

Index

X