Oracle® E-Business Suite

Setup Guide

Release 12.2

Part No. E22953-10

November 2013

ORACLE

Oracle E-Business Suite Setup Guide, Release 12.2

Part No. E22953-10

Copyright © 1994, 2013, Oracle and/or its affiliates. All rights reserved.

Primary Author: Mildred Wang, Robert Farrington, Melody Yang

Contributing Author: Max Arderius, Santiago Bastidas, George Buzsaki, Anne Carlson, Steve Carter, Steven Chan, Siu Chang, Christina Clark, Lauren Cohn, Ramya Damodoran, Ivo Dujmovic, Elanchelvan Elango, Osama Elkady, Mark Fisher, Clara Jaeckel, Ramkarthik Kalyanasundaram, Takafumi Kamiya, Senthil Madhappan, Teresa Mak, Revathy Narasimhan, Sarita Nori, Mladena Novakovic, Muhannad Obeidat, Gursat Olgun, Richard Ou, Lisa Parekh, Elke Phelps, Traci Short, Jan Smith, Seth Stafford, Susan Stratton, Leslie Studdard, Venkat Vengala, Mark Warren, Aaron Weisberg, Sara Woodhull

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Send Us Your Comments

Preface

1 Fundamental Oracle E-Business Suite Setup Tasks

Oracle E-Business Suite Setup Steps 1-1
Secure Configuration of Oracle E-Business Suite 1-1
Configuring the Login Page for Oracle E-Business Suite 1-2
Personalizing the Oracle E-Business Suite Home Page 1-4
Administering Oracle HTTP Server 1-5
AdminAppServer Utility 1-8
Administering Server Security 1-13
Restricting Access to Responsibilities Based on User's Web Server 1-16
Oracle Application Object Library AOL/J Setup Test Suite 1-17
Configuration for Oracle Application Framework Features 1-19
AutoConfig and Oracle Applications Manager 1-22
User Desktop Setup 1-23

2 System Administration Setup Tasks

Basic Setup Tasks for the Oracle E-Business Suite System Administrator 2-1
Setup Checklist 2-1
Setup Steps 2-2
Setup Tasks for Oracle Applications Manager 2-4

3 Technical Configuration

Introduction 3-1

Using AutoConfig to Manage Oracle E-Business Suite Services 3-5
Using AutoConfig Tools for System Configuration 3-9
Customizing AutoConfig-Managed Configurations 3-16
Patching AutoConfig 3-27
Advanced AutoConfig Features and Additional Utilities 3-29
Using AutoConfig on an Oracle RAC Instance 3-33
Managing Oracle HTTP Server Configurations 3-34
Managing Configuration of Oacore, Oafm, Forms and Forms-c4ws Applications 3-36
Using Profiles in Oracle WebLogic Server 3-46
Changing the Oracle WebLogic Server Administration User Password 3-47
Known Issues 3-48

4 Concurrent Processing

Overview of Concurrent Processing 4-1
Concurrent Requests, Programs, and Processes 4-1
Life cycle of a concurrent request 4-2
Service Management 4-4
Concepts 4-6
Network Failure Recovery 4-6
Failover Sensitive Workshifts 4-7
Overview of Concurrent Programs and Requests 4-7
Controlling Access to Concurrent Programs and Limiting Active Requests for a User 4-8
Organizing Programs into Request Sets 4-11
Defining Request Sets 4-11
Request Sets and Owners 4-23
System Administrator Request Set Privileges 4-24
Request Set Incompatibilities 4-25
Sharing Parameters in a Request Set 4-26
Request Sets Report 4-27
Report Parameters 4-28
Report Headings 4-28
Organizing Programs into Request Groups 4-28
Request Security Groups 4-29
Using Codes with Request Groups 4-29
Customizing the Submit Requests Window 4-30
Customizing the Submit Requests Window using Codes 4-32
Report Group Responsibilities Report 4-33
Report Parameters 4-33
Defining Program Incompatibility Rules 4-34
Incompatible and Run Alone Programs 4-34

Concurrent Conflict Domains ... 4-35
Enforcing Incompatibility Rules ... 4-37
Custom Concurrent Programs ... 4-38
Log and Output Filenames ... 4-39
Oracle Tool Concurrent Programs ... 4-40
Pro*C Concurrent Programs ... 4-41
Host Language Concurrent Programs ... 4-44
Submitting Concurrent Requests (CONCSUB) ... 4-45
Copying and Modifying Program Definitions ... 4-50
Copying and Renaming a concurrent program ... 4-51
Alter Program Priority ... 4-52
Modifying an Incompatible Programs List ... 4-53
Concurrent Program Parameters ... 4-53
Control the Behavior of Request Parameters ... 4-54
Example of modifying a program's parameters ... 4-60
Conflict Domains ... 4-61
Defining Logical Databases ... 4-62
Concurrent Program Details Report ... 4-63
Report Parameters ... 4-63
Report Headings ... 4-63
Concurrent Programs Report ... 4-63
Report Parameters ... 4-64
Report Headings ... 4-64
Request Groups Window ... 4-65
Concurrent Program Executable Window ... 4-67
Concurrent Programs Window ... 4-71
Concurrent Programs Block ... 4-71
Copy to Window ... 4-78
Session Control Window ... 4-78
Incompatible Programs Window ... 4-80
Concurrent Program Parameters Window ... 4-82
Data Groups Window ... 4-86
Data Groups Block ... 4-87
Application-ORACLE ID Pairs Block ... 4-87
Concurrent Conflicts Domains Window ... 4-88
Concurrent Programs HTML UI ... 4-89
Search for Concurrent Programs ... 4-89
Create Concurrent Program ... 4-89
Concurrent Program - Add Parameter ... 4-97
Defining Managers and their Work Shifts in the Oracle Forms UI ... 4-100
Work Shift Definitions ... 4-102

Using Work Shifts to Balance Processing Workload 4-104
Using Time-Based Queues 4-106
Creating Services within Oracle Applications Manager 4-107
Completed Concurrent Requests Report 4-114
Report Parameters 4-114
Report Headings 4-114
Work Shift by Manager Report 4-115
Report Parameters 4-115
Report Headings 4-115
Work Shifts Report 4-115
Report Parameters 4-115
Report Headings 4-115
Specializing Managers to Run Only Certain Programs 4-116
Introduction to Specialization Rules 4-116
Defining Specialization Rules 4-116
Examples - Using Specialization Rules 4-121
Defining Combined Specialization Rules 4-126
Using Combined Rules 4-128
Differences Between Specialization and Combined Rules 4-131
Grouping Programs by Request Type 4-132
Controlling Concurrent Managers 4-134
Controlling the Internal Concurrent Manager from the Operating System 4-138
Overview of Parallel Concurrent Processing 4-143
What is Parallel Concurrent Processing? 4-143
Parallel Concurrent Processing Environments 4-144
How Parallel Concurrent Processing Works 4-145
Managing Parallel Concurrent Processing 4-147
Defining Concurrent Managers 4-148
Administering Concurrent Managers 4-149
Administer Concurrent Managers Window 4-151
Administer Concurrent Managers Block 4-151
The actions you can choose for controlling a manager are: 4-152
Reviewing a Specific Manager 4-154
Concurrent Processes Window 4-155
Concurrent Requests Window 4-158
Request Diagnostics Window 4-159
Concurrent Managers Window 4-160
Concurrent Managers Block 4-160
Work Shifts Window 4-163
Specialization Rules Window 4-165
Work Shifts Window 4-167

Combined Specialization Rules Window 4-169
Combined Specialization Rules Block 4-170
Specialization Rules Block 4-170
Concurrent Request Types Window 4-172
Concurrent Request Types Block 4-172
Viewer Options Window 4-173
Viewer Options Block 4-174
Nodes Window 4-175
Nodes Block 4-175
Setting Up, Starting, and Shutting Down Concurrent Managers 4-176
Setting Up Concurrent Managers 4-176
Starting the Concurrent Managers 4-179
Restarting the Concurrent Managers 4-181
Shutting Down the Concurrent Manager Service (Windows) 4-182
Removing the Concurrent Manager Service (Windows) 4-183
Starting and Stopping the Concurrent Manager with the Concurrent Manager Operator User 4-183
File Conventions 4-184
Directory Privileges 4-185
Printing 4-187
Managing Concurrent Processing with Oracle Applications Manager 4-194
Service Instances 4-195
Status Overview 4-201
OAM Generic Collection Service 4-206
Concurrent Processing Charts and Reports 4-207
Multilingual Support for Concurrent Requests 4-217
Request Submission 4-217
Runtime Behavior 4-217
MLS Functions 4-218
Multiple Organizations Reporting 4-218
The Output Post Processor 4-219
Delivery Options for Concurrent Request Output 4-220
Managing Delivery Options 4-223
Reviewing Requests, Request Log Files, and Report Output Files 4-224
How To View Request Status and Output 4-224
Setting End User Report and Log File Access Privileges 4-225
Defining the Reports Viewer 4-226
Viewing HTML Report Output 4-226
Online Report Review using Other Applications 4-226
Changing the Status of Concurrent Requests and Request Sets 4-227
Changing a Request's Phase and Status 4-227

Changing the Priority of a Pending or Inactive request 4-228
Managing Request Sets 4-229
Managing Concurrent Processing Files and Tables 4-229
Concurrent Processing User Profile Settings 4-231
Purge Concurrent Request and/or Manager Data Program 4-233

5 User Profiles and Profile Options in Oracle Application Object Library

Overview of Setting User Profiles 5-1
Setting Profile Options 5-4
Using Profile Options in Other Oracle E-Business Suite Features 5-7
Examples of User Profile Options 5-7
Profile Categories 5-8
User Profile Option Values Report 5-9
Report Parameters 5-9
Report Headings 5-9
Profile Options in Oracle Application Object Library 5-9

6 Setting Up Printers

Printers and Printing 6-1
Overview 6-1
Printer Types, Print Styles, and Printer Drivers 6-2
Sequence of Printing Events 6-4
Setting Character-Mode vs. Bitmap Printing 6-5
Setting Up Your Printers 6-8
Printing Setup Interrelationships 6-9
Printer Setup Information Is Cached On Demand 6-9
Printer Setup with Pasta 6-10
Overview 6-10
Setup for Basic Printing with Pasta 6-10
Defining Configuration Files for Specific Printers 6-11
Using a Different Configuration File as the Default 6-11
Modify an Existing Printer Type to Use Pasta 6-12
Add a New Printer Type to Use Pasta 6-13
Setting Margins 6-13
Printing a Report Generated Using the noprint Option 6-14
Generating Other Formats Using the Preprocessing Option 6-15
Font Source 6-16
Language-Specific Font Support 6-17
Configuration File Options 6-18
Command Line Parameters 6-24

Using PrintForms 6-25
Create the PrintForm 6-26
Common UNIX Printing System (CUPS) Integration 6-28
Customizing Printing Support in Oracle E-Business Suite 6-31
Hierarchy of Printer and Print Style Assignments 6-39
Hierarchy of Printer Assignments 6-40
Hierarchy of Print Style Assignments 6-41
System Administrator Printer and Print Style Settings 6-42
End User Printer and Print Style Settings 6-43
Postscript Printing in UNIX 6-43
Printer Types Window 6-47
Printer Types Block 6-48
Printer Drivers Block 6-48
Printers Window 6-49
Printers Block 6-49
Print Styles Window 6-50
Print Styles Block 6-51
Layout Block 6-52
Printer Drivers Window 6-53
Printer Drivers Window Fields 6-54
Driver Method Region 6-54
Driver Method Parameters Region 6-55

7 Setting Up Oracle E-Business Suite Help

Setting Oracle E-Business Suite Help Profile Options 7-1
Customizing Oracle E-Business Suite Help 7-1
Downloading and Uploading Help Files 7-2
Linking Help Files 7-5
Updating the Search Index 7-8
Customizing Help Navigation Trees 7-8
The Help Builder User Interface 7-9
Customizing Help in a Global Environment 7-16
Using Oracle Tutor 7-17

8 Time Zone Support

User-Preferred Time Zones 8-1
Time Zone Concepts 8-2
User-Preferred Time Zones and Products 8-3
Upgrade Considerations 8-16
Implementation Details 8-17

9 Managing Globalization

Overview of Globalization Support ... 9-1
Language Values for User Sessions ... 9-1
Language Values for User Sessions using AppsLocalLogin.jsp ... 9-1
Language Value from Login External to Oracle E-Business Suite ... 9-3
Language Values for Oracle Workflow Notifications ... 9-3
Date Formats in NLS Implementations ... 9-4
Support for Calendars ... 9-4
Numeric Format Support ... 9-5
Lightweight MLS ... 9-6
Multilingual External Documents ... 9-6
Translations Window ... 9-8
Currencies Window ... 9-9
Languages Window ... 9-9
Languages Record ... 9-9
Natural Languages Window ... 9-10
Natural Languages Record ... 9-10
Territories Window ... 9-11
Territories Block ... 9-11

10 Defining Document Sequences

What is a Document Sequence? ... 10-1
Defining a Document Sequence ... 10-1
Defining Document Categories ... 10-4
Assigning a Document Sequence ... 10-5
Document Numbering vs. Document Entry ... 10-7
Document Sequences Window ... 10-7
Document Sequences Block ... 10-8
Document Categories Window ... 10-10
Document Categories Block ... 10-10
Sequence Assignments Window ... 10-12
Sequence Assignments Block ... 10-12
Assignment Region ... 10-13
Document Flexfield ... 10-14

11 Applications DBA System Configuration Tools

About System Configuration ... 11-1
AD Splicer ... 11-3
File Character Set Converter ... 11-6

12 Developer Tools

Developer Tools 12-1
Form Personalization 12-1
Work Directory 12-2
Web Enabled PL/SQL Window 12-3
Portlet Generator 12-4
Integration with Application Development Framework 12-7

13 Using Loaders

Generic Loader 13-1
Overview 13-1
FNDLOAD Executable 13-3
Configuration File 13-5
Data File 13-9
Oracle Application Object Library Configuration Files 13-9
Attachments Setup Data Configuration File 13-10
Concurrent Program Configuration File 13-12
Flexfields Setup Data Configuration File 13-15
Flexfield Value Sets 13-15
Descriptive Flexfields 13-15
Key Flexfields 13-16
Folders Configuration File 13-16
Lookups Configuration File 13-19
Messages Configuration File 13-20
Profile Options and Profile Values Configuration File 13-20
Request Groups Configuration File 13-21
Security Information Configuration File 13-22
Message Dictionary Generator 13-23
Message Repositories 13-23
Usage 13-24
Generic File Manager Access Utility (FNDGFU) 13-25
Usage 13-25
Example of FNDGFU Upload 13-26
Purging Generic File Manager Data 13-28
Purge Obsolete Generic File Manager Data 13-28
Program Parameters 13-28

14 Oracle Applications Tablespace Model

Introduction to Oracle Applications Tablespace Model 14-1

Advantages of Migrating to OATM....14-1
OATM Tablespaces....14-5
Customizations and Extensions....14-8
Introduction to the Oracle Applications Tablespace Migration Utility....14-9
Planning for Migration....14-9
Setting Up the Tablespace Migration Utility....14-10
Phase 1: Preparatory Steps....14-13
Phase 2: Migration Steps....14-18
Phase 3: Post Migration Steps....14-21

15 Growing a System with Oracle Real Application Clusters

Introduction to Oracle Real Application Clusters....15-1
Prerequisites....15-1
Migrating to Oracle RAC....15-2
Establishing the Oracle E-Business Suite Environment for Oracle RAC....15-2
Configuring Parallel Concurrent Processing with Oracle RAC....15-2

16 Integration of Oracle E-Business Suite with Other Products

Integration of Oracle E-Business Suite with Other Products....16-1

17 Cloning Oracle E-Business Suite with Rapid Clone

Introduction to Cloning....17-1
Prerequisite Tasks....17-3
Standard Cloning Tasks....17-6
Finishing Tasks....17-14
Advanced Cloning Options....17-15
Automating Rapid Clone....17-24
Other Cloning Procedures....17-26

Index

Send Us Your Comments

Oracle E-Business Suite Setup Guide, Release 12.2

Part No. E22953-10

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 Oracle E-Business Suite Release Online Documentation CD available on My Oracle Support 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 12.2 of the *Oracle E-Business Suite Setup Guide.*

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

- The principles and customary practices of your business area.
- Computer desktop application usage and terminology.

If you have never used Oracle E-Business Suite, we suggest you attend one or more of the Oracle E-Business Suite training classes available through Oracle University.

> **Note:** Some of the screenshots used in this guide depict Oracle's default corporate browser Look-and-Feel (LAF), while others depict an alternative LAF. Although the colors and interface elements of these images may vary, the underlying functionality they illustrate remains the same, regardless of the LAF that you have implemented.

See Related Information Sources on page xvi for more Oracle E-Business Suite product information.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit

http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Structure

1 Fundamental Oracle E-Business Suite Setup Tasks
2 System Administration Setup Tasks
3 Technical Configuration
4 Concurrent Processing
5 User Profiles and Profile Options in Oracle Application Object Library
6 Setting Up Printers
7 Setting Up Oracle E-Business Suite Help
8 Time Zone Support
9 Managing Globalization
10 Defining Document Sequences
11 Applications DBA System Configuration Tools
12 Developer Tools
13 Using Loaders
14 Oracle Applications Tablespace Model
15 Growing a System with Oracle Real Application Clusters
16 Integration of Oracle E-Business Suite with Other Products
17 Cloning Oracle E-Business Suite with Rapid Clone

Related Information Sources

This book is included in the Oracle E-Business Suite Documentation Library, which is supplied in the Release 12.2 Media Pack. If this guide refers you to other Oracle E-Business Suite documentation, use only the latest Release 12.2 versions of those guides.

Online Documentation

All Oracle E-Business Suite documentation is available online (HTML or PDF).

- **Online Help** - Online help patches (HTML) are available on My Oracle Support.
- **PDF Documentation** - See the Oracle E-Business Suite Documentation Library for current PDF documentation for your product with each release.
- **Release Notes** - For information about changes in this release, including new features, known issues, and other details, see the release notes for the relevant product, available on My Oracle Support.
- **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 E-Business Suite product. This information helps you convert data from your existing applications and integrate Oracle E-Business Suite data with non-Oracle applications, and write custom reports for Oracle E-Business Suite products. The Oracle eTRM is available on My Oracle Support.

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 E-Business Suite Concepts

This book is intended for all those planning to deploy Oracle E-Business Suite Release 12.2, or contemplating significant changes to a configuration. After describing the Oracle E-Business Suite 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 E-Business Suite Installation Guide: Using Rapid Install

This book is intended for use by anyone who is responsible for installing or upgrading Oracle E-Business Suite. It provides instructions for running Rapid Install either to carry out a fresh installation of Oracle E-Business Suite Release 12.2, or as part of an upgrade to Release 12.2.

Oracle E-Business Suite Maintenance Guide

This guide explains how to patch an Oracle E-Business Suite system, describing the adop patching utility and providing guidelines and tips for performing typical patching operations. It also describes maintenance strategies and tools that can help keep a system running smoothly.

Oracle E-Business Suite Security Guide

This guide contains information on a comprehensive range of security-related topics, including access control, user management, function security, data security, and auditing. It also describes how Oracle E-Business Suite can be integrated into a single sign-on environment.

Oracle Alert User's Guide

This guide explains how to define periodic and event alerts to monitor the status of your Oracle E-Business Suite data.

Oracle Application Framework Developer's Guide

This guide contains the coding standards followed by the Oracle E-Business Suite development staff to produce applications built with Oracle Application Framework. This guide is available in PDF format on My Oracle Support 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 Diagnostics Framework User's Guide

This manual contains information on implementing and administering diagnostics tests for Oracle E-Business Suite using the Oracle Diagnostics Framework.

Oracle E-Business Suite CRM System Administrator's Guide

This manual describes how to implement the CRM Technology Foundation (JTT) and use its System Administrator Console.

Oracle E-Business Suite Developer's Guide

This guide contains the coding standards followed by the Oracle E-Business Suite development staff. It describes the Oracle Application Object Library components needed to implement the Oracle E-Business Suite user interface described in the *Oracle E-Business Suite User Interface Standards for Forms-Based Products*. It provides information to help you build your custom Oracle Forms Developer forms so that they integrate with Oracle E-Business Suite. In addition, this guide has information for customizations in features such as concurrent programs, flexfields, messages, and logging.

Oracle E-Business Suite Flexfields Guide

This guide provides flexfields planning, setup, and reference information for the Oracle E-Business Suite implementation team, as well as for users responsible for the ongoing maintenance of Oracle E-Business Suite product data. This guide also provides information on creating custom reports on flexfields data.

Oracle E-Business Suite User's Guide

This guide explains how to navigate, enter and query data, and run concurrent requests using the user interface (UI) of Oracle E-Business Suite. This guide also includes information on setting user profiles and customizing the UI.

Oracle E-Business Suite User Interface Standards for Forms-Based Products

This guide contains the user interface (UI) standards followed by the Oracle E-Business Suite development staff. It describes the UI for the Oracle E-Business Suite products and how to apply this UI to the design of an application built by using Oracle Forms.

Oracle iSetup User's Guide

This guide describes how to use Oracle iSetup to migrate data between different instances of the Oracle E-Business Suite and generate reports. It also includes configuration information, instance mapping, and seeded templates used for data migration.

Oracle Web Applications Desktop Integrator Implementation and Administration Guide

Oracle Web Applications Desktop Integrator brings Oracle E-Business Suite functionality to a spreadsheet, where familiar data entry and modeling techniques can be used to complete Oracle E-Business Suite tasks. You can create formatted spreadsheets on your desktop that allow you to download, view, edit, and create Oracle E-Business Suite data, which you can then upload. This guide describes how to implement Oracle Web Applications Desktop Integrator and how to define mappings, layouts, style sheets, and other setup options.

Oracle Workflow Administrator's Guide

This guide explains how to complete the setup steps necessary for any product that

includes workflow-enabled processes. It also describes how to manage workflow processes and business events using Oracle Applications Manager, how to monitor the progress of runtime workflow processes, and how to administer notifications sent to workflow users.

Oracle Workflow Developer's Guide

This guide explains how to define new workflow business processes and customize existing Oracle E-Business Suite-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 Workflow Client Installation Guide

This guide describes how to install the Oracle Workflow Builder and Oracle XML Gateway Message Designer client components for Oracle E-Business Suite.

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. It also explains how to use Collaboration History that records all business transactions and messages exchanged with trading partners.

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

Oracle XML Publisher Report Designer's Guide

Oracle XML Publisher is a template-based reporting solution that merges XML data with templates in RTF or PDF format to produce a variety of outputs to meet a variety of business needs. Using Microsoft Word or Adobe Acrobat as the design tool, you can create pixel-perfect reports from the Oracle E-Business Suite. Use this guide to design your report layouts.

This guide is available through the Oracle E-Business Suite online help.

Oracle XML Publisher Administration and Developer's Guide

Oracle XML Publisher is a template-based reporting solution that merges XML data with templates in RTF or PDF format to produce a variety of outputs to meet a variety of business needs. Outputs include: PDF, HTML, Excel, RTF, and eText (for EDI and EFT transactions). Oracle XML Publisher can be used to generate reports based on existing Oracle E-Business Suite report data, or you can use Oracle XML Publisher's data extraction engine to build your own queries. Oracle XML Publisher also provides a

robust set of APIs to manage delivery of your reports via e-mail, fax, secure FTP, printer, WebDav, and more. This guide describes how to set up and administer Oracle XML Publisher as well as how to use the Application Programming Interface to build custom solutions.

This guide is available through the Oracle E-Business Suite online help.

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.

You can navigate to the Oracle Integration Repository through Oracle E-Business Suite Integrated SOA Gateway.

Do Not Use Database Tools to Modify Oracle E-Business Suite Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle E-Business Suite 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 E-Business Suite data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle E-Business Suite tables are interrelated, any change you make using an Oracle E-Business Suite form can update many tables at once. But when you modify Oracle E-Business Suite data using anything other than Oracle E-Business Suite, 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 E-Business Suite.

When you use Oracle E-Business Suite to modify your data, Oracle E-Business Suite automatically checks that your changes are valid. Oracle E-Business Suite 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

Fundamental Oracle E-Business Suite Setup Tasks

Oracle E-Business Suite Setup Steps

Oracle E-Business Suite is installed using the Rapid Install tool. Refer to *Oracle E-Business Suite Installation Guide: Using Rapid Install* for detailed information on running Rapid Install.

The following steps are carried out as part of, or after, Oracle E-Business Suite installation:

1. Run Rapid Install
2. Test Web listener virtual directories
3. Test Oracle HTTP Server configuration
4. Create DBC files
5. Test Java Servlet setup
6. Set Web Server profile options

The above is *not* an inclusive list of the installation and post-installation tasks that may be needed on a particular system. Additional Oracle E-Business Suite features and options may need to be set up. These are detailed in the following sections.

Secure Configuration of Oracle E-Business Suite

Refer to My Oracle Support Knowledge Document 403537.1, "Secure Configuration Guide for Oracle E-Business Suite Release 12", for important information on securing your Oracle E-Business Suite installation.

Configuring the Login Page for Oracle E-Business Suite

Oracle E-Business Suite uses a configurable login page, which can be tailored to suit the needs of different organizations.

Oracle E-Business Suite Login page

Users log in to Oracle E-Business Suite using a client web browser. From the Oracle E-Business Suite Login page, users access the Oracle E-Business Suite Home Page, which provides a single point of access to HTML-based applications and forms-based applications. Users access the Oracle E-Business Suite Login page from the following URL:

```
http://<server:port>/OA_HTML/AppsLogin
```

For example,

```
http://oraapps1.oracle.com:8000/OA_HTML/AppsLogin
```

From this URL, you will be redirected to the central login page, "AppsLocalLogin.jsp".

Central Login Page

The following features are displayed in the default login page: Username field, Password field, Login button, and the Language Picker (if more than one language is installed).

The ICX login page (http://server:port/OA_HTML/US/ICXINDEX.htm) redirects the user to the central login page, "AppsLocalLogin.jsp". If, in a previous release, you customized the ICX login page previously with a custom logo, you should make a copy of the new ICX login page and replace the existing image with your custom image in the copied file. The location for the company logo is $OA_MEDIA/FNDSSOCORP.gif. Ensure that the image is appropriately sized. Also, you should change the text of the

message 'FND_ORACLE_LOGO' in Message Dictionary to the appropriate text. The following login URL is supported, but no new features are being added to it:

```
http://server:port/OA_HTML/jtflogin.jsp
```

If the Oracle E-Business Suite instance is Single Sign-On enabled, the servlet directs the user to the Single Sign-On login page.

Passwords

Note that in previous releases of Oracle E-Business Suite, user passwords were treated as case insensitive. Now, Oracle E-Business Suite user passwords can optionally be treated as case sensitive, depending on the mode you choose. Case-sensitivity in passwords is controlled by the profile option Password Case Option.

Language

The current browser language, if it exists in the applications database also, is used for the login page. The user can choose a different language on the login page (if the Language Picker is enabled) and refresh the page to have it appear in that language.

If the current browser language does not exist in the Oracle E-Business Suite installation, the language set in the site-level setting of the ICX_LANGUAGE profile option is used to render the login page.

A user can override the value of the ICX_LANGUAGE profile option for a given session only.

Oracle Applications Manager

Oracle Applications Manager uses this central login page as well to authenticate users.

Customizing the Oracle E-Business Suite Login page

The values for the Username Hint, Password Hint, Forgot Password URL text, Register URL text, and the Corporate Policy Message, as well as the Copyright text, are stored in Message Dictionary. You can update these messages using the Messages form or HTML page.

The table below lists the message names and their default values.

Login Page Message Names and Default Values

Message Name	Default value
FND_SSO_HINT_USERNAME	(example: michael.james.smith)
FND_SSO_HINT_PASSWORD	(example: 4u99v23)

Message Name	Default value
FND_SSO_FORGOT_PASSWORD	Forgot your password?
FND_SSO_REGISTER_HERE	Register here
FND_SSO_SARBANES_OXLEY_TEXT	Corporate Policy Message
FND_SSO_COPYRIGHT_TEXT	Copyright (c) 1998, 2013, Oracle and/or its affiliates. All rights reserved.

Personalizing the Oracle E-Business Suite Home Page

The Oracle E-Business Home Page can be personalized to display the Worklist and to display the Applications Navigator in "Tree" or "Flat" mode. By default, the Applications Navigator is shown in Flat mode.

Refer to the *Oracle Application Framework Personalization Guide* for more information on personalizing Oracle Application Framework pages. Ensure that you have the relevant profile options set appropriately.

Follow these steps to use the Tree mode:

1. Log in to Oracle E-Business Suite with the system administrator responsibility.
2. From the Oracle E-Business Suite Home page, select **Personalize Page**.
3. Select **Apply** in the "Choose Personalization Context" page.
4. Under "Personalization Structure", expand the following nodes: 'Table Layout: (topTableLayoutContainer)' > 'Row Layout: (tableLayoutRow)' > 'Cell Format: (worklistResponsibilityLeftCell)' > 'Table Layout: Home Contenttable'.
5. Under the node 'Table Layout: Home Contenttable' > Row Layout: (responsibilityRow), click the Personalize icon.
6. Change the value for the 'Rendered' property for the column 'Function: Applications Home Page' to 'false'. Select **Apply.**
7. Click on the Personalize icon for 'Table Layout: Home Contenttable' > 'Row Layout: (appsNavTreeRow)'.
8. Change the value for the 'Rendered' property for the column 'Function: Applications Home Page' to 'true'. Select **Apply.**

9. Select **Return to Application.**

Administering Oracle HTTP Server

Oracle HTTP Server Powered by Apache provides the communication services within Oracle Application Server. This facilitates deployment of HTML-based applications within a multi-tiered computing environment.

Oracle HTTP Server

All incoming client requests to Oracle Internet Application Server (AS) are handled by the Communication Services component of AS. The Oracle HTTP Server accepts and processes these requests. The Apache technology adopted by Oracle HTTP Server provides an extremely stable, scalable, and extensible platform on which to deploy web-based applications. The modular design of the Apache server allows for extension of the capabilities of the Oracle HTTP Server. In addition to the standard Apache modules (often referred to as modules, or simply mods), a number of Oracle specific modules are provided along with an extension to the functionality of several standard modules.

These modules include:

- **mod_ossl** - This module provides secure listener communications using an Oracle provided encryption mechanism using 128-bit *Secure Sockets Layer* (SSL). The mod_ossl module replaces mod_ssl. In contrast to the OpenSSL module, mod_ossl is based on the Oracle implementation of SSL, which supports SSL version 3 and uses the Oracle Wallet Manager for Certificate Management. An example for the location of the Apache HTTP Server SSL configuration file, *ssl.conf*, is `/u02/FMW_Home/webtier/instances/EBS_web_VIS1012_al1/config/OHS/EBS_web_component`.

- **mod_perl** - This module forwards Perl requests to the Perl Interpreter. The Perl Interpreter is embedded within the Oracle HTTP Server, removing the necessity to spawn an external interpreter as well as providing a caching mechanism such that modules and scripts need only be loaded or compiled once. Oracle E-Business Suite does not currently utilize mod_perl.

  For Oracle E-Business Suite Release 12.2., the Oracle HTTP Server is powered by version 2.2 of Apache. A number of books have been published describing the operation of the Apache server. To further add to your knowledge of the Apache server, you may wish to consult one of these.

  > **Note:** Refer to the Oracle HTTP Server documentation for a more detailed description of the operation and configuration of the Oracle HTTP Server. The information in this section is supplementary to that provided in the Oracle HTTP Server books.

Oracle E-Business Suite Installation Guide: Using Rapid Install should be consulted for additional information on directory structures and file locations referred to in this guide.

Configuration Files

Apache is configured through directives contained in one or more configuration files. The directives necessary for operating Apache within the Oracle environment will be entered into the configuration files during the install process. It should not be necessary to modify these files unless the system is being re-configured.

> **Warning:** An invalid directive entered into a configuration file will prevent Apache from starting. An incorrect definition provided to a directive may cause Apache to behave in an unintended fashion.

Location

The Apache configuration files are installed as part of the Oracle E-Business Suite Rapid Install process. For example, the files could be placed in `/u02/FMW_Home/webtier/instances/EBS_web_VIS1012_al1/config/OHS/EBS_web_component` (on UNIX).

Secure Sockets Layer Configuration

Secure Sockets Layer (SSL) allows the Apache listener to encrypt HTML pages and transmit them on the network using the HTTPS protocol for secure transmissions.

SSL uses an encrypting method called public key cryptography, where the server provides the client with a public key for encrypting information. The server's private key is required to decrypt this information. The client uses the public key to encrypt and send information to the server, including its own key which identifies it to the server.

In order for the Oracle HTTP Server, powered by Apache, to function in secure mode it is also necessary to utilize certificates which validate the server's identity. These certificates are used to ensure that the owner of a public key is who they say they are. Typically you will want to use a private key with an officially signed certificate, validated by a Certificate Authority (CA). The CA validates the company details, sets expiration dates on the certificates and may place policies on what information is contained within the certificate. A number of CAs exist, and include such authorities as Verisign, RSA, and GTE CyberTrust.

To obtain a CA approved certificate it is necessary to generate a certificate request, which includes details of the organization applying as well as the public key to be distributed, this is then sent to the CA, validated and returned. Authorities may also require proof of ownership of the company applying for the certificate, as well as proof of ownership of the domain name specified in the certificate request.

It is possible to generate a self-signed certificate that can be used to test SSL operation. Most browsers are configured to accept certificates from a number of recognized authorities. Receiving a certificate from an organization other than one of these will generate a warning, and the user will be prompted to accept or reject this certificate. A self-signed certificate will generate this warning.

For more information on SSL configuration, refer to My Oracle Support Knowledge Document 1367293.1, *Enabling SSL in Oracle E-Business Suite Release 12.2*.

Configure Apache for SSL Connections using AutoConfig

Ensure that the following parameters are correctly set in your Oracle E-Business Suite applications context file:

Variable	Non-SSL Value	SSL Value
s_url_protocol	http	https
s_local_url_protocol	http	https
s_webentryurlprotocol	http	https
s_active_webport	Same as s_webport	Same as s_webssl_port
s_help_web_agent	URL constructed with http protocol and s_webport	URL constructed with http protocol and s_webport (same as non-SSL)
s_login_page	URL constructed with http protocol and s_webport	URL constructed with https protocol and s_webssl_port

Run AutoConfig, either through the Oracle Applications Manager interface, or by using the `adautocfg.sh` script in the application tier $ADMIN_SCRIPTS_HOME directory.

For more details, see the chapter Technical Configuration, page 3-1 and My Oracle Support Knowledge Document 1367293.1, *Enabling SSL in Oracle E-Business Suite Release 12.2*.

Test Startup of Apache and Oracle E-Business Suite Sign-On

Restart the application tier Apache services using the `adapcctl.sh` script in the $ADMIN_SCRIPTS_HOME directory. Ensure that Apache can startup successfully, and that you successfully get an SSL connection to the default Apache banner screen using https:/<host.domain>:<SSL_port>, where <host.domain> is the fully qualified name of the machine running Apache, and <SSL_port> is the SSL port number defined in ssl.conf.

Once you have signed on to Oracle E-Business Suite, select a responsibility and process that will launch a Forms-based application. For example, System Administrator responsibility and the Define User process.

> **Note:** For further details of using SSL, see My Oracle Support Knowledge Document 376700.1, *Enabling SSL with Oracle Application Server 10g and E-Business Suite Release 12*, and Knowledge Document 376694.1, *Using the Oracle Wallet Manager Command Line Interface with Oracle E-Business Suite Release 12*.

AdminAppServer Utility

Using the AdminAppServer Utility

The Java script AdminAppServer is used to create .dbc files and to enable or disable application server security.

Prior to running AdminAppServer you must ensure that:

- JDBC classes are in the classpath
- $JAVA_TOP is in the classpath

For UNIX platforms, the script is run as:

```
java oracle.apps.fnd.security.AdminAppServer [parameters]
```

For Windows platforms, the script is run as:

```
jre -classpath %CLASSPATH% \
   oracle.apps.fnd.security.AdminAppServer [parameters]
```

The following commands are supported:

- **ADD** - create a new .dbc file
- **UPDATE** - update an existing .dbc file
- **DELETE** - delete an existing .dbc file
- **STATUS** - check the serverID status for a database
- **AUTHENTICATION** - toggle authentication mode

Additional parameters depend on the operation. These include:

- **DBC** - The .dbc file to modified, or used to connect to the database. Used with UPDATE, DELETE, STATUS, AND AUTHENTICATION.
- **SECURE_PATH** - Used with ADD. This parameter specifies in which directory the

.dbc file should be created, and defaults to the current directory if not provided. This parameter should always point to $FND_TOP/secure.

- **DB_HOST** - Required. The host machine of database.
- **DB_PORT** - Required. The port of database. The default is 1521.
- **DB_NAME** - For thin drivers. The database SID.
- **APPS_JDBC_DRIVER_TYPE** - THICK or THIN. This parameter must be set to THIN.
- **GUEST_USER_PWD** - Any valid applications user. This parameter defaults to the value of GUEST_USER_PWD profile if not provided. If passed with no arguments to an UPDATE call, it will refresh with the value from database.
- **GWYUID** - For thick drivers.
- **FNDNAM** - For thick drivers.
- **TWO_TASK** - For thick drivers. Name of database.
- **WALLET_PWD** - Used with the TCF Socket Server in SSL mode.
- **SERVER_ADDRESS** - Used with authentication.
- **SERVER_DESCRIPTION** - Used with authentication.
- **FND_MAX_JDBC_CONNECTIONS** - The maximum number of open connections in the JDBC connection cache. This number is dependent on the amount of memory available, number of processes specified in the init.ora file of the database and the per-processor file descriptor limit.
- **FND_IN_USE_CONNECTION_TIMEOUT** - The maximum number of seconds a connection can be in use. In order to avoid connections being locked up for too long, the connection cache uses this parameter to forcibly close connections that have been locked for longer than this specified limit. If this parameter is unspecified, connections in use will not be cleaned up. This should be set to a number larger than the time taken to complete the largest transaction.
- **FND_UNUSED_CONNECTION_TIMEOUT** - The maximum number of seconds an unused connection can remain in the cache. The connection cache will close and remove from the cache any connection that has been idle for longer than this specified limit.

Important: The following parameters are required:

APPS_JDBC_DRIVER_TYPE (must be set to THIN), DB_HOST, and DB_PORT.

Administering .dbc Files

The .dbc file is contained on the web/applications server and holds information used by the database for authentication. The web/application server passes the information from the .dbc file, as well as login information, to the database server to authenticate the user. The authentication process is handled by the standard applications security feature.

The .dbc files required by the application server security system are not part of the delivered product and must be created after installation.

The Java utility AdminAppServer is used to create the .dbc files.

Prior to running AdminAppServer you must ensure that:

- JDBC classes are in the CLASSPATH
- $JAVA_TOP is in the classpath

Creating a .dbc file

Use the AdminAppServer utility to create a .dbc file for the application server to access the database server. In addition to creating the .dbc file this utility registers the application server with the database for the Applications Server Security feature.

To access additional database servers from the same application server, you must rerun the AdminAppServer utility for each additional database. You must run the AdminAppServer utility each time you create a .dbc file, and each .dbc file only allows access to one database.

To create a .dbc file for an application server:

1. You must set the username/password value for the GUEST_USER_PWD parameter. Create a valid username ("guest", for example) and password in Oracle E-Business Suite. Then use the username/password combination as the value for GUEST_USER_PWD. The syntax is illustrated in the following example:

   ```
   GUEST_USER_PWD=guest/password
   ```

   Oracle recommends that you do not assign any responsibilities for this user.

2. From the command line, enter the appropriate command for your platform:

   For UNIX platforms:

   ```
   java oracle.apps.fnd.security.AdminAppServer \
    ADD DB_HOST=database_host_name DB_PORT=database_port \
    DB_NAME=database_sid \
    [env_name=env_value] SECURE_PATH=$FND_TOP/secure
   ```

For Windows platforms:

```
jre -classpath %CLASSPATH% \
 oracle.apps.fnd.security.AdminAppServer \
 ADD DB_HOST=database_host_name DB_PORT=database_port \
 DB_NAME=database_sid \
 [env_name=env_value] SECURE_PATH=$FND_TOP/secure
```

You will be prompted for the APPS username and password using standard I/O.

GWYUID, FNDNAM, and GUEST_USER_PWD will be defaulted if not provided explicitly. dbc files should be located in $FND_TOP/secure, so SECURE_PATH should always be set to that, or this should be run directly out of of the $FND_TOP/secure area.

Updating a .dbc file (or Server ID)

When updating the .dbc file you can change as many parameters as you want, including the server ID, but you must enter at least one. Settings that you do not update retain their value.

To update a .dbc file or server ID:

Enter from the command line:

UNIX

```
java oracle.apps.fnd.security.AdminAppServer \
UPDATE DBC=$FND_TOP/secure/file.dbc APPL_SERVER_ID \
  [env_name=env_value]
```

Windows

```
jre -classpath %CLASSPATH% \
oracle.apps.fnd.security.AdminAppServer \
UPDATE DBC=$FND_TOP/secure/file.dbc APPL_SERVER_ID \
[env_name=env_value]
```

You will be prompted for the APPS username and password using standard I/O.

If APPL_SERVER_ID is not passed, AdminAppServer will attempt to synchronize the current server ID value in the .dbc file and the database, generating a new one if neither contains a value at all. Passing in APPL_SERVER_ID by itself will force a brand new application server ID to be created always, overwriting any existing one.

Deleting a .dbc file

To delete a .dbc file, enter on the command line:

UNIX

```
java oracle.apps.fnd.security.AdminAppServer \
DELETE DBC=$FND_TOP/secure/file.dbc
```

Windows

```
jre -classpath %CLASSPATH% \
oracle.apps.fnd.security.AdminAppServer \
DELETE DBC=$FND_TOP/secure/file.dbc
```

You will be prompted for the APPS username and password using standard I/O.

This command deletes the .dbc file and disallows access to the indicated database if

Server Security is active.

JDBC Parameters

Note that in the DBC files, any line that starts with `JDBC\:` indicates that there parameters that are passed directly to the JDBC drivers when a connection is opened. You should refer to the JDBC documentation on Oracle Technology Network for more information on these parameters and how they are used.

For example,

```
JDBC\:oracle.jdbc.maxCachedBufferSize
```

is a parameter used to limit the amount of memory available to the drivers that perform this caching. By default it has been configured to better control the memory used by the drivers to cache results from queries. If this parameter is not set, the JDBC drivers will use all the unused memory on any JVM retaining weak references to available memory.

For more information specifically on memory management in JDBC drivers, see the white paper "Oracle JDBC Memory Management" located at http://www.oracle.com/technetwork/database/enterprise-edition/memory.pdf.

Troubleshooting

The following are possible problems you may encounter and suggested solutions.

- Database connection failed.

  Check to see if your JDBC environment is correct. See: AdminAppServer Utility, page 1-8.

- File I/O error while adding the server.

  Check to see if the path you supplied as SECURE_PATH exists and that you have write permissions on it.

- Unable to read environment file.

  A value for SECURE_PATH may not have been specified. If a value is not specified, the AdminAppServer utility assumes you are running from JAVA_TOP and looks for the file $JAVA_TOP/oracle/apps/env.html to find the value of FND_TOP. Retry the command specifying the value of SECURE_PATH.

- Database error: Unique constraint violated.

  There can be only one entry for each application server per database. If you do not specify the value for SERVER_ADDRESS, the AdminAppServer utility will default the IP address of the machine from which you are running the command. To resolve this issue, run the STATUS command of AdminAppServer to ensure you are not trying to create a duplicate entry. Delete the old entry if you want to replace it. Retry, supplying the correct value for SERVER_ADDRESS.

Administering Server Security

Oracle E-Business Suite is deployed in a multi-tier configuration with one database server and many possible middle-tier application servers. The application servers can include Apache JSP/Servlet, Forms, other servers, and some client programs such as Application Desktop Integrator. Any program which makes a SQL*Net connection to the Oracle E-Business Suite database needs to be trusted at some level. Oracle E-Business Suite uses the Server Security feature to ensure that such SQL*Net connections are coming from trusted machines and/or trusted programs.

The Server Security feature of Oracle Application Object Library supports authentication of application server machines and code modules in order to access the database. When Server Security is activated, application servers are required to supply server IDs (like passwords) and/or code IDs to access a database server. Server IDs identify the machine from which the connection is originating. Code IDs identify the module and patch level from which the connection is originating. Code IDs are included in applications code by development. The database server can be set to allow access only from specific machines and/or by code at a desired patch level.

The application server security feature is not initially activated. You should activate it by using the commands described in this section.

Application Server Security can be in one of three states:

- SECURE - Full trust is required for access to the database. Only registered application server machines may connect to the database, and only trusted code modules may connect to the database. This is the default setting for Release 12 and higher.

- OFF - Server security is not checked. Any application server machine can access the database. Code IDs are also not checked. Use this option on test systems or if you have full control over the software on all machines which can physically access your database.

- ON - Some level of trust is required to access the database. Either the application server must be registered with the database or the code must pass a module and version ID known to be trusted. Use this option only if you wish to maintain compatibility with application servers that you cannot yet patch to the code level required for best security.

Server Security Status

You can check the Server Security status for a particular database using the STATUS command in the AdminAppServer utility, and providing the dbc file corresponding to that database. The STATUS command will display whether server security is turned on and whether the server ID in the dbc file is currently valid.

To check the Server Security status for a database:

On UNIX:

```
java oracle.apps.fnd.security.AdminAppServer \
     STATUS DBC=<dbc file path>
```

On Windows:

```
jre -classpath %CLASSPATH% \
     oracle.apps.fnd.security.AdminAppServer \
     STATUS DBC=<dbc file path>
```

You will be prompted for the APPS username and password using standard I/O.

> **Important:** Check the Server Security status of your databases before you activate server security and ensure that all desired application servers have been registered.

Adding, Updating, or Deleting Server IDs

Application servers can be registered as trusted machines with a database server. This works by generating a large random ID number and storing that in both the application server and the database. When the application server attempts to connect to the database it will supply its server ID and the database will verify that it matches a trusted machine. The server ID is stored as a parameter in the DBC file for each application server. It can also be defined as an environment variable. The AdminAppServer utility is used to generate server IDs and then register them with a database server.

To add a server ID

Server ID values are generated by the AdminAppServer utility, and therefore cannot be supplied on the command line. They will be added to the database automatically when the AdminAppServer is used to create a dbc file.

On UNIX:

```
java oracle.apps.fnd.security.AdminAppServer \
ADD [SECURE_PATH=$FND_TOP/secure] \
DB_HOST=<database host> \
DB_PORT=<database port> \
DB_NAME=<database sid>
```

On Windows:

```
jre -classpath %CLASSPATH \
oracle.apps.fnd.security.AdminAppServer \
ADD [SECURE_PATH=$FND_TOP/secure] \
DB_HOST=<database host> \
DB_PORT=<database port> \
DB_NAME=<database sid>
```

You will be prompted for the APPS username and password using standard I/O.

To update a server ID

You can update an application server's server ID at any time. From the command line enter:

On UNIX:

```
java oracle.apps.fnd.security.AdminAppServer \
UPDATE DBC= <dbc file path> APPL_SERVER_ID
```

On Windows:

```
jre -classpath %CLASSPATH% \
oracle.apps.fnd.security.AdminAppServer  \
UPDATE DBC= <dbc file path> APPL_SERVER_ID
```

You will be prompted for the APPS username and password using standard I/O.

By providing the APPL_SERVER_ID argument, you will force a new server ID to be generated and added to your DBC file. If the APPL_SERVER_ID argument is not provided, AdminAppServer will take care of syncing up the server ID's of your dbc file and your database automatically, if required.

To delete a server ID

Server IDs can be deleted. This must be done using the AdminAppServer utility as follows:

On UNIX:

```
java oracle.apps.fnd.security.AdminAppServer \
DELETE DBC= <dbc file path> APPL_SERVER_ID
```

On Windows:

```
jre -classpath %CLASSPATH% \
oracle.apps.fnd.security.AdminAppServer \
DELETE DBC= <dbc file path> APPL_SERVER_ID
```

You will be prompted for the APPS username and password using standard I/O.

Activating Server Security

You can turn the server security feature to OFF, ON, or SECURE mode using the AdminAppServer utility. When you turn off server security, you will not change or delete your server IDs. You can re-enable server security without having to recreate server IDs for existing registered application servers.

On UNIX:

To activate basic server security on UNIX, enter on the command line:

```
java oracle.apps.fnd.security.AdminAppServer \
    AUTHENTICATION ON DBC=<dbc file path>
```

To activate full server security (SECURE mode) on UNIX, enter:

```
java oracle.apps.fnd.security.AdminAppServer \
    AUTHENTICATION SECURE DBC=<dbc file path>
```

To deactivate server security on UNIX, enter:

```
java oracle.apps.fnd.security.AdminAppServer \
    AUTHENTICATION OFF DBC=<dbc file path>
```

You will be prompted for the APPS username and password using standard I/O.

On Windows:

```
jre -classpath %CLASSPATH% \
 oracle.apps.fnd.security.AdminAppServer \
 AUTHENTICATION ON DBC=<dbc file path>
```

To activate full server security (SECURE mode), from the command line, enter:

```
jre -classpath %CLASSPATH% \
    oracle.apps.fnd.security.AdminAppServer \
    AUTHENTICATION SECURE DBC=<dbc file path>
```

To deactivate server security, from the command line, enter:

```
jre -classpath %CLASSPATH% \
    oracle.apps.fnd.security.AdminAppServer \
    AUTHENTICATION OFF DBC=<dbc file path>
```

You will be prompted for the APPS username and password using standard I/O.

Restricting Access to Responsibilities Based on User's Web Server

Oracle E-Business Suite has the capability to restrict access to a responsibility based upon the Web server from which the user logs in. This capability is provided by tagging Web servers with a "server trust level."

The server trust level indicates the level of trust associated with the Web server. Currently, three trust levels are supported: 1) administrative, 2) normal, and 3) external. Web servers marked as "administrative" are typically those used exclusively by system administrators. These servers are considered secure and may access restricted responsibilities and information. Servers marked as "normal" are those used by employees within a company's firewall. Users logging in from normal servers have access to only a limited set of responsibilities. Lastly, servers marked as "external" are those used by customers or employees outside of a company's firewall. These servers have access to an even smaller set of responsibilities.

Setting the Server Trust Level for a Server

To assign a trust level to a Web server, set the Node Trust Level profile option. The Node Trust Level profile option uses the Server profile hierarchy type, meaning that the value of the profile depends on the particular middle-tier server accessing the profile. This profile option can be set to either 1, 2, or 3, with the following meanings.

- 1 - Administrative

- 2 - Normal
- 3 - External

To avoid having to set the Node Trust Level profile option for every Web server, you may wish to set it to a default level of trust at the site level, such as level 1. If no value is set for the Node Trust Level profile option for a Web server, the Web server is assumed to have a trust level of 1 (Administrative).

Restricting Access to a Responsibility

To restrict access to a responsibility, set the security-based Responsibility Trust Level (internal name APPL_SERVER_TRUST_LEVEL) profile option value for that responsibility to be the number 1, 2, or 3. Setting this profile value ensures that only Web servers with the same or greater privileged trust level may access that responsibility.

Like the Node Trust Level profile option, the default value for the Responsibility Trust Level is 1.

When fetching the list of valid responsibilities for a user, Oracle E-Business Suite checks to find only responsibilities with a Responsibility Trust Level value greater than or equal to the Web server's Node Trust Level. In this way, a responsibility with Responsibility Trust Level set to 1 would only be available if the Web server has the Node Trust Level set to 1 as well. A responsibility with Responsibility Trust Level set to 2 would only be available if the Web server has Node Trust Level set to either 1 or 2.

Oracle Application Object Library AOL/J Setup Test Suite

The AOL/J Setup Test Suite consists of Java Server Pages (JSPs) and can be used to diagnose AOL/J setup issues. These JSPs exercise various features of AOL/J and provide feedback on the results. The test suite is accessed from the URL:

```
http://<host_name>:<port_number>/OA_HTML/jsp/fnd/aoljtest.jsp
```

where host_name and port_number correspond to the host name and port number of your instance's Apache listener. The host name and port number values are normally found in the APPS_SERVLET_AGENT profile option.

When you access the test suite, you will be asked to provide login information for your instance:

- Applications Schema Name
- Applications Schema Password
- TWO_TASK
- Host Name

- Port Number

The following is a list of functions and tests you can run with your instance.

- Connection Test
  - Locate DBC File
  - Verify DBC Settings
  - AOL/J Connection Test
- Virtual Directory settings
- APPS_WEB_AGENT
  - Virtual Directory Settings
  - FND_WEB.PING
  - Custom Authentication
  - GFM (Generic File Manager)
- APPS_SERVLET_AGENT
  - Virtual Directory Settings
  - Servlet Ping
  - Jsp Ping
- APPS_FRAMEWORK_AGENT
  - Virtual Directory Settings
  - Servlet Ping
  - Jsp Ping
  - Cabo Setup Tests
  - X Server Accessibility
  - OA Framework System Info
  - Versions for Loaded Classes
- HELP_WEB_AGENT

- TCF
  - Test Connection
- Tool Launcher Profile Settings
  - ICX_FORMS_LAUNCHER
  - ICX_REPORT_LAUNCHER
  - ICX_DISCOVERER_LAUNCHER
- Application Login
  - Login Page
  - Show Responsibilities
  - Show Session Properties

Configuration for Oracle Application Framework Features

Oracle Application Framework is an Oracle E-Business Suite development and deployment platform. It is a 100% Java and XML, middle-tier application framework and services for the rapid development and deployment of HTML-based applications.

The following user interface features are controlled by the profile options listed in the table below.

> **Note:** See the *Oracle E-Business Suite User's Guide* for additional information on these features.

Feature	Profile Option	Profile Option Values	Default Value on Site Level
Configurable Home Page	FND: Disable Configurable Home Page	True, False	False
Slideout Menu for Navigator and Favorites	FND: Disable Navigator and Favorites Rich Menu	True, False	False

Feature	Profile Option	Profile Option Values	Default Value on Site Level
Inline Attachments	FND: Disable Inline Attachments	True, False	False
Look-ahead LOV	FND: Disable Look Ahead LOV	True, False	False
Look Ahead in Search field for Navigator Menu	FND: Minimum Characters for Look Ahead	[Numeric value]	3
AutoVue Document Print Service for File-type Attachments	FND Attachment AutoVue Server	AutoVue web service's endpoint URL	NULL
Number of Open Notifications Shown in Home Page	Sign-On:Notification	Yes, No	No
Disabling Access Keys in All Accessibility Modes	FND:Disable Access Keys	Yes, No	No

The following profiles are retained for backward compatibility with Release 12.1.2:

- Enable Configurable Homepage (FND_CONFIG_HOMEPAGE_ENABLED).
- FND Slideout Menu (FND_SLIDEOUT_MENU)

To disable the Navigator and Favorites menus, the profile values should be set as follows:

- FND Slideout Menu (FND_SLIDEOUT_MENU) = 'Disabled'. This profile provides backward compatibility with Release 12.1.2.
- FND: Disable Navigator and Favorites Rich Menu (FND_DISABLE_NAVIGATOR_AND_FAVORITES_RICH_MENU) = 'True'

Inline Attachments

Inline Attachments can only be enabled if the accessibility mode is set to None using the profile option Self Service Accessibility Features. Inline Attachments are disabled if accessibility is set to "Screen Reader Optimized" mode or "Standard Accessibility" mode. See the *Oracle E-Business Suite User's Guide* for more information on accessibility.

Printing File-type Attachments with Autovue Document Print Service

The AutoVue Document Print Service feature can be used to print attachments of the type File. To enable this feature, the profile FND Attachment AutoVue Server (FND_AUTOVUE_SERVER) must be set to AutoVue web service's endpoint URL, in the form `http://<host>:<port>/AutoVueWS/VueBeanWS`.

Once this profile is set, the following appears in the UI for Attachments:

- In the Attachments page, a print column will be displayed in the Attachments table. This icon will be enabled only for File- type attachments and disabled for Text and URL type attachments.

- Clicking on the Print icon shall invoke the Print Attachment popup window, in which the user can choose a printer from the list of available printers, choose the paper size from the list of available paper sizes, and specify the print orientation as portrait or landscape.

- An error message is displayed if the web service could not be invoked to fetch the list of printers and paper sizes.

- Clicking on Print button shall print the file with the selected setting. A processing icon shall be rendered while printing when invoked from the Attachment List popup.

  Note: A user can only select the Print Attachment icon when the file-type attachment is committed to the database. If the Automatic Save property of an attachment region item is set to False, a user must explicitly commit the transaction of any newly-added file attachments before the user can print those attachments.

Note: For additional information about Oracle AutoVue, refer to the AutoVue Documentation Library.

Disabling Access Keys

End users have the option of disabling the access keys in all the accessibility modes using the Disable Access Keys check box in the Preferences page with name Disable Access Keys. When the box is checked, access keys will be disabled and when it is unchecked access keys are enabled for that user. By default the check box is unchecked.

Access keys can also be controlled through the profile option FND:Disable Access Keys (FND_DISABLE_ACCESS_KEYS). Valid values are:

- Yes - Access keys are disabled in all the pages regardless of the accessibility mode.

- No - Access keys are enabled in all the pages regardless of the accessibility mode.

The default value of this profile is No.

The user setting in the Preferences page overrides the profile value setting.

Related Topics

Oracle Application Framework Developer's Guide, available from My Oracle Support Knowledge Document 1315485.1

OA Extension Component Reference, available from My Oracle Support Knowledge Document 1315505.1

Oracle Application Framework Profile Options, available from My Oracle Support Knowledge Document 1315510.1

AutoConfig and Oracle Applications Manager

Oracle Applications Manager (OAM) provides access to the AutoConfig configuration management tool via the AutoConfig tab in the Site Manager screen. Previous configuration settings can easily be compared with the current settings, allowing changed settings to be identified and rolled back as necessary. The configuration editor also helps you add custom variables to the context files, to allow AutoConfig and AutoPatch to preserve your system configuration customizations.

For more information on AutoConfig, see: Using AutoConfig to Manage Oracle E-Business Suite Services, page 3-5 and Using AutoConfig Tools for System Configuration, page 3-9.

For more information on Oracle Applications Manager, see: Introduction to Oracle Applications Manager, page 2-4.

From the Oracle Applications Manager Site Map, AutoConfig is on the Administration tab under the System Configuration heading. Click on the AutoConfig link to go to the tool's home page.

From the AutoConfig home page, you can view details of a context file by clicking Show. For example, you can see Version, Path, Creation Date. Last Updated By, Status (for example, Write succeeded), and Comments.

You can select a context file and choose Edit Parameters. This is the only supported way to modify parameters that AutoConfig maintains: if you edit any context files manually, they will be overwritten the next time AutoConfig is run.

Under Edit Parameters, choosing the Local tab will open a screen that shows environment-related details, including local host machine name, virtual hostname (used to connect), local domain name, and platform type.

You can select a context file and click on Show History, to see details of any previous configurations.

The Custom tab under Edit Parameters allows you to add a new context variable (not maintained by AutoConfig) to the context file. This is the only supported way to add

customizations.

Related Topics

Configuration Overview, page 2-4

User Desktop Setup

Refer to the following documents for information on setting up hardware and software on users' desktop clients for use with Oracle E-Business Suite:

- "Oracle E-Business Suite Desktop Client Hardware and Software Requirements," My Oracle Support Document 1155883.1
- "Recommended Browsers for Oracle E-Business Suite Release 12," My Oracle Support Document 389422.1
- "Using Microsoft Office 2007 and 2010 with Oracle E-Business Suite 11i and R12," My Oracle Support Document 1077728.1

2

System Administration Setup Tasks

Basic Setup Tasks for the Oracle E-Business Suite System Administrator

This section contains an overview of each task you need to complete before you can use Oracle E-Business Suite products.

Setup Checklist

After you log on to Oracle E-Business Suite as a system administrator, complete the following steps to set up your Oracle E-Business Suite instance:

- Create Accounts for Implementors to Complete Setting Up, page 2-2
- Create New Responsibilities (Optional), page 2-2
- Create Additional Users, page 2-2
- Set Up Oracle Applications Manager, page 2-2
- Set Up Your Printers, page 2-2
- Specify Your Site-level and Application-level Profile Options (Required with Defaults), page 2-3
- Define Your Concurrent Managers (Optional), page 2-3
- Define Request Sets (Optional), page 2-3
- Set Up AuditTrail (Optional), page 2-4
- Define Globalization Options (Optional), page 2-4
- Specify Preferences for Oracle Workflow Notifications, page 2-4

Setup Steps

Create Accounts for Implementors to Complete Setting Up

Create individual Oracle E-Business Suite accounts for users who will be completing the implementation of your Oracle E-Business Suite. Assign these users the full access responsibilities for the products they will be implementing.

> **Note:** Updates by the SYSADMIN user are treated as seed data when applying patches.

See: Overview of Oracle E-Business Suite Security, *Oracle E-Business Suite Security Guide*.

Create New Responsibilities (Optional)

A responsibility in Oracle E-Business Suite is a level of authority that determines how much of an application's functionality a user can use, what requests and concurrent programs the user can run, and which applications' data those requests and concurrent programs can access. Oracle E-Business Suite provides a set of predefined responsibilities that you can use. You can also define your own responsibilities if the ones provided do not meet your needs.

See: Overview of Oracle E-Business Suite Security, *Oracle E-Business Suite Security Guide*.

Create Additional Users

You should use the procedure outlined in Step 1 to create additional application users. When you define a new user, you assign one or more responsibilities and a password that the user changes after the initial logon. You can use the LOV in the Responsibility field to get a list of the standard responsibilities for each application you specify. You can assign multiple responsibilities to a user.

See: Overview of Oracle E-Business Suite Security, *Oracle E-Business Suite Security Guide*.

Set Up Oracle Applications Manager

Oracle Applications Manager (OAM) allows you to configure and maintain many components of the Oracle E-Business Suite system. For information on setting up OAM, see: Oracle Applications Manager Setup, page 2-4.

Set Up Your Printers

Read the Setting Up Your Printers, page 6-8 page to learn how to set up your printers. You must define any printer types used at your site that are not shipped with Oracle E-Business Suite, then register each printer with its name as determined by your operating system.

For every custom printer type or specialized print style you define, use the Printer

Drivers form to assign a printer driver to use with each print style used by a printer type.

If you need more information on how to find your printer operating system names, refer to the Printing section of *Oracle E-Business Suite Installation Guide: Using Rapid Install* .

For more information on setting up your printers, see: Printers and Printing, page 6-1.

Specify Your Site-Level and Application-Level Profile Options

Use the System Profile Values form (Profile > System) to set site-level and other profile optons..

Optionally set your Site Name profile option to your site name.

Many profile options are set by AutoConfig, and their values can be reviewed in Oracle Applications Manager.

Define Your Concurrent Managers (Optional)

Concurrent Processing is a feature of Oracle E-Business Suite that lets you perform multiple tasks simultaneously. Oracle E-Business Suite Concurrent Processing lets you run long, data-dependent functions at the same time as your users perform online operations. Concurrent managers are components of concurrent processing that monitor and run your time-consuming tasks without tying up your computers.

Oracle E-Business Suite automatically installs one standard concurrent manager that can run every request. You may want to take advantage of the flexibility of concurrent managers to control throughput on your system.

You can define as many concurrent managers as you need. Keep in mind, however, that each concurrent manager consumes additional memory.

You can specialize each of your concurrent managers so that they run all requests, requests submitted by a particular user, requests submitted by a particular application, or other constraints, or any combination of these constraints.

If you are using Parallel Concurrent Processing in a cluster, massively parallel, or homogeneous networked environment, you should register your Nodes and then assign your concurrent managers to primary and secondary nodes. You can spread your concurrent managers, and therefore your concurrent processing, across all available nodes to fully utilize hardware resources.

Use the Define Concurrent Manager form to define new concurrent managers.

Define Request Sets (Optional)

A request set is a group of reports or programs which you submit with one request. To define and maintain request sets, use the Request Sets form.

Users can also define their own request sets.

Set Up AuditTrail (Optional)

If you want to keep track of the changes made to your data by application users, you should set up AuditTrail for the relevant tables.

Defining AuditTrail for your site involves defining Audit Groups, which are groups of tables and columns for which you intend to track changes. You then define Audit Installations to instruct AuditTrail which ORACLE IDs you want to audit. Finally, you run the Audit Trail Update Tables Report, which allows your AuditTrail definitions to take effect.

Define Globalization Options (Optional)

Optionally define settings for globalization (formerly internationalization) features. These include, but are not limited to, the following features.

Modify Language Prompts (Optional)

If you want to modify the field name displayed in the Translations window, you should change the Description value for the language you want to modify in the Languages window.

Modify Territory LOV Values (Optional)

If you want to modify the territory value displayed in LOVs, you should change the Description value for the territory you want to modify in the Territories window.

Specify Preferences for Oracle Workflow Notifications (Required)

The SYSADMIN user is the default recipient for some types of notifications in Oracle E-Business Suite, such as error notifications. You need to specify how you want to receive these notifications by defining the notification preference and e-mail address for the SYSADMIN user.

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 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.

Setup Tasks for Oracle Applications Manager

Introduction to Oracle Applications Manager

Oracle Applications Manager (OAM) allows administrators to manage Oracle E-Business Suite systems from an HTML console. Utilities available from OAM include

Oracle Workflow Manager, Patch Wizard, and Concurrent Processing monitoring tools.

With Oracle Applications Manager, system administrators can view information on general system activity including the statuses of the database, concurrent managers and other services, concurrent requests, and Oracle Workflow processes. OAM provides a summary of configuration changes, infrastructure usage, performance, required maintenance activities, potential security issues, status of business flows, and diagnostic test results. In addition, they can manage downtime and patching. System administrators can also start or stop services, and submit concurrent requests.

Using Oracle Workflow Manager, administrators can control Workflow system services, such background engines, the Notification Mailer, agent listeners, queue propagation, and purging obsolete Workflow data.

OAM utilities are generally available from two main screens: the Applications Dashboard and Site Map. See: Applications Dashboard, *Oracle E-Business Suite Maintenance Guide* and The Site Map, page 2-4 for more information on these.

Refer to the following documentation for additional information:

- *Oracle E-Business Suite System Maintenance Guide*
- *Oracle Workflow Administrator's Guide*

The Service Fulfillment Manager may optionally be installed. For more information, see: http://www.oracle.com/appsnet/products/service/index.htm.

Function Security and Oracle Applications Manager

Oracle Applications Manager is accessed with the Oracle Application Object Library's function security model. It is included on the menus for the seeded System Administration and System Administrator responsibilities. In addition, you can create custom responsibilities and menus to control access to specific OAM features. These features can thus be directly available from the Oracle E-Business Suite Home Page.

The Site Map

The Site Map lists the features and applications available in the Oracle Applications Manager. Features are grouped into the following categories: Administration, Monitoring, Maintenance, and Diagnostics and Repair.

Site Map - Administration

System Configuration

These features provide detailed information on the configuration of your system. You can update many of your configuration settings from these links also.

- Hosts - For each of your hosts, you can view its status and configuration. You can also bring it online or offline, or disable it.
- AutoConfig - View and update your AutoConfig settings here.

- License Manager - With License Manager you can license additional products, country-specific functionalities and languages. You can also generate reports on the licenses for your installation.

Application Services

Use these links to see information on various types of application services.

- Generic Services
- Request Processing Managers
- Transaction Managers

Workflow

Use these links to navigate to the Oracle Workflow Manager.

- Home
- Work Item Metrics
- Agent Activity
- Background Engines
- Notification Mailer
- Service Components
- Purge

Concurrent Request

Use these links to submit a new concurrent request or to view details on existing requests.

- Submit New - this link launches the Oracle Application Framework Schedule Request page in a separate window.
- Pending
- Running
- Completed (Last Hour)

Service Fulfillment Manager

Use this link to access the Service Fulfillment Manager. Service Fulfillment Manager (SFM) provides a complete set of tools to automate step-by-step fulfillment activities and integrate business flows for any type of service across multi-vendor application systems.

Others

- Applications Manager Log

Site Map - Monitoring

The Monitoring section links you to features to help you monitor your Oracle E-Business Suite.

Availability

Use these links to navigate to pages on the availability of these components.

- Hosts
- Database
- Web components
- Internal Concurrent Manager
- Request Processing Managers
- Transaction Managers
- Forms
- Workflow
- Business Flows
- Weblogic Administration Console
- Fusion Middleware Control

Performance

Use these links to see information regarding performance of these components.

- SQL Activity
- Forms Sessions
- Forms Runtime Processes
- Concurrent Processing Reports
- Concurrent Processing Charts
- Workflow

Current Activity

Use these links to view activity information for their respective areas.

- System Alerts
- Database Sessions
- Invalid Objects
- Forms Runaway Processes
- Forms Sessions
- Forms Runtime Processes
- Application Services
- Activity Monitors
- Concurrent Requests
- Critical Activities
- Logs

System Configuration

From the Overview link you can navigate to the Configuration Overview page. See: System Configuration Overview, page 2-9.

You can also access information on the database init.ora parameter settings, and the applications context configuration information, and site-level profile settings.

Usage

Use these reports to learn more about application usage and concurrent processing.

Custom Reporting Utilities

The SQL Extensions page enables you to run seeded and custom scripts.

Site Map - Maintenance

These features help you maintain your Oracle E-Business Suite installation.

Patches and Utilities

- Applied Patches
- File History
- Patch Wizard
- Timing Reports
- Manage Downtime Schedules - this feature should not be used. Refer to the Patching section of the *Oracle E-Business Suite Maintenance Guide* for information on managing downtimes.

- Codelevels Summary - The Codelevels Summary page lists the code level for each product in your system.

Critical Activities

- Setup
- Monitor

Site Map - Diagnostics and Repair

Use these features in diagnostics and troubleshooting.

Diagnostics

The Diagnostics tab provides links to the following Oracle E-Business Suite Diagnostics pages in which you can run and manage tests and view reports:

- Execute Diagnostic Tests
- View Diagnostics Reports
- View Scheduled Diagnostics Tests
- Diagnostics Configuration

For more information on Oracle E-Business Suite Diagnostics, see the *Oracle Diagnostics Framework User's Guide*.

In addition, the Debug Workbench enables you to centrally control and monitor the debugging of Oracle E-Business Suite components. Using the Debug Workbench, you can set up debug rules for system components and view the debug information that has been collected. For more information, see: Debug Workbench, *Oracle E-Business Suite Maintenance Guide*.

The Client Configurations link allows you to view client configuration data about the hardware and software of your computer. For more information about using the Client System Analyzer from the Oracle E-Business Suite, see My Oracle Support Knowledge Document 277904.1.

Troubleshooting Wizards

- Concurrent Manager Recovery
- Service Infrastructure
- GCS and Forms Monitoring
- CP Signature
- Dashboard Collection Signature

System Configuration Overview

This page contains configuration information for the Oracle E-Business Suite system's configuration.

Overview

This page provides information on the following:

- Database Configuration
- Operating Units
- Registered Oracle Schemas
- Registered Applications
- Base and Installed Languages
- Localization Modules

Database

The Database section lists the database instances for the system, with this information:

- Host Name
- Instance Name
- Version
- Instance Number

Click on **NLS Parameters** to see a list of these parameters and their values. Click on **Initialization Parameters** to see a listing of these parameters.

Concurrent Processing

This section lists general information related to concurrent processing for this Oracle E-Business Suite system and its concurrent processing settings.

A list of concurrent processing servers with their host names and platforms is shown. Also, the following site-level values for the Internal Concurrent Manager (ICM) are shown:

- Concurrent: Attach URL - If this profile option is set to Yes, a URL is attached to request completion notifications. When a user submits a request, and specifies people to be notified in the Defining Completion Options region, everyone specified is sent a notification when the request completes. If this profile option is set to Yes, a URL is appended to the notification that enables them to view the

request results online.

- Concurrent: Sequential Requests - You can force your requests to run one at a time (sequentially) according to the requests' start dates and times, or allow them to run concurrently when their programs are compatible. A value of Yes prevents your requests from running concurrently. Requests run sequentially in the order they are submitted. A No value means your requests can run concurrently when their concurrent programs are compatible.

- Sleep Time - The duration of time in seconds that the ICM should wait before checking for new requests. PMON Cycle Time - The duration of time in seconds between "process monitor checks" (checks for failed workers). Queue Sizing Interval - The duration of time in seconds between "worker quantity checks" (checks for the number of active workers).

Click on **ICM Environment** to see the environment variables and their values.

Forms

This section shows the ICX: Forms Launcher profile option setting, which should be set to the base URL for launching Oracle E-Business Suite forms.

A List of Forms Servers is also shown, with the Host Name, Port, and Log File location for each Oracle Forms server.

Web

This section shows the following profile option settings:

- Applications Web Agent - Provides the base URL for the Apps Schema's WebServer DAD. Oracle E-Business Suite uses the value of this profile option to construct URLs for 'WWW' type functions, Attachments, Export, and other features.

- Applications Servlet Agent - This profile option must be set to the URL base for the servlet execution engine on Apache. Oracle E-Business Suite uses the value of this profile option to construct URLs for JSP and SERVLET type functions.

  The syntax is: http://<hostname>:<port>/<servlet_zone>

- Application Framework Agent - This profile option must point to the Apache Server.

A list of Web Servers is also shown, with the Host Name, Port, and Log File location for each web server.

Other

This section shows the following profile option settings:

- TCF:HOST and TCF Port - These profile options identify the network location of the TCF Server. The TCF Server supports various parts of the Oracle E-Business Suite

user interface by executing some of their associated server logic and providing access to the database.

Oracle Applications Manager and Oracle Enterprise Manager 11g Fusion Middleware Control

You can use links from the Oracle Applications Manager Site Map Monitoring tab to access the Oracle Enterprise Manager 11g Fusion Middleware Control console and the Oracle WebLogic Server 11g Administration Console.

In addition, from the Oracle Enterprise Manager 11g Fusion Middleware Control console, you can launch some of the functions of Oracle Applications Manager, including:

- Performance
- Hosts
- Forms Sessions
- Forms Runtime Processes
- AutoConfig
- Manage Downtime Schedules

These functions are linked from the E-Business Suite > Management target from the Home page. For more information on these features, see the *Oracle E-Business Suite Maintenance Guide*.

For more information on Oracle Enterprise Manager, see: *Oracle Enterprise Manager Getting Started with Oracle Fusion Middleware Management 11g Release 1 (11.1.0.1)* and related documentation.

Setup Tasks

Oracle Applications Manager (OAM) allows you to customize how certain components are monitored and how metrics are collected.

Navigation: Setup (global icon)

The Dashboard Setup page displays a summary of data collection for metrics and services. For each metric, you can see whether collection is enabled and whether you have alerting enabled for the metric. If alerting is enabled, you see the condition which must be met for an alert to be raised. To update the setup for the data collection, use the Dashboard Setup Wizard.

MetaLink Credentials

You maintain your My Oracle Support (formerly Oracle*MetaLink*) username, password, and e-mail address from the Metalink Credentials page. These credentials will be used

when querying My Oracle Support through Knowledge Base links. Also, you should ensure that the following profile options are set appropriately for your proxy server:

- Applications Proxy Bypass Domains
- Applications Proxy Port
- Applications Server-Side Proxy Host and Domain

Business Flows Setup

Enable or disable monitoring of business flows.

Click the Metrics Refresh link to schedule requests for the OAM: KBF Metrics Rollup Program to update the setup status of your business flows

Knowledge Base

The Knowledge Base provides a catalog of useful documents relevant to managing your system.

Concurrent Requests

From this page you can enable alerting for concurrent requests that have been running or pending for a long time. You can specify the thresholds for which request must reach before alerts are raised.

Specifically, you can enable the system to do the following:

- Raise a general alert for any long-running requests, for any concurrent program.
- Raise an alert for a long-running request for a specific program. Specify the concurrent program(s) you want to monitor.
- Raise a general alert for any long-pending requests, for any concurrent program.
- Raise an alert for a long-pending request for a specific program. Specify the concurrent program(s) you want to monitor.

Signon Audit Setup

From this page, select the Enable Auditing button to set the Sign-On: Audit Level profile option to 'FORM', which enables Forms monitoring. Selecting the Disable Auditing button sets this profile option to 'NONE'.

Use the Enable Alerting/Disable Alerting buttons to control whether an alert should be raised if the Sign-On: Audit Level profile option is set at a value other than 'FORM'.

Dashboard Setup Wizard

Select the Update button to configure how data for the following are collected.

Metrics

Specify how you want metrics to be collected for the following:

- Activity
- Configuration Changes (last 24 hours)
- System Alerts
- Web Components Status
- User-Initiated Alerts

Services

Specify which services you want to monitor. For a given service, you can specify if you want to collect data for it and whether you want to be alerted if the service is in a specified status.

3

Technical Configuration

Introduction

AutoConfig has historically been the tool to support automated configuration of the Oracle E-Business Suite instance. The information required for configuring an Applications instance is collected into two local repositories, called the Applications context file and the database context file. When AutoConfig runs on the application tier, it uses information from the Applications context file to generate configuration files and update database profiles. When AutoConfig runs on the database tier, it uses information from the database context file to generate all configuration files used on the database tier and update database profiles.

Configuration Management Tools

Fusion Middleware Control: This tool provides a high-level view of Oracle WebLogic Server (WLS). More significantly for the Oracle E-Business Suite DBA, it is used to configure Oracle HTTP Server. HTTP settings include: virtual hosts, performance directives, log configuration, ports, mod_perl, and mod_wl_ohs. Fusion Middleware Control also includes links to Oracle Application Manager and Oracle WebLogic Server Admin Console.

WebLogic Server Administration Console: Handles Oracle WebLogic Server settings and managed servers. Examples include: oacore, oafm, forms, and forms-c4ws services.

Oracle Application Manager and AutoConfig: Handles Oracle Database settings. Examples include: SID name, Listener, dbPorts.) Also handles some Oracle E-Business Suite settings. Examples include: Concurrent Processing, Profile Options, Developer 10g settings, product-specific settings.

Configuration Management Changes

In Oracle E-Business Suite Release 12.2, OC4J has been replaced with *Oracle WebLogic Server*. This has resulted in a reduced role for AutoConfig in the configuration of the

Oracle HTTP Server and the oacore, oafm, forms and forms-c4ws services.

Up to and including Oracle E-Business Suite Release 12.1.3, AutoConfig was used to manage the entire Oracle HTTP Server configuration and OC4J instance configuration. In Oracle E-Business Suite Release 12.2, it manages only a part of the Oracle HTTP Server configuration. It also only partially manages the configuration of the oacore, oafm, forms and forms-c4ws services. The remaining scope of AutoConfig remains the same as prior to Oracle E-Business Suite Release 12.2.

This chapter details those aspects of configuration management that are still undertaken by AutoConfig. It goes on to describe the role of Oracle WebLogic Server in Oracle E-Business Suite Release 12.2, and also mentions some important WLS administrative and troubleshooting tasks.

Two key configuration changes in Release 12.2 include:

- **Fusion Middleware Control** provides a high-level view of Oracle WebLogic Server, and is the only place where you can configure your HTTP Server. Examples of HTTP settings are: virtual hosts, performance directives, log configuration, ports, mod_perl, and mod_wl_ohs. Fusion Middleware Control also has links to Oracle Application Manager and WLS Admin Console.

- **WebLogic Server Administration Console** handles Oracle WebLogic Server settings and managed servers. Examples include oacore, oafm, forms, and forms-c4ws services.

In a wider context , the changes are:

Summary of Configuration Management Changes in Release 12.2

Configuration Activity	Prior to Release 12.2	In Release 12.2
Oracle E-Business Suite Database, Concurrent Processing, Oracle Developer 10g, profile options, and other Oracle E-Business Suite components.	Oracle Applications Manager.	Oracle Applications Manager.

Configuration Activity	Prior to Release 12.2	In Release 12.2
Changes to HTTP Configuration.	All HTTP configuration was managed via AutoConfig templates. Configuration changes were done by editing the respective context variables and subsequently running AutoConfig.	Most HTTP configuration is managed via native Oracle WebLogic Server tools, Fusion Middleware Control, or manually editing of the configuration files. Only a limited set of HTTP configuration files are maintained via AutoConfig. More details are given later in this chapter.
Changes to configuration of oacore, oafm, forms and forms-c4ws services.	All configuration settings for the oacore, oafm, forms and forms-c4ws services were managed via AutoConfig templates. Configuration changes were accomplished by editing context variables and running AutoConfig.	Properties for the oacore, oafm, forms and forms-c4ws services, including the classpath and JVM arguments, need to be updated through native WebLogic tools such as WebLogic Administration Console. The context variable values are used only to set the initial values during managed server creation. More details are given later in this chapter.
Managing JVM instances of the oacore, oafm, forms and forms-c4ws services.	The number of instances of a service was controlled via Oracle Process Manager (OPMN). This number could be modified by editing the nprocs context variable, running AutoConfig, then stopping and restarting the services.	Each JVM instance of a service corresponds to a managed server of that service type. The number of instances needs to be controlled by explicitly creating or deleting managed servers for the service. More details are given later in this chapter.

Oracle WebLogic Server Requirements and Usage

Oracle E-Business Suite Release 12.2 requires WebLogic Server Basic. This is a license-constrained version of WebLogic Server that is available in licenses for certain Oracle products.

WebLogic Server Basic is used in Oracle E-Business Suite Release 12.2 to support the

following features:

- WLS Clusters. Specifically, one WLS cluster per EBS domain.
- Hardware-based load-balancers in front of a WLS cluster in an EBS domain.
- Use of the WLS proxy on an OHS server, directing load to one or more WLS instances on one or more managed servers in a WLS Cluster. The cluster is defined by AutoConfig via the configuration file deployed on the OHS server.
- Session re-instantiation from one managed server to another managed server within the same cluster. Although transactions in progress during failure of one managed server will be lost, the user's session will be re-established and migrated to another managed server in the cluster.

AutoConfig Scope and Components

The Release 12.2 application tier is AutoConfig-enabled, and has an Applications context file stored in the INST_TOP as `<INST_TOP>/appl/admin/<CONTEXT_NAME>.xml`. The Release 12.2 database tier created via Rapid Install is also AutoConfig-enabled, and has a database context file stored in the RDBMS ORACLE_HOME as `<RDBMS_ORACLE_HOME>/appsutil/<CONTEXT_NAME>.xml`.

Key AutoConfig components include:

Component	Description
Applications context	An XML repository located in the INST_TOP that contains information specific to the APPL_TOP.
Database context	An XML repository located in the RDBMS ORACLE_HOME that contains information specific to that database tier.
AutoConfig template files	Files containing named tags that are replaced with instance-specific information from the appropriate context, in the process of *instantiation*.
AutoConfig driver files	Every Oracle E-Business Suite product maintains a driver file used by AutoConfig. The driver file lists the AutoConfig file templates and their destination locations.

Component	Description
AutoConfig scripts	A set of scripts that provide a simplified interface to the AutoConfig APIs.

Using AutoConfig to Manage Oracle E-Business Suite Services

This section describes how AutoConfig manages Oracle E-Business Suite services and processes.

In previous Oracle E-Business Suite releases, the Applications services were categorized into service groups according to the type of service provided. In Oracle E-Business Suite Release 12.2, this concept has been extended by the introduction of additional services and service groups.

Most notably, the *Web Administration service group* has been introduced in Release 12.2. This service group contains WebLogic Administration server, and - unlike other service groups - can be enabled only on one of the Application tier nodes. In other words, it is not supported to enable WebLogic Administration server on any other Application tier node except the node on which it was enabled during Rapid Install.

> **Note:** Release 12.2 uses two independent port pools. This is transparent to the user. You may find it useful to make a note of the WLS Admin port, however.

Also, unlike previous releases of Oracle E-Business Suite Release 12.x, the *Root Service Group* now comprises Node Manager and not Oracle Process Manager (OPMN). In Oracle E-Business Suite Release 12.2, OPMN only manages Oracle HTTP Server. Consequently, it is now part of the *Web Entry Point Services* service group.

The following table shows the AutoConfig-managed service groups and services that exist in Release 12.2.

> **Note:** Only the UNIX versions of the service control scripts are shown: the Windows equivalents have a `.cmd` suffix instead of `.sh`.

Service Group	Service(s)	Service Control Script
Root Service	Node Manager	`adnodemgrctl.sh`
Web Administration	WebLogic Admin Server	`adadminsrvctl.sh`

Service Group	Service(s)	Service Control Script
Web Entry Point Services	Oracle HTTP Server	adapcctl.sh
	Oracle Process Manager	adopmnctl.sh
Web Application Services	oacore	admanagedsrvctl.sh
	oafm	admanagedsrvctl.sh
	forms	admanagedsrvctl.sh
	forms-c4ws	admanagedsrvctl.sh
Batch Processing Services	Oracle TNS Listener	adalnctl.sh
	Concurrent Manager	adcmctl.sh
	Fulfillment Server	jtffmctl.sh
	Oracle ICSM	ieoicsm.sh
Other Services	Forms Server	adformsrvctl.sh
	Oracle MWA Service	mwactlwrpr.sh

Note: A particular service will be started or stopped via the adstrtal or adstpall scripts only if the service and its service group are **both** enabled.

Modifying AutoConfig-Managed Services and Service Groups

Depending on the requirement of a particular Applications instance, it is possible to modify the set of Applications services and service groups that will be started and stopped via the adstrtal and adstpall scripts respectively. This can be done by enabling the required services and service groups, and disabling those that are not required.

- **Checking whether a service group/service is enabled or disabled**

  There are two options here:

  - A complete list of the status of all service groups and services is available in Section 2 of the report generated by the Check Config utility (adchkcfg).

  - The log files generated by the adstrtal and adstpall scripts list the Application service groups and the services that are managed via AutoConfig. This list also reports whether a particular service group or service is enabled or disabled.

- **Enabling or disabling a service group**

  In Oracle E-Business Suite Release 12.2, all services except the Web Administration Service Group can be enabled or disabled on any or all Application tier nodes. This is done as follows:

  1. Check the value of the 'status' context variable corresponding to the service group.

  2. Change the value of this variable to 'enabled' to enable the service group, or to 'disabled' to disable the service group.

- **Enabling a service**

  A service can be enabled as follows:

  1. Check the value of the 'status' context variable corresponding to the service group to which the service belongs.

  2. If the service group is 'disabled', enable the service group.

  3. If the value of the 'status' context variable corresponding to the service is not already set to 'enabled', change it to 'enabled'.

  4. In case of the oacore, oafm, forms and forms-c4ws services, there may be more than one instance of the service. To cater for this, check that the context variable corresponding to the name of the service contains the name of the managed server to be enabled. If not, add it to the list, using a comma as a separator.

     For example, if there are two oacore managed servers (oacore_server1 and oacore_server2) defined on the same Application tier node, and oacore_server2 needs to be enabled, then the value of the context variable s_oacorename needs to be changed from 'oacore_server1' to 'oacore_server1,oacore_server2'. Also, if the context variable 's_oacore_managed_server' does not contain an entry for 'oacore_server2', the value of this context variable also needs to be changed to 'oacore_server1,oacore_server2'.

- **Disabling a Service**

  A service can disabled as follows:

  1. In most cases, you simply need to set the service's 'status' context variable to 'disabled'.

  2. Services of type oacore, oafm, forms or forms-c4ws require an additional step. Where such a service's 'status' context variable is set to 'enabled' and there is more than one instance of the service, you must remove the name of the managed server to be disabled from the context variable corresponding to the name of the service.

For example, if there are two oacore managed servers (oacore_server1 and oacore_server2) defined on the same Application tier node, and oacore_server2 needs to be disabled, then the value of the context variable s_oacorename must be changed from 'oacore_server1, oacore_server2' to 'oacore_server1'.

Commands to Manage Oracle E-Business Suite Service Processes

- **Commands for managing processes on the Applications tier**

  The adstrtal and adstpall scripts can be used to start and stop all the AutoConfig-managed application tier services in a single operation. Alternatively, it is possible to administer the individual services separately using their respective service control scripts. The oacore, oafm, forms and forms-c4ws services can also be managed by starting and stopping the respective managed servers via the WebLogic Server Administration Console.

  All the scripts listed below are located in `<INST_TOP>/admin/scripts`.

Functionality	UNIX Command	Windows Command
Start Applications services	`adstrtal.sh`	`adstrtal.cmd`
Stop Applications services	`adstpall.sh`	`adstpall.cmd`
Start individual service (except those that are part of the Web Application Services service group)	`<control_script> start`	`<control_script> start`
Stop individual service (except those that are part of the Web Application Services service group)	`<control_script> stop`	`<control_script> stop`
Start individual managed server (all services that are part of the Web Application Services service group)	`admanagedsrvctl.sh start <managed_server_name>`	`admanagedsrvctl.cmd start <managed_server_name>`

Functionality	UNIX Command	Windows Command
Stop individual managed server (all services that are part of the Web Application Services service group) The 'stop' command will shut down the managed server only after no user sessions remain connected, while the 'abort' command will shut down the managed server immediately.	admanagedsrvctl stop <managed_server_name> admanagedsrvctl abort <managed_server_name>	admanagedsrvctl.cmd stop <managed_server_name> admanagedsrvctl.cmd abort <managed_server_name>

- **Commands for managing processes on the Database tier**

  All the scripts listed below are located in <RDBMS ORACLE_HOME>/appsutil/scripts/<CONTEXT_NAME>.

Functionality	UNIX Command	Windows Command
Start database listener process	addlnctl.sh start <SID>	addlnctl.cmd start <SID>
Stop database listener process	addlnctl.sh stop <SID>	addlnctl.cmd stop <SID>
Start database server process	addbctl.sh start	addbctl.cmd start
Stop database server process	addbctl.sh stop	addbctl.cmd stop

Using AutoConfig Tools for System Configuration

The following table summarizes the AutoConfig tools, including previously-existing tools such as adautocfg.sh and new tools such as adRegisterWLSListeners.pl and adSyncContext.pl. Further details of each tool are provided later in this chapter.

> **Note:** On Windows, command files (.cmd suffix) are the equivalent of UNIX scripts (.sh suffix).

Script Name	Location	Purpose
adautocfg.sh/cmd	On Applications Tier: <INST_TOP>/admin/scripts On Database Tier: <RDBMS_ORACLE_HOME>/appsutil/scripts/<CONTEXT_NAME>	This script is used for running AutoConfig.
adRegisterWLSListeners.pl	On Applications Tier: <AD_TOP>/bin	This script is used to listen to changes to the WebLogic Server configuration parameters and update the context variables accordingly.
adSyncContext.pl	On Applications Tier: <AD_TOP>/bin	This script is used to explicitly pull the values of the WebLogic Server configuration parameters and synchronize the context variable values with these. These are needed for synchronization of the OHS parameters.
adchkcfg.sh/cmd	On Applications Tier: <AD_TOP>/bin On Database Tier: <RDBMS_ORACLE_HOME>/appsutil/bin	This script may be run before running AutoConfig to review the changes on running AutoConfig. This will generate a report showing the differences between the existing configuration and what the configuration would be after running AutoConfig.
GenCtxInfRep.pl	On Applications Tier: <FND_TOP>/patch/115/bin On Database Tier: <RDBMS_ORACLE_HOME>/appsutil/bin	This script can be used to find out detailed information about context variables and the templates in which they are used, given all or part of a context variable name as a keyword.

Script Name	Location	Purpose
adtmplreport.sh/cmd	On Applications Tier: <AD_TOP>/bin On Database Tier: <RDBMS_ORACLE_HOME>/appsutil/bin	This script can be used to gather information regarding the location of the AutoConfig templates, provided the location of the instantiated files and vice versa.
admkappsutil.pl	On Applications Tier: <AD_TOP>/bin	This script is used while applying patches to the database tier. Running this script generates appsutil.zip, which may be copied over to the database tier to migrate the patch to the database tier.

As well as the above tools, there are start and stop tools that are used to manage the run-time processes of Oracle E-Business Suite services. These tools are described later in this chapter.

As mentioned earlier, AutoConfig is used to automate system configuration. This section describes how the AutoConfig tools can be used for this purpose. Actions performed by these tools will typically be performed in the order shown in the following subsections.

Preview Effects of Running AutoConfig

Before running AutoConfig, the Check Config utility may be run to review the changes that would occur in the file system as well as the database in the next AutoConfig run. This step is optional.

Execute the applicable command to run the Check Config utility:

Database Tier

On UNIX:

```
sh <RDBMS_ORACLE_HOME>/appsutil/bin/adchkcfg.sh \
contextfile=<CONTEXT_FILE>
```

On Windows:

```
C:\><RDBMS_ORACLE_HOME>\appsutil\bin\adchkcfg.cmd
contextfile=<CONTEXT_FILE>
```

Application Tier

On UNIX:

```
$ sh <AD_TOP>/bin/adchkcfg.sh \
contextfile=<CONTEXT_FILE>
```

On Windows:

```
C:\><AD_TOP>\bin\adchkcfg.cmd contextfile=<CONTEXT_FILE>
```

This utility is described in more detail later.

Run AutoConfig on the Database Tier

Running AutoConfig on the database tier is required after:

- Migrating a patch to the database tier, the Check Config utility reports any potential changes to the templates.
- Performing customizations on the database tier.
- Performing a database or application tier upgrade.
- Restoring the database or Oracle Home from a backup.
- Upgrading JDK on the database tier.
- Manually cleaning up the Net Services Topology Information using one of the supported procedures (such as `fnd_conc_clone.setup_clean`). Subsequently, AutoConfig must be run on the application tier nodes as well.
- Registration of an Oracle RAC node.
- Setting up the APPL_TOP on a shared file system.
- Carrying out any other operation where the documentation states that AutoConfig should be run on the database tier.

Execute the following command to run AutoConfig on the database tier:

On UNIX:

```
$ sh <RDBMS_ORACLE_HOME>/appsutil/scripts/<CONTEXT_NAME>/adautocfg.sh
```

On Windows:

```
C:\><RDBMS_ORACLE_HOME>\appsutil\scripts\<CONTEXT_NAME>\adautocfg.cmd
```

Be aware of the following important points:

- The database server and database listener must remain running during an AutoConfig run, but all other database tier services should be shut down.
- AutoConfig may change your environment files, so after running it you should always set the environment (and thereby pick up any changed variables) before you run any Oracle E-Business Suite utilities.

Run SyncContext on the Application Tier

In Oracle E-Business Suite Release 12.2, some configurations are managed via AutoConfig, and some are managed natively via Fusion Middleware Control and WebLogic Server Administration Console. A new mechanism has been introduced to keep the context variables and the OHS configuration parameters (where applicable) in synchronization. This mechanism is called the 'feedback loop'. The *SyncContext* tool is one of the tools used for explicit synchronization of the context variables with the WLS configuration parameters.

This tool can be run on all application tier nodes as follows:

On UNIX:

```
$ perl <AD_TOP>/bin/adSyncContext.pl contextfile=<CONTEXT_FILE>
```

> **Important:** Node Manager and WebLogic Admin Server must be running during execution of the `adSyncContext.pl` script.

The `adSyncContext.pl` script reads the WLS configuration parameter values and synchronizes them with the context variables. This is used, for example, to synchronize the OHS configuration parameters with the respective context variables.

The `adRegisterWLSListeners.pl` script, in contrast, simply listens for changes to WLS configuration parameters in order to faciliate synchronizing their values with the relevant context variables.

Stop Application Tier Services

Before running AutoConfig on the Application tier, all application tier services must be stopped. This can be done using the following command:

On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adstpall.sh
```

On Windows:

```
C:\><ADMIN_SCRIPTS_HOME>\adstpall.cmd
```

Run AutoConfig on the Application Tier

Run AutoConfig on all application tier nodes by executing the applicable command below.

On UNIX:

```
$ sh <INST_TOP>/admin/scripts/adautocfg.sh
```

On Windows:

```
C:\><INST_TOP>\admin\scripts\adautocfg.cmd
```

Be aware of the following important points:

- The database server and database listener must remain available during the AutoConfig run, but all other database tier services should be shut down.
- Running AutoConfig may change your existing environment files.
- After running AutoConfig, you should always set the environment before you run any Oracle E-Business Suite utilities, to pick up any changed environment variables.

Review AutoConfig Log Files

AutoConfig logfiles are stored under:

Tier	Directory
Application	<INST_TOP>/admin/log/<MMDDhhmm>
Database	<RDBMS_ORACLE_HOME>/appsutil/log/<CONTEXT_NAME>/<MMDDhhmm>

One log file is created per AutoConfig session. It will contain details of every action that AutoConfig performed during the run.

Start All Application Tier Services

After running AutoConfig, start up all the application tier services by executing the applicable command below:

On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adstrtal.sh
```

On Windows:

```
C:\><ADMIN_SCRIPTS_HOME>\adstrtal.cmd
```

Run adRegisterWLSListeners Tool on the Application Tier

As noted previously, some Oracle E-Business Suite Release 12.2 configurations are managed via AutoConfig, and some are managed natively via Fusion Middleware Control or Oracle WebLogic Server Administration Console.

The `adRegisterWLSListeners` tool has been introduced to perform synchronization of context variables with the associated Oracle WebLogic Server configuration parameters.

Note: This tool does *not* listen for changes to the Oracle HTTP Server configuration parameters.

Execute the following command to start `adRegisterWLSListeners.pl`:

```
$ perl <AD_TOP>/bin/adRegisterWLSListeners.pl contextfile=<CONTEXT_FILE>
```

Once started, `adRegisterWLSListeners` keeps running, listening for changes to the WebLogic Server configuration and synchronizing the context files stored in the database.

Important: The `adRegisterWLSListeners` tool automatically stops when WebLogic Administration Server is shut down. On UNIX instances, the tool is started automatically started whenever WebLogic Administration Server is started. On Windows instances, you must start the tool explicitly every time WebLogic Administration Server is started.

Rolling Back an AutoConfig Session

Each AutoConfig run creates a rollback script you can use to revert to the previous configuration settings if necessary. The script and backup configuration files from each AutoConfig session are stored in the following locations:

Tier	Directory
Application	<INST_TOP>/admin/out/<MMDDhhmm>
Database	<RDBMS_ORACLE_HOME>/appsutil/out/<CONTEXT_NAME><MDDhhmm>

where: <MMDDhhmm> = <month, day, hour, minute> of AutoConfig run.

Note: Rollback only needs to be performed in the event of a problem with an AutoConfig session.

To roll back an AutoConfig session, execute the following commands:

On UNIX:

```
$ restore.sh
```

On Windows:

```
C:\>restore.cmd
```

Customizing AutoConfig-Managed Configurations

AutoConfig simplifies and standardizes configuration management tasks in an Oracle E-Business Suite environment. At the start of each AutoConfig-managed file, you will see the following header:

```
################################################################
# Do not edit settings in this file manually. They are managed
# automatically and will be overwritten when AutoConfig runs.
# For more information about AutoConfig, refer to
# Oracle E-Business Suite Setup Guide.
################################################################
```

The configuration generated by AutoConfig may not always meet your specific requirements, however, and it may be necessary to customize AutoConfig for your environment.

Examples where you might want to customize AutoConfig include:

- Start additional services or processes when you start Oracle E-Business Suite services
- Extend Oracle Forms to integrate with a third party Java version
- Develop custom applications that are to be maintained by AutoConfig

Implementing AutoConfig Customizations

This section addresses the different types of AutoConfig customizations and how to implement them. After identifying your customization needs, perform the steps associated with them.

Oracle supports the following types of customization:

- Changing the value of an existing context variable
- Adding a new context variable to the context file
- Customizing an AutoConfig template file delivered by Oracle
- Creating a customer owned AutoConfig template file

Each of these will be considered in turn.

- **Changing the Value of an Existing Context Variable**

  1. **Edit the context variable value**

     Use the Oracle Applications Manager Context Editor to change values of existing context variables. Refer to *Oracle E-Business Suite Setup Guide* for details of how to use the Oracle Applications Manager Context Editor.

2. **Run AutoConfig**

   Depending on whether the context variable belongs to the application tier or the database tier, run AutoConfig on the appropriate tier, following the steps mentioned earlier in this chapter.

- **Adding a New Context Variable to the Context File**

  1. **Add the context variable**

     Use the Oracle Applications Manager Context Editor if you want to add a context variable that is not maintained by AutoConfig. Refer to the Help pages available on Oracle Applications Manager. The relevant information for adding custom context variables is located in the System Configuration > AutoConfig > Manage Custom Parameters section.

  2. **Run AutoConfig**

     Depending on whether the context variable belongs to the application tier or the database tier, run AutoConfig on the appropriate tier, following the steps mentioned earlier in this chapter.

- **Customizing an AutoConfig Template File Delivered by Oracle**

  If you want to customize an existing AutoConfig template file, perform the following steps in the order listed:

  1. **Determine the AutoConfig template file you want to customize**

     Execute the appropriate command below to identify the AutoConfig template file that corresponds to the configuration file you want to customize:

     UNIX

Tier	Command
Application	`<AD_TOP>/bin/adtmplreport.sh contextfile=<CONTEXT_FILE> target=<configurationfile>`
Database	`<RDBMS ORACLE_HOME>/appsutil/bin/adtmplreport.sh \ contextfile=<CONTEXT_FILE> target=<configurationfile>`

     Windows

Tier	Command
Application	<AD_TOP>\bin\adtmplreport.cmd contextfile=<CONTEXT_FILE> target=<configurationfile>
Database	<RDBMS ORACLE_HOME>\appsutil\bin\adtmplreport.cmd contextfile=<CONTEXT_FILE> target=<configurationfile>

For example, if you want to customize:

```
$INST_TOP/admin/install/afwebprf.sh
```

On UNIX:

```
$ $AD_TOP/bin/adtmplreport.sh contextfile=$CONTEXT_FILE \
target=$INST_TOP/admin/install/afwebprf.sql
```

On Windows:

```
C:\>%AD_TOP%\bin\adtmplreport.cmd contextfile=%CONTEXT_FILE%
target=%INST_TOP%\admin\install\afwebprf.sql
```

The adtmplreport utility returns the name and location of the AutoConfig template file. For the above UNIX example, it would return:

```
$INST_TOP/admin/install/afwebprf.sql
```

Be aware that you cannot customize all AutoConfig template files. A file cannot be customized if the keyword "LOCK" appears in the template file's entry in the product driver file. AutoConfig ignores custom template files that are marked with "LOCK". For example, the following entry in `<AD_TOP>/admin/driver/adtmpl.drv` will prevent customization of the file `adconfig.txt`:

```
ad admin/template adconfig.txt INSTE8 <s_at>/admin adconfig.txt
600 LOCK
```

2. **Create the Custom Template Directory**

   Create a directory named `custom` at the location where the AutoConfig template file resides.

   For example, if you want to customize `<FND_TOP>/admin/template/afwebprf.sql`, run the following command as the applmgr user:

   On UNIX:

   ```
   $ mkdir $FND_TOP/admin/template/custom
   ```

   On Windows:

   ```
   C:\>mkdir %FND_TOP%\admin\template\custom
   ```

3. **Copy the AutoConfig template File**

Copy the AutoConfig template file to the custom template file. Run the following command as the applmgr user:

On UNIX:

```
$ cp -i <AutoConfig template file> <Custom template file>
```

For example:

```
$ cp -i $FND_TOP/admin/template/afwebprf.sql \
$FND_TOP/admin/template/custom/afwebprf.sql
```

On Windows:

```
C:\>copy <AutoConfig template file> <Custom template file>
```

For example:

```
C:\> copy %FND_TOP%\admin\template\afwebprf.sql
%FND_TOP%\admin\template\custom\afwebprf.sql
```

4. **Edit the Custom Template File**

Edit the custom template file with the text editor of your choice, such as vi on UNIX or Wordpad on Windows (do not use Word or any other editor that introduces non-printing characters).

> **Warning:** It is not supported to edit *AutoConfig* template files. Be sure to edit only *custom* template files.

5. **Verify Your Customizations**

Execute the `adchkcfg` utility as described earlier. When this utility runs, it instantiates any custom template files in place of the corresponding AutoConfig template file. The utility generates a report with information about all files and profile options that will be changed during the next normal run of AutoConfig. Verify that your customizations will be applied as expected in your next AutoConfig run.

The `adchkcfg.sh` (on UNIX) and `adchkcfg.cmd` (on Windows) scripts instantiate the templates into the following location:

Tier	Directory
Application	`<INST_TOP>/admin/log/<MMDDhhmm>`
Database	`<RDBMS_ORACLE_HOME>/appsutil/log/<CONTEXT_NAME>/<MMDDhhmm>`

6. **Run AutoConfig**

Run AutoConfig as described earlier. When AutoConfig runs, it instantiates any custom template files in place of the corresponding AutoConfig template files.

- **Creating a Customer-Owned AutoConfig Template File**

  You can create an AutoConfig template file of your own. This will enable you to develop custom applications that AutoConfig can configure and maintain. Perform the following steps in the order listed:

  1. **Define a product_top**

     Use the Oracle Applications Manager Context Editor to add your custom product_top to the context file. Refer to the Help pages available in Oracle Applications Manager. The relevant information for adding custom context variables is located in the "System Configuration" > "AutoConfig" > "Manage Custom Parameters" section.

     > **Note:** There is a known issue with using the Oracle Applications Manager Context Editor to add your custom product_top to the context file. See the Known Issues, *Oracle E-Business Suite Developer's Guide* for more information.

     Choose the variable type "PROD_TOP" when adding the product_top variable. The default value for the product_top variable must be set to `%s_at%/<product name>/12.0.0`.

     For example, you could define the product top for the product 'my' as follows:

     ```
     OA_VAR = c_mytop
      Default Value = %s_at%/my/12.0.0
      Title = My Product top
      Description = This is my product top
      OA_TYPE = PROD_TOP
     ```

     > **Note:** The string "%s_at%" should not be replaced. This string must remain exactly as %s_at% so AutoConfig can replace it later with the corresponding APPL_TOP directory structure for both run and patch edition file systems.
     >
     > Context files for both run and patch file systems will be updated at the same time after you saving. When you create the product_top variable and save your changes, the system will create that variable in both context files.

     > **Note:** When you choose the PROD_TOP radio button, the default value is prepended with `%s_at%/` if you haven't entered that in already. Enter the PROD_TOP variable relative

to the APPL_TOP variable %s_at%

2. **Create the customer-owned AutoConfig template directory**

   Create the directory where your AutoConfig template files will reside. Run the applicable command as the applmgr user:

   On UNIX, create a directory with a command such as:

   ```
   $ mkdir <c_mytop>/admin/template
   ```

   On Windows, create a directory with a command such as:

   ```
   C:\>mkdir <c_mytop>\admin\template
   ```

3. **Develop the customer-owned AutoConfig template file**

   Create the custom AutoConfig template file in the custom product top AutoConfig template directory. There are no file name restrictions, and the new template file can be of any type that can store text, such as text file, shell script, Perl script, or SQL script. To use AutoConfig instantiation, enter your context variables in the file. When AutoConfig runs, it replaces the context variables with the associated values from the context file.

   On UNIX, create a file such as:

   ```
   <c_mytop>/admin/template/myTemplate.txt
   ```

   On Windows, create a file such as:

   ```
   <c_mytop>\admin\template\myTemplate.txt
   ```

4. **Create the customer-owned AutoConfig driver directory**

   Create the directory where your AutoConfig driver file will reside. Run the following command as the applmgr user:

   On UNIX:

   ```
   $ mkdir <c_mytop>/admin/driver
   ```

   On Windows:

   ```
   C:\>mkdir <c_mytop>\admin\driver
   ```

5. **Develop a customer-owned AutoConfig driver file**

   Every file you want AutoConfig to instantiate needs an entry in an *AutoConfig driver file*, which instructs AutoConfig where to place a generated configuration file.

   The driver file must be placed in the custom AutoConfig driver directory. The name for the driver file is defined as `<product>tmpl.drv`. Driver file entries consist of <TAB> or <SPACE> separated fields.

   The following table lists the fields and their contents.

AutoConfig Driver File Format

Field Name	Description
Product Name	Specifies the short name for the product
AutoConfig template directory	Directory underneath the product top that hosts the AutoConfig template file
AutoConfig template file	Name of the template file to be processed by AutoConfig

Field Name	Description
Action	Action that AutoConfig performs on the AutoConfig template file. Possible values are: • **INSTE8**: AutoConfig instantiates the template each time it is run. • **INSTE8_SETUP**: In addition to instantiating, execute the resulting configuration file during the SETUP Phase of each run of AutoConfig. • **INSTE8_PRF**: In addition to instantiating, execute the resulting configuration file during the PROFILE Phase of each run of AutoConfig. • **INSTE8_APPLY**: In addition to instantiating, execute the resulting configuration file during the APPLY Phase of each run of AutoConfig. • **INSTALL**: AutoConfig instantiates the template file only if the resulting configuration file does not already exist. • **INSTALL_SETUP**: In addition to instantiating, AutoConfig executes the resulting configuration file during the SETUP phase if the configuration file does not already exist. • **INSTALL_PRF**: In addition to instantiating, AutoConfig executes the resulting configuration file during the PROFILE phase if the configuration file does not already exist. • **INSTALL_APPLY**: In addition to instantiating, AutoConfig executes

Field Name	Description
	the resulting configuration file during the APPLY phase if the configuration file does not already exist.
Configuration directory	AutoConfig places the instantiated configuration file in this directory.
Configuration file	Name of the instantiated configuration file.
Configuration file permission (UNIX only)	AutoConfig generates the configuration file with the provided UNIX-style permissions.

For example, a driver file entry might contain the following (on one line):

```
my admin/template myTemplate.txt INSTE8 <s_pt>
myConfiguration.txt 660
```

In this example, AutoConfig would instantiate the template file `<MY_TOP>/admin/template/myTemplate.txt` and generate the configuration file `myConfiguration.txt` into the Portal directory (the Portal directory is instantiated from `<s_pt>`) with permissions of 660 (read and write for user and group).

To have AutoConfig instantiate the above example template file, the driver file would need to contain the line:

```
my admin/template myTemplate.txt INSTE8 <s_pt>
myConfiguration.txt 660
```

Note: This should be entered on a single line.

In this example, the driver file would be named `mytmpl.drv`.

Advanced Features of AutoConfig Customizations

This section discusses advanced features available when using AutoConfig Customizations.

- **Debugging customizations**

  If problems arise with customizations that you implemented, it may be useful to run AutoConfig with the AutoConfig template files, ignoring any custom template

files. Run the following command:

On UNIX:

```
$ <AD_TOP>/bin/adconfig.sh -nocustom contextfile=<CONTEXT_FILE>
```

On Windows:

```
C:\><AD_TOP>\bin\adconfig.cmd -nocustom contextfile=<CONTEXT_FILE>
```

- **Preserving customizations after updates**

  You must review your customizations whenever a TXK patch delivers a new version of an AutoConfig template file for which you edited the corresponding custom template file. If the customizations are still required, copy the new version of the AutoConfig template file to the custom template directory, and edit the custom template file with your customizations.

  > **Important:** AutoConfig checks that your custom template files are of the same versions as the AutoConfig template files, and will not run if it detects any version mismatch.

- **Using Reports**

  The report produced by the `adtmplreport` utility can:

  - List all customized files in an Oracle E-Business Suite instance.
  - List all AutoConfig template files, their custom template files and their configuration files.
  - Identify the name and location of the AutoConfig template file and the custom template file for a given configuration file.
  - Identify the name and location of the configuration file for a given AutoConfig template file.

  The report utility is located at:

Platform	Command
UNIX	`<AD_TOP>/bin`
Windows	`<RDBMS ORACLE_HOME>/appsutil/bin`

  To list all files that you customized in an Oracle E-Business Suite instance:

Platform	Command
UNIX	adtmplreport.sh contextfile=<CONTEXT_FILE> listcustom
Windows	adtmplreport.cmd contextfile=<CONTEXT_FILE> listcustom

To list all configuration files, their AutoConfig template files and their custom template files:

Platform	Command
UNIX	adtmplreport.sh contextfile=<CONTEXT_FILE>
Windows	adtmplreport.cmd contextfile=<CONTEXT_FILE>

To identify the configuration file for a given AutoConfig template file:

Platform	Command
UNIX	adtmplreport.sh contextfile=<CONTEXT_FILE> \ template=<templatefilepath>
Windows	adtmplreport.cmd contextfile=<CONTEXT_FILE> \ template=<templatefilepath>

To identify the AutoConfig template and custom template file for a given configuration file:

Platform	Command
UNIX	adtmplreport.sh contextfile=<CONTEXT_FILE> target=<configurationfile>
Windows	adtmplreport.cmd contextfile=<CONTEXT_FILE> target=<configurationfile>

AutoConfig Customization Limitations and Restrictions

This section discusses the limitations and restrictions that exist when using AutoConfig customizations.

- **Limitations with customizing tnsnames.ora, listener.ora and sqlnet.ora**

  When running AutoConfig on the application tier, it generates tnsnames.ora, listener.ora and sqlnet.ora files based on the information present in the FND_NET_SERVICES table. If you need to add custom parameters, you must use the IFILE feature of these configuration files. For example, if there is a need to add TRACE_DIRECTORY to tnsnames.ora, you must add this to the file pointed to by the IFILE parameter rather than adding it directly to the tnsnames.ora file.

  > **Note:** It is not supported to customize `admk80ln_ux.sql` or any other file involved in the generation of these files.

Patching AutoConfig

In Release Oracle E-Business Suite Release 12.2, AutoConfig must be enabled on both the application tier and the database tier if it is to be patched.

Applying the Latest AutoConfig Updates

To obtain the latest AutoConfig updates on both the applications tier and the database tier, perform the following steps in the order listed.

1. **Copy AutoConfig to the RDBMS ORACLE_HOME**

   Update the RDBMS ORACLE_HOME file system with the new AutoConfig files by performing the following steps:

   1. On the application tier (as the APPLMGR user).

      1. Log in to the APPL_TOP environment and source the environment file.

      2. Create the appsutil.zip file by running the following command:

         ```
         $ perl <AD_TOP>/bin/admkappsutil.pl
         ```

         This will create appsutil.zip in <INST_TOP>/admin/out.

   2. On the database tier (as the ORACLE user):

      1. Copy or FTP the appsutil.zip file to the RDBMS ORACLE_HOME, then run the following commands:

```
$ cd <RDBMS ORACLE_HOME>
$ unzip -o appsutil.zip
```

2. **Run AutoConfig**

   Run AutoConfig on the database tier, and then on the applications tier.

Enabling AutoConfig on a New Oracle Home

In Release 12.2, AutoConfig is enabled by default on the application tier. However, it might not be enabled on the database tier in the following scenarios:

- The database tier was not created by Rapid Install.
- Cross-platform migration has been performed on the database tier.
- The database has been upgraded to Oracle Database 11g.
- The database tier has been upgraded as part of an Oracle E-Business Suite upgrade from Release 11*i* to 12.2.

To enable AutoConfig on the database tier, perform the following steps in the order listed:

1. **Copy AutoConfig to the RDBMS ORACLE_HOME**

   Update the RDBMS ORACLE_HOME file system by following the steps in *Applying the Latest AutoConfig Updates* above.

2. **Install Java Runtime Environment (JRE) on the Database tier**

   The JRE resides in the `<ORACLE_HOME>/appsutil/jre` directory on the database tier. Ensure that the JRE version on the database tier is at least at:

   - **Solaris**: Java SE Runtime Environment (build 1.6.0_21-b06)
   - **Linux x86-64**: Java SE Runtime Environment (build 1.6.0_17-b04)

   You can obtain the JRE from the official download location [http://www.oracle.com/technetwork/java/javase/downloads/index.html].

   > **Warning:** Do **not** download the Java SE Development Kit (JDK). Instead, download the JRE that supports 64-bit JVM. For additional information on installation, see *Using Latest Java 6.0 Update With Oracle E-Business Suite Release 12* (Doc ID 455492.1), disregarding the note about downloading 32-bit JRE.

3. **Generate the Database Context File**

Execute the following command to create your database context file:

```
$ perl <RDBMS_ORACLE_HOME>/appsutil/bin/adbldxml.pl
```

> **Important:** If you run the adbldxml.pl utility for an instance that is part of an Oracle RAC environment, all the Oracle RAC instances must be running so that adbldxml.pl can connect to them and gather information about the configuration.

4. **Run AutoConfig on the Database tier**

   Run AutoConfig on the database tier by executing one of the following commands:

   On UNIX:

   ```
   $ <RDBMS_ORACLE_HOME>/appsutil/bin/adconfig.sh \
   contextfile=<context_file>
   ```

   On Windows:

   ```
   C:\><RDBMS_ORACLE_HOME>\appsutil\bin\adconfig.cmd
   contextfile=<context_file>
   ```

Advanced AutoConfig Features and Additional Utilities

This section gives an overview of some of the advanced AutoConfig features and utilities.

Running AutoConfig in Parallel Across Multiple Nodes

This feature enables AutoConfig to be executed simultaneously across multiple nodes of an Oracle E-Business Suite Release 12.2 instance. AutoConfig can be run in this parallel mode by executing the following command:

- On the Application tier:

  ```
  $ perl <AD_TOP>/bin/adconfig.pl contextfile=<CONTEXT_FILE> \
  [product=<product_top>] -parallel
  ```

- On the Database tier:

  ```
  $ perl <ORACLE_HOME>/appsutil/bin/adconfig.pl \
  contextfile=<CONTEXT_FILE> -parallel
  ```

> **Important:** When running AutoConfig simultaneously on multiple nodes, the `-parallel` option must be specified when running AutoConfig on each node. If it is not, the execution of AutoConfig processes on individual nodes will not be synchronized, and possibly result in inconsistent filesystem or database.

Profiling An AutoConfig Run

The AutoConfig Performance Profiler feature can be used to profile an AutoConfig run and generate a consolidated report in HTML format. The report displays a summarized view listing all the product tops along with the total instantiation/execution time of the templates within them. The profile report comprises the following sections:

- **Summary**

  This section of the report shows the profile information for all product tops processed in the current AutoConfig run. It shows the following:

  - **Product Top**: Short name of each product top.
  - **Instantiation Time**: Total time taken to instantiate templates from each product top.
  - **Execution Time**: Total time taken to execute scripts from each product top.
  - **Time (%)**: Percentage of AutoConfig execution time taken to instantiate and execute scripts from each product top.
  - **Status**: Whether or not all the templates from each product top were successfully instantiated and executed.

  The profile information for individual templates can be seen by drilling down into the product tops listed in the summary section.

- **Details**

  This section contains the profile information for all product templates instantiated and executed in the current AutoConfig run. It shows the following:

  - **Script Name**: Target name of the template.
  - **Instantiation Time**: Time taken to instantiate the template.
  - **Execution Time**: Time taken to execute the instantiated template.
  - **Time (%)**: Percentage of product top processing time taken to process the template.
  - **Status**: Whether or not the template was successfully processed.
  - **Execution Summary**: Contains the source and target locations of the template and the execution report of the script. This summary can be viewed by clicking on the script name link in the detailed report.

AutoConfig can be run in profile mode by issuing the following commands:

- On the Application Tier:

```
$ perl <AD_TOP>/bin/adconfig.pl contextfile=<CONTEXT_FILE> \
[product=<product_top>] -profile
```

- On the Database Tier:

```
$ perl <ORACLE_HOME>/appsutil/bin/adconfig.pl \
contextfile=<CONTEXT_FILE> [product=<product_top>] -profile
```

Using the Check Config Utility

The Check Config utility (adchkcfg) is used to review the configuration changes that will take effect on an Oracle E-Business Suite instance during the next AutoConfig run. It identifies the potential changes to both the file system as well as the database. It can be run on both the application tier and the database tier.

The utility is located in the following location:

Tier	Location
Application	<AD_TOP>/bin
Database	<ORACLE_HOME>/appsutil/bin

The Check Config utility is run by executing the following command:

- On UNIX:

```
$ adchkcfg.sh contextfile=<CONTEXT_FILE>
```

- On Windows:

```
C:\>adchkcfg.cmd contextfile=<CONTEXT_FILE>
```

This script generates both HTML and text reports. The reports provide information about all file changes, profile option changes and other important database updates that will be done during the next normal execution of AutoConfig. Information is organized under the following two tabs:

- **File System Changes**

  This report is divided into the following sections:

  - **AutoConfig Context File Changes**: Displays information about the location of the context file, the content of the currently active context file, the content of the context file that will be generated in the next AutoConfig run. In addition it also displays an HTML report highlighting the differences between the current and the new context file, if any.

- **Service Group Status**: Displays the status of the Service Groups and the corresponding services on the applications tier. This section is present only in the report generated on the applications tier.

- **Changed Configuration Files**: Displays a list of all the files that will be changed during an AutoConfig execution. For each file, information is displayed about the location of the runtime file, the content of the currently active file, the content of the file that will be generated in the next AutoConfig run. In addition, an HTML report highlights the differences between the current and the new configuration file, plus the location of the AutoConfig template file (if applicable).

- **New Configuration Files**: Displays a list of all the new files that will be created during an AutoConfig execution. For each file, information is displayed about the location of the runtime file, the content of the new file and the location of the AutoConfig template file.

- **Template Customizations**: Displays a list of all customized AutoConfig templates and reports the difference between the original AutoConfig template and the customized template.

- **Database Changes**

  This report is divided into the following sections:

  - **Profile Value Changes**: Displays details of only those profiles whose value would be changed in the next AutoConfig run. For each such profile, the current value in the Database, the new AutoConfig value that would be set for it, the Profile Level and the name of the AutoConfig script that changes the profile value is displayed.

  - **Profile Values**: Displays details as in previous section for all Apps Database profiles managed by AutoConfig, irrespective of whether their value would change or not in the next AutoConfig run.

  - **Other Database updates**: Displays details for important database updates (non-profile changes) that will be performed in the next AutoConfig run. The table name, column name, current column value in the database, and new AutoConfig value are displayed, along with the name of the updating AutoConfig script and a brief description.

The script also creates a zip file report, ADXcfgcheck.zip, which contains all the files and reports mentioned above. The ADXcfgcheck.zip file can be copied to a local client device, and the HTML report viewed there without breaking the hyperlinks in the report.

Using the Context Variable Information Utility

This command line utility can be used to find out detailed information about context variables and the templates in which they are used. The utility accepts all or part of a context variable name and generates an HTML or text report containing information about the matched context variables, including description, default vale, and current value. The variable description contains recommended settings, range of allowed values, and links to documents that provide detailed usage information. Additionally, the utility lists the configuration templates where the respective context variables are used.

After the Application tier environment file has been sourced, you can run the Context Variable Information utility as follows:

- On the Application tier:

```
$ perl <FND_TOP>/bin/txkrun.pl -script=GenCtxInfRep \
-keyword="<keyword>"
```

- On the Database tier:

```
$ perl <ORACLE_HOME>/appsutil/bin/txkrun.pl \
-script=GenCtxInfRep -keyword="<keyword>"
```

The utility takes the following arguments:

- **contextfile (optional)**: Complete path to the context file. By default, it is set to the value of <CONTEXT_FILE!>
- **keyword (required)**: All or part of a context variable name.
- **reporttype (optional)**: The report type. Valid values are 'html' (the default) and 'text'.
- **outfile (required)**: The report file. If only the name of the report file is provided, and not the complete path, the report will be generated in the <APPLTMP> directory.

Using AutoConfig on an Oracle RAC Instance

This section guides you through the steps that need to be performed when your Oracle Release 12.2 instance is running in an Oracle Real Application Clusters (Oracle RAC) environment.

Oracle E-Business Suite Release 12.2 delivers the infrastructure to generate a complete tnsnames.ora file required for Oracle RAC. This includes:

- Instance aliases for each database tier node.
- Load balance aliases with address lists for each database tier node.

- FNDSM and FNDFS aliases (used by the CP Service Manager) for each application tier node.

- Virtual hostname support.

The tnsnames.ora file is dynamically generated using the Net Services Topology Data Model. The Net Services Topology Data Model stores the entire topological information about a single Oracle E-Business Suite environment.

Complete the steps below to support AutoConfig on Oracle RAC:

1. **Review init.ora**: AutoConfig will not overwrite your existing init.ora file. However, when no init.ora file exists, AutoConfig will generate an init.ora that is compatible with Oracle RAC. It is advisable to create a backup of the existing init.ora file, and let AutoConfig generate a new init.ora file. This will ensure that the init.ora file conforms to Oracle's standards: for example, use of DB_Name as the service name, and handling of local and remote listeners.

2. **Migrate AutoConfig Patch to Database tier**: Follow the steps given above to migrate the AutoConfig Patch to the database tier.

3. **Run AutoConfig on all Database tier nodes**: Run AutoConfig on all database tier nodes, following the instructions given earlier.

4. **Run AutoConfig on Application tier**: Run AutoConfig on each Application tier node. Use the adautocfg.sh (or adautocfg.cmd) command described earlier.

5. **Restart the database listener**: Stop and restart your database listener.

Your system is now AutoConfig-enabled with Oracle RAC, and the system configuration can be managed as described earlier.

Managing Oracle HTTP Server Configurations

Prior to Oracle E-Business Suite Release 12.2, AutoConfig managed all Oracle HTTP Server configuration files throughout the system lifecycle. In Release 12.2, AutoConfig is only involved in the initial setup of the Oracle HTTP Server configuration files.

Later, it can optionally be used to manage and customize a limited set of Oracle HTTP Server configuration files. Otherwise, as described later in this section, native Oracle WebLogic Server and Fusion Middleware tools are used to manage these files.

Updating AutoConfig-Managed Oracle HTTP Server Configurations

The table below shows the Oracle HTTP Server configuration templates that are managed by AutoConfig in Oracle E-Business Suite Release 12.2. The configurations defined in these files can be customized by customizing the templates.

Template Name	Configuration File
ssl_terminator_conf_FMW.tmp	ssl_terminator.conf
trusted_conf_FMW.tmp	trusted.conf
oracle_apache_conf_FMW.tmp	oracle_apache.conf
url_fw_conf_FMW.tmp	url_fw.conf
url_fw_ws_conf_FMW.tmp	url_fw_ws.conf
security_dmz_conf_FMW.tmp	security_dmz.conf
custom_conf_FMW.tmp	custom.conf

Updating Seeded Oracle HTTP Server Configurations

During creation of the Oracle HTTP Server instance, AutoConfig performs the first instantiation of configuration files such as httpd.conf and mod_wl_ohs. After the installation or upgrade to Release 12.2 is complete, seeded Oracle HTTP Server configurations are generally managed via Fusion Middleware Control. In certain specific cases, however, manual editing of the native configuration files may be needed: this will be advised separately where required.

> **Note:** For further information, refer to *3.3: Getting Started Using Oracle Enterprise Manager Fusion Middleware Control* in *Oracle Fusion Middleware Administrator's Guide*. This book can be found in the Oracle Fusion Middleware Documentation Library [http://docs.oracle.com/cd/E23943_01/index.htm].

When modifying Oracle HTTP Server protocols or port numbers, these values must be updated in both the context file as well as in the HTTP configuration files. This is to ensure that the new values are updated in the AutoConfig-managed database profile values as well.

To modify HTTP server protocols or port numbers, perform the following steps:

1. Edit the HTTP server configuration parameter from the Oracle Enterprise Manager Console.

2. Run the following command on all application tier nodes:

```
$ perl <AD_TOP>/bin/adSyncContext.pl contextfile=<CONTEXT_FILE>
```

3. Run AutoConfig on all application tier nodes.

Running Oracle HTTP Server on a Privileged Port

On a UNIX system the TCP/IP port numbers below 1024 are special in that only processes with root privileges are allowed to listen on those ports. If you configure Oracle HTTP Server to run on a privileged port, then you must perform additional steps to enable Oracle HTTP Server to run as root. These steps are required for both SSL and non-SSL privileged ports.

1. Execute the following commands as the root user on both the run file system and the patch file system.

```
chown root $FMW_HOME/webtier/ohs/bin/.apachectl
chmod 6750 $FMW_HOME/webtier/ohs/bin/.apachectl
```

2. Restart HTTP services by running `$ADMIN_SCRIPTS_HOME/adapcctl.sh` on the run file system.

You must also perform additional steps during the online patching cycle when you are running Oracle HTTP Server on a privileged port. See: Requirements When Running Oracle HTTP Server on a Privileged Port, *Oracle E-Business Suite Maintenance Guide*.

For more information, see: Starting Oracle HTTP Server on a Privileged Port, *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server*.

Managing Configuration of Oacore, Oafm, Forms and Forms-c4ws Applications

In Oracle E-Business Suite Release 12, the oacore, oafm, forms and forms-c4ws services were OC4J instances and were managed by Oracle Process Manager (OPMN). Because Oracle WebLogic Server has replaced OC4J in Oracle E-Business Suite Release 12.2, these services are now deployed as applications on individual managed servers.

Consequently, only part of the configuration of these applications and managed servers is still managed via AutoConfig. This section describes those areas that remain managed by AutoConfig, and those that do not.

Updating Configuration Parameters for Individual Applications

The basic configurations of the oacore, oafm, forms and forms-c4ws applications are maintained in their respective deployment plans. The deployment plan contains configurable parameters for each application, such as session timeout, log file rotation size, and log file rotation time. Each of these deployment plans is delivered as an AutoConfig template, containing a very limited set of context variables.

The deployment plan is instantiated initially by AutoConfig. Subsequently, AutoConfig only updates the existing deployment plan: this is needed in case any of the context

variables used in the deployment plan have been customized.

The only parameters that can be updated or changed using AutoConfig are the following parameters in the deployment plan for oacore:

Parameter	Value
help_InitParam_ohwConfigFileURL	`file:%s_ora_config_home%/FMW/oacore/config/ohwconfig.xml`
CGIServlet_InitParam_cgiDir	`%s_config_home%/admin/install`
CGIServlet_InitParam_*.pl	`%s_adperlprg%`
oowa_InitParam_log_main_file	`%s_logs_dir%/ora/FMW/oacore/oowa.log`
oowa_InitParam_log_spl2_file	`%s_logs_dir%/ora/FMW/oacore/oowa.log`
LoopProcServlet_InitParam_ARCHIVE	`%s_applptmp%`
TransmitServlet_InitParam_ARCHIVE	`%s_applptmp%`

The remaining parameters for each application need to be updated through the WebLogic Server Administration Console, as follows:

1. Log on to the WebLogic Server Administration Console.
2. Click on the 'Deployments' link in the left panel for 'Domain Structure'.
3. Examine the page that gives a summary of all deployments.
4. If the configuration is to be customized at the application level, click on the application whose configuration is to be updated. For example, if updating oacore configuration, click on the link for 'oacore' and take the desired action on the page that allows editing the settings of the application.
5. If the configuration changes are to be done at the Web Application level, click on the link for the respective Web Application in the Deployments page, and select the 'Configuration' tab to customize the configuration parameters for the application.
6. After editing and setting the parameters, click on the 'Save' button. This updates the deployment plan for the application.
7. Once all configuration changes have been saved, click on the 'Activate Changes' button in the 'Change Center' panel to activate the changes.

Note: For further information, refer to *3.4: Getting Started Using Oracle WebLogic Server Administration Console* in *Oracle Fusion Middleware Administrator's Guide 11g Release 1 (11.1.1)*. This book can be found in the Oracle Fusion Middleware Documentation Library [http://docs.oracle.com/cd/E23943_01/index.htm]

Managing Configuration Parameters for Individual Managed Servers

The managed server configuration can be customized via native WebLogic tools like the WebLogic Server Administration Console and WLST commands.

Use the following steps to customize the managed server configuration via the WebLogic Server Administration Console.

1. Log on to the WebLogic Server Administration Console.
2. Click on the "Servers" link. This link takes you to a page containing a summary of the WebLogic Administration Server and all managed servers.
3. Click on the managed server whose configuration needs to be updated. A page containing various tabs for the settings of the managed server appears.
4. Update the configuration parameters as needed.
5. Click the *Save* button to save the configuration changes.
6. Once the customizations are complete and saved, click the *Activate Changes* button in the 'Change Center' panel to activate the changes.

Note: If the managed server port number is to be changed, then, in addition to updating the port number via the WebLogic native tools, also perform the additional steps in the section Changing the port numbers of the managed servers, page 3-40.

Managing classpath and JVM arguments for the managed server

The classpath and JVM arguments of a managed server can be modified from the WebLogic Server Administration Console or through WLST commands as specified above.

In addition, these properties can also be set from the backend as follows:

- To set the managed server classpath, execute the following command:

```
$ perl <AD_TOP>/patch/115/bin/adProvisionEBS.pl
ebs-set-managedsrvproperty \
-contextfile=<CONTEXT_FILE> -managedsrvname <MANAGED SERVER NAME> \
-managedsrvclasspath "<MANAGED SERVER CLASSPATH TO BE SET>"
```

- To set the managed server JVM arguments, execute the following command:

```
$ perl <AD_TOP>/patch/115/bin/adProvisionEBS.pl
ebs-set-managedsrvproperty  \
-contextfile=<CONTEXT_FILE> -managedsrvname <MANAGED SERVER NAME> \
-serverstartargs "<JVM arguments to be set>"
```

During startup of a managed server using the adstrtal.sh/cmd script or the individual control script admanagedsrvctl.sh/cmd, the Oracle E-Business Suite-specific Library Path is updated in the managed server JVM arguments. In case the JVM arguments are already customized for the managed server, the EBS-specific Library Path is prepended to the list of customized library paths.

Resetting managed server classpath and JVM arguments to default values

At any point of time, if it is desired to reset the managed server classpath or JVM arguments to the default values, this can be done by referring to the default values from the context file and then updating these properties using native WebLogic tools or using the adProvisionEBS.pl script as mentioned above.

The default value for the managed server classpath can be referred to from the context variables listed in the following table:

Service Type	Context Variable
oacore	s_oacore_nodes
oafm	s_oafm_nodes
forms	s_forms_nodes
forms-c4ws	s_forms-c4ws_nodes

The default value for the managed server JVM arguments can be referred to from the following context variables:

Service Type	Context Variable
oacore	s_oacore_jvm_start_options
oafm	s_oafm_jvm_start_options

Service Type	Context Variable
forms	s_forms_jvm_start_options
forms-c4ws	s_forms-c4ws_jvm_start_options

Changing the port numbers of the managed servers

To change the port number of a managed server, you must first edit the port number by following the steps in the above subsection, "Managing other configuration parameters for managed servers". Then perform the following steps to edit the port numbers for the managed server in the Oracle HTTP Server configuration files.

1. In the Application tier context files of all nodes of the Oracle E-Business Suite Release 12.2 instance, edit the node-related context variables corresponding to the particular managed server type to replace the old port number with the new port number. This step is done automatically if the adRegisterWLSListeners.pl script has been running on the primary node.

   Shown below is a table with the list of context variables to be updated for each managed server type:

Service Type	Context Variable
oacore	s_oacore_nodes
oafm	s_oafm_nodes
forms	s_forms_nodes
forms-c4ws	s_forms-c4ws_nodes

2. On the node containing the Oracle HTTP Server, run AutoConfig. Then execute the following script to update the Oracle HTTP Server configuration files with the new managed server port number:

   ```
   perl <FND_TOP>/patch/115/bin/txkSetAppsConf.pl \
   -ctxfile=$CONTEXT_FILE -outfile=<OUT_FILE>
   ```

3. On the node that contains the Oracle HTTP Server, restart the Oracle HTTP Server as follows:

   On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adapcctl.sh stop
$ sh <ADMIN_SCRIPTS_HOME>/adapcctl.sh start
```

On Windows:

```
C:\><ADMIN_SCRIPTS_HOME>/adapcctl.cmd stop
C:\><ADMIN_SCRIPTS_HOME>/adapcctl.cmd start
```

Customizing the number of instances of a particular service type

By default, every Application tier node contains only a single instance of the oacore, oafm, forms and forms-c4ws services. If more instances of a particular service are required on an Application tier node, new managed servers must be created. Similarly, if the number of instances of a particular service needs to be reduced, the respective managed server needs to be deleted.

> **Note:** For the oacore service, Oracle recommends up to 150-200 concurrent users per JVM of size 2 GB.

Adding a new Managed Server

Addition of managed servers needs to be done separately for both the RUN file system and the PATCH file system.

To add a new managed server of a specific service type, perform the following steps on the RUN file system:

1. Execute the following command to add a new managed server. This will create a managed server, deploy it into the cluster of the respective service, and add a new entry to the context file for starting and stopping the new managed server via the adstrtal and adstpall scripts:

   ```
   $ perl <AD_TOP>/patch/115/bin/adProvisionEBS.pl \
   ebs-create-managedserver -contextfile=<CONTEXT_FILE> \
   -managedsrvname=<MANAGED_SERVER_NAME> -servicetype=<SERVICE_TYPE> \
   -managedsrvport=<managed_server_port> -logfile=<LOGFILE>
   ```

   For example, to add a managed server 'oacore_server2' of type 'oacore', run the following command:

   ```
   $ perl <AD_TOP>/patch/115/bin/adProvisionEBS.pl \
   ebs-create-managedserver -contextfile=<CONTEXT_FILE> \
   -managedsrvname=oacore_server2 -servicetype=oacore \
   -managedsrvport=<oacore_port> -logfile=<LOGFILE>
   ```

   > **Note:** The managed server name provided must be of the form `<SERVICE_TYPE>_server<n>`, where n is an integer.

   > **Note:** While specifying the managed server port number, ensure that the port is free. The managed server port number should be

distinct for each managed server. No two managed servers on the RUN file system and the PATCH file system should have the same value.

2. After addition of all managed servers, run AutoConfig on all Application tier nodes.

3. On the node containing the Oracle HTTP Server, execute the following script to update the Oracle HTTP Server configuration files with the new managed server port number:

```
$ perl <FND_TOP>/patch/115/bin/txkSetAppsConf.pl \
-ctxfile=$CONTEXT_FILE -outfile=<OUT_FILE>
```

4. On the node that contains the Oracle HTTP Server, restart the Oracle HTTP Server as follows:

   On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adapcctl.sh stop
$ sh <ADMIN_SCRIPTS_HOME>/adapcctl.sh start
```

   On Windows:

```
C:\><ADMIN_SCRIPTS_HOME>/adapcct.cmd stop
C:\><ADMIN_SCRIPTS_HOME>/adapcctl.cmd start
```

Once the managed server addition is completed on the RUN file system, perform the following steps to add the managed servers to the PATCH file system.

1. Start up the WebLogic Administration Server of the PATCH file system by executing the following command on the PATCH file system of the primary node.

   On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adadminsrvctl.sh start forcepatchfs
```

   On Windows:

```
C:\> <ADMIN_SCRIPTS_HOME>\adadminsrvctl.cmd start
```

2. Execute the following command **from the RUN file system** to add the new managed server.

```
$ perl <AD_TOP>/patch/115/bin/adProvisionEBS.pl
ebs-create-managedserver \
-contextfile=<PATCH_CONTEXT_FILE>
-managedsrvname=<MANAGED_SERVER_NAME> \
-servicetype=<SERVICE_TYPE> -managedsrvport=<managed_server_port> \
-logfile=<LOGFILE>
```

3. After addition of all managed servers on the PATCH file system, run adSyncContext.pl **from the RUN file system** on all Application tier nodes, passing the context file of the PATCH file system.

```
$ perl <AD_TOP>/bin/adSyncContext.pl
-contextfile=<PATCH_CONTEXT_FILE>
```

4. Instantiate the fsclone_config.txt file on the PATCH file system of all Application tier nodes by running the following command after sourcing the RUN file system:

```
java oracle.apps.ad.autoconfig.InstantiateFile -e
<PATCH_CONTEXT_FILE> -tmpl  \
<PATCH_AD_TOP>/admin/template/fsclone_config_txt.tmp -out  \
<PATCH_INST_TOP>/appl/admin/fsclone_config.txt
```

5. On the node containing the Oracle HTTP Server, execute the following script **from the RUN file system**, passing the context file from the PATCH file system. This will update the Oracle HTTP Server configuration files on the PATCH file system with the new managed server port number.

```
$ PERL <FND_TOP>/patch/115/bin/txkSetAppsConf.pl
-ctxfile=<PATCH_CONTEXT_FILE> \
-outfile=<OUT_FILE>
```

6. Shut down the WebLogic Administration Server of the PATCH file system by executing the following command on the PATCH file system of the primary node.

   On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adadminsrvctl.sh stop
```

   On Windows:

```
C:\> <ADMIN_SCRIPTS_HOME>\adadminsrvctl.cmd stop
```

> **Note:** Managed server creation should be done only through the adProvisionEBS.pl script as mentioned above. Managed servers should not be created from the WebLogic Server Administration Console.

Removing a Managed Server

The following steps should be executed in order to remove/delete a managed server from the domain. As in case of managed server addition, deletion of managed servers also need to be done explicitly on the RUN and PATCH file systems.

In order to delete a managed server of a specific service type from the RUN file system, perform the following on the RUN file system.

1. Execute the below command on the Application tier node where the managed server resides. This will delete the managed server and also update the respective context variables that contain references to the deleted managed server.

```
$ perl <AD_TOP>/patch/115/bin/adProvisionEBS.pl
ebs-delete-managedserver \
-contextfile=<CONTEXT_FILE> -managedsrvname=<MANAGED_SERVER_NAME> \
-servicetype=<SERVICE_TYPE> -logfile=<LOGFILE>
```

   For example, for deleting a managed server 'oacore_server2' of type 'oacore', execute the below command:

```
$ perl <AD_TOP>/patch/115/bin/adProvisionEBS.pl
ebs-delete-managedserver \
-contextfile=<CONTEXT_FILE> -managedsrvname=oacore_server2 \
-servicetype=oacore -logfile=<LOGFILE>
```

2. On the node containing the Oracle HTTP Server, run AutoConfig. Then execute the following script to remove details of the deleted managed server from the Oracle HTTP Server configuration files.

```
$ perl <FND_TOP>/patch/115/bin/txkSetAppsConf.pl
-ctxfile=$CONTEXT_FILE \
-outfile=<OUT_FILE>
```

3. On the node that contains the Oracle HTTP Server, bounce the Oracle HTTP Server as follows:

   On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adapcctl.sh stop
$ sh <ADMIN_SCRIPTS_HOME>/adapcctl.sh start
```

   On Windows:

```
C:\><ADMIN_SCRIPTS_HOME>/adapcct.cmd stop
 C:\><ADMIN_SCRIPTS_HOME>/adapcctl.cmd start
```

Once the managed servers are deleted on the RUN file system, delete them from the PATCH file system by performing the following steps:

1. Start up the WebLogic Administration Server of the PATCH file system by executing the following command on the PATCH file system of the primary node.

   On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adadminsrvctl.sh start forcepatchfs
```

   On Windows:

```
C:\> <ADMIN_SCRIPTS_HOME>\adadminsrvctl.cmd start
```

2. Execute the following command **from the RUN file system** to delete the respective managed server.

```
$ perl <AD_TOP>/patch/115/bin/adProvisionEBS.pl
ebs-delete-managedserver /
-contextfile=<PATCH_CONTEXT_FILE>
-managedsrvname=<MANAGED_SERVER_NAME> /
-servicetype=<SERVICE_TYPE> -logfile=<LOGFILE>
```

3. After deletion of the managed server on the PATCH file system, run adSyncContext.pl **from the RUN file system** on all Application tier nodes, passing the context file of the PATCH file system.

```
$ perl <AD_TOP>/bin/adSyncContext.pl
-contextfile=<PATCH_CONTEXT_FILE>
```

4. Instantiate the fsclone_config.txt file on the PATCH file system of all Application tier nodes by running the following command after sourcing the RUN file system:

```
java oracle.apps.ad.autoconfig.InstantiateFile -e
<PATCH_CONTEXT_FILE> -tmpl  \
<PATCH_AD_TOP>/admin/template/fsclone_config_txt.tmp -out  \
<PATCH_INST_TOP>/appl/admin/fsclone_config.txt
```

5. On the node containing the Oracle HTTP Server, execute the following script **from the RUN file system**, passing the context file from the PATCH file system. This will remove the entry of the deleted managed servers from the Oracle HTTP Server configuration files on the PATCH file system.

```
$ perl <FND_TOP>/patch/115/bin/txkSetAppsConf.pl \
-ctxfile=<PATCH_CONTEXT_FILE> -outfile=<OUT_FILE>
```

6. Shut down the WebLogic Administration Server of the PATCH file system by executing the following command on the PATCH file system of the primary node.

   On UNIX:

```
$ sh <ADMIN_SCRIPTS_HOME>/adadminsrvctl.sh stop
```

   On Windows:

```
C:\> <ADMIN_SCRIPTS_HOME>\adadminsrvctl.cmd stop
```

> **Note:** Managed server deletion should be done only through the adProvisionEBS.pl script as mentioned above. Managed servers should not be deleted from the WebLogic Server Administration Console.

Additional Steps Needed On Non-Shared Multi-Node Systems

Configuration changes peformed via Oracle WebLogic Server (WLS) Console to any of the OACORE, Forms, OAFM and Forms-C4WS applications are reflected in the associated *deployment plans*. . The changes are made automatically to the deployment plan on the node where the Admin Server is running.

If you have a multi-node system, you must manually update the deployment plans on the other nodes, so that the plans on all nodes will be in synchronization.

The deployment plans are located as follows:

- <EBS_ORACLE_HOME>/deployment_plans/oacore/plan.xml
- <EBS_ORACLE_HOME>/deployment_plans/oafm/plan.xml
- <EBS_ORACLE_HOME>/deployment_plans/forms/plan.xml
- <EBS_ORACLE_HOME>/deployment_plans/forms-c4ws/plan.xml

If any configuration changes are made to a deployment plan via WLS Console, follow the steps below to synchronize the deployment plans on the other nodes.

1. Edit the relevant deployment plan to enter the new configuration value in the

variable-definition section.

For example, if the `Session cookies max age parameter` is changed via WLS console, you must make a corresponding update to the variable `WeblogicApplication_SessionDescriptor_CookieMaxAgeSecs` in the variable-definition section of the deployment plan.

2. Save the deployment plan.

3. Restart the managed server.

Using Profiles in Oracle WebLogic Server

Within Oracle E-Business Suite Release 12.2, Oracle WebLogic Server provides the ability to view runtime statistics related to JDBC datasources via the Administration Console. When performance issues or details from monitored runtime statistics indicate there may be a problem within the domain, specific profiles can be enabled to help you pinpoint the source of the problem. To do this, go to Administrative Console and navigate as follows:

Domain Structure: Domain Services => DataSources => (Defined Data Source) => Diagnostics Tab

For more information, refer to the standard Oracle WebLogic Server documentation, specifically Monitoring WebLogic JDBC Resources [http://docs.oracle.com/cd/E15051_01/wls/docs103/jdbc_admin/monitor.html].

While profiling can be a valuable diagnostic tool, within Oracle E-Business Suite Release 12.2 the Data Source configuration is specifically configured to maintain long-running connections. This means that all objects collected when profiling is enabled *will remain in memory until the connection is destroyed*, eventually resulting in out of memory errors within Oracle WebLogic Server.

The time needed for an out of memory error to occur will depend several factors, including:

- The amount of profiling enabled
- Size of the JVM
- Number of users accessing the application

To minimize the chances of experiencing an out of memory error, the following guidelines for profiling are recommended:

- If possible, only use profiling when access to the system is limited
- Before enabling profiling, try to identify the steps needed to reproduce the profiling data that is to be captured

- After profiling is complete, restart Oracle WebLogic Server before opening the system to the general user population.

In addition, alternative strategies to profiling should be considered for production environments. These include enabling JDBC Debug, or obtaining periodic dumps using WebLogic Diagnostic Framework tools.

Changing the Oracle WebLogic Server Administration User Password

The option to set the Oracle WebLogic Server Administration user password to a non-default value is available during Oracle E-Business Suite installation. This section describes the procedure to use if you need to change the password at a later time.

For more information, see: "Changing the Administrative User Password" in Chapter 3 of *Oracle Fusion Middleware Administrator's Guide 11g Release 1 (11.1.1)* .

1. Log in to the Oracle WebLogic Server Administration console at `http://<server>.<domain>:<wls_admin_port>/console` with the current user name and password.
2. Choose the Lock and Edit button.
3. Select the 'Domain' link under the Domain Configuration section.
4. Select the Security tab.
5. Select Advanced.
6. Enter the new Node Manager password.
7. Choose the Save button and then the Activate Changes button.
8. Choose the 'Security Realms' link on the Oracle E-Business Suite 12.2 domain configuration, and then select the 'myrealm' link.
9. Select the Users and Groups tab and choose the 'weblogic' link.
10. Select the Passwords tab and enter the same password you entered in Step 6.
11. Choose the Save button.
12. Back up the following files:
    - `$FMW_HOME/user_projects/domains/EBS_domain_<SID>/servers/AdminServer/security/boot.properties`
    - `$FMW_HOME/user_projects/domains/EBS_domain_<SID>/servers/forms-c4ws_server1/data/nodemanager/boot.properties`

- `$FMW_HOME/user_projects/domains/EBS_domain_<SID>/servers/forms_server1/data/nodemanager/boot.properties`
- `$FMW_HOME/user_projects/domains/EBS_domain_<SID>/servers/oacore_server1/data/nodemanager/boot.properties`
- `$FMW_HOME/user_projects/domains/EBS_domain_<ID>/servers/oafm_server1/data/nodemanager/boot.properties`

13. Use a text editor to edit each of the above files and change the password to the new password.

    Note: After the services are restarted in the next step, the password you entered will be obfuscated (not human readable).

14. Stop and restart all application tier services using the `adstpall.sh` and `adstrtall.sh` scripts (`.cmd` command files on Windows).

15. Ensure all services start successfully, and then test the Oracle WebLogic Server console login and Oracle E-Business Suite login.

Known Issues

This section lists any known issues with the AutoConfig-related configuration management of Oracle E-Business Suite Release 12.2 environments.

- **Problem**: On a multi-node installation with the Forms Services and Batch Processing Services enabled on separate nodes, OAM fails to update the context variables on the Batch Processing Services node.

  Solution: Check whether the Listener Service is up on the Forms Services node. If the service is down, start the service using one of the following options:

  - Start the TNS listener service manually using the following command:

    On UNIX

    ```
    $ $INST_TOP/admin/scripts/adalnctl.sh start <TWO_TASK>
    ```

    On Windows:

    ```
    C:\>%INST_TOP%\admin\scripts\adalnctl.cmd start <LOCAL>
    ```

  - Enable management of the TNS Listener Service via the adstrtal or adstpall scripts on the Forms Services node.

    1. Enable the TNS Listener Service by following the steps mentioned earlier.
    2. Stop all the application tier services using `adstpall.sh` or

`adstpall.cmd`.

3. Start all the application tier services using `adstrtal.sh` or `adstrtal.cmd`.

4

Concurrent Processing

Overview of Concurrent Processing

This section explains how a request to run a concurrent program is handled by Oracle E-Business Suite, and what the life cycle of a concurrent request is.

In Oracle E-Business Suite, concurrent processing simultaneously executes programs running in the background with online operations. As System Administrator, you can manage when programs are run and how many operating system processes Oracle E-Business Suite devotes to running programs in the background.

Concurrent Requests, Programs, and Processes

When a user runs a report, a request to run the report is generated. The command to run the report is a *concurrent request*. The program that generates the report is a *concurrent program*. Concurrent programs are started by a *concurrent manager*.

Concurrent Managers start concurrent programs

Every time your users request a concurrent program to be run, their request is inserted into a database table, and is uniquely identified by a request ID. Concurrent managers read requests from this table.

Part of a manager's definition is how many operating system processes it can devote to running requests. This number is referred to as the manager's number of *target processes*.

Running concurrent programs

A concurrent program actually starts running based on:

- When it is scheduled to start
- Whether it is placed on hold

- Whether it is incompatible (cannot run) with other programs
- Its request priority

Concurrent Request Priorities

The priority of a concurrent request is determined by application username, and is set by the System Administrator using the Concurrent:Priority user profile option.

The first available concurrent manager compares the request's priority to other requests it is eligible to process, and runs the request with the highest priority.

When choosing between requests of equal priority, the concurrent manager runs the oldest request first.

Parent requests and Child requests

Often, several programs may be grouped together, as in a request set. Submitting the request set as a whole generates a request ID, and as each member of the set is submitted it receives its own request ID. The set's request ID identifies the *Parent* request, and each of the individual programs' request ID identifies a *Child* request.

Life cycle of a concurrent request

A concurrent request proceeds through three, possibly four, life cycle stages or *phases:*

Pending	Request is waiting to be run
Running	Request is running
Completed	Request has finished
Inactive	Request cannot be run

Within each phase, a request's condition or *status* may change. The following table shows a listing of each phase and the various states that a concurrent request can go through.

Phase	Status	Description
PENDING	Normal	Request is waiting for the next available manager.

Phase	Status	Description
PENDING	Standby	Program to run request is incompatible with other program(s) currently running.
PENDING	Scheduled	Request is scheduled to start at a future time or date.
PENDING	Waiting	A child request is waiting for its Parent request to mark it ready to run. For example, a report in a report set that runs sequentially must wait for a prior report to complete.
RUNNING	Normal	Request is running normally.
RUNNING	Paused	Parent request pauses for all its child requests to complete. For example, a report set pauses for all reports in the set to complete.
RUNNING	Resuming	All requests submitted by the same parent request have completed running. The Parent request is waiting to be restarted.
RUNNING	Terminating	Running request is terminated, by selecting *Terminate* in the Status field of the Request Details zone.
COMPLETED	Normal	Request completes normally.
COMPLETED	Error	Request failed to complete successfully.
COMPLETED	Warning	Request completes with warnings. For example, a report is generated successfully but fails to print.

Phase	Status	Description
COMPLETED	Cancelled	Pending or Inactive request is cancelled, by selecting *Cancel* in the Status field of the Request Details zone.
COMPLETED	Terminated	Running request is terminated, by selecting *Terminate* in the Status field of the Request Details zone.
INACTIVE	Disabled	Program to run request is not enabled. Contact your system administrator.
INACTIVE	On Hold	Pending request is placed on hold, by selecting *Hold* in the Status field of the Request Details zone.
INACTIVE	No Manager	No manager is defined to run the request. Check with your system administrator.

Related Topics

Reviewing Requests, Request Log Files, and Report Output Files, page 4-224

How to View Request Status and Output, page 4-224

Setting End User Report and Log File Access Privileges, page 4-225

Managing Concurrent Processing Files and Tables, page 4-229

Service Management

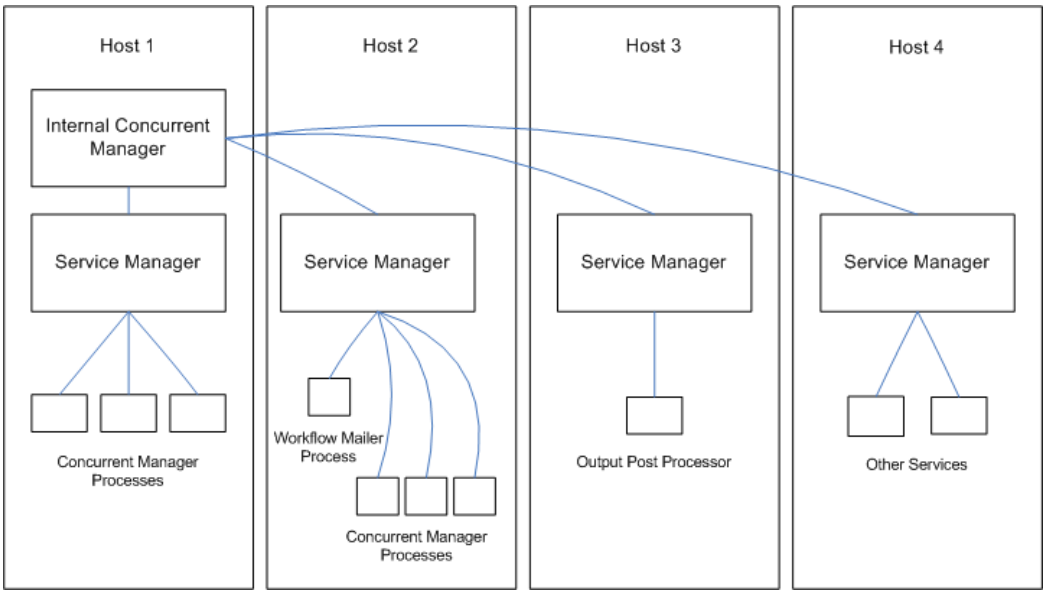
An Oracle E-Business Suite system depends on a variety of services such as Concurrent Managers and Workflow Mailers. Such services are composed of one or more processes that must be kept running for the proper functioning of the applications. Previously many of these processes had to be individually started and monitored by system administrators. Management of these processes was complicated by the fact that these services could be distributed across multiple host machines. Service Management helps to greatly simplify the management of these processes by providing a fault tolerant service framework and a central management console built into Oracle Applications

Manager.

Generic Service Management (GSM, or simply Service Management) is an extension of concurrent processing, which provides a powerful framework for managing processes on multiple host machines. With Service Management, virtually any application tier service can be integrated into this framework. Services such as the Oracle Workflow Mailer or Java services can be managed under Generic Service Management.

With Service Management, the Internal Concurrent Manager (ICM) manages the various service processes across multiple hosts. On each host, a Service Manager acts on behalf of the ICM, allowing the ICM to monitor and control service processes on that host. System administrators can then configure, monitor, and control services though a management console which communicates with the ICM.

Generic Service Management



Service Management provides a fault tolerant system. If a service process exits unexpectedly, the ICM will automatically attempt to restart the process. If a host fails, the ICM may start the affected service processes on a secondary host. The ICM itself is monitored and kept alive by Internal Monitor processes located on various hosts.

Service Management provides significant improvements in the manageability of Oracle E-Business Suite. System administrators can now use the central console in Oracle Applications Manager (OAM) to manage a variety of services that formerly had to be managed independently on separate hosts. The entire set of system services may be started or stopped with a single action. Service Management also provides a great benefit by automatically compensating for certain system failures.

Service processes are very much like concurrent manager and transaction manager processes. They must be kept running on a middle tier for the proper functioning of their respective products. The concurrent processing management feature has been

built for concurrent managers and transaction managers, to provide fault tolerance, process distribution, and simplified configuration and control.

Benefits of Service Management

- The service processes will no longer need to be manually and individually started and monitored by Oracle E-Business Suite system administrators.
- Services can take advantage of the process distribution and fault tolerance capabilities that have been developed for concurrent processing.
- As with concurrent manager processes, system administrators can use work shifts to determine the number of processes that will be active for a service on a given node for a given time period.

To extend process management support to the various Applications services, the Internal Concurrent Manager must be able to start, monitor, and control processes on all Applications tiers. Every node of every tier will have an Oracle RPC-based Service Manager installed. The ICM will use the Service Manager to manage processes.

Concepts

Service

A service is a process or collection of processes that perform actions at the request of client processes. A concurrent manager is a type of service where the client submits a request for actions to be processed while the client continues to do other work.

While active, a service must have one or more listener processes that wait to process requests from clients. An example of a listener is a concurrent manager process which periodically polls a queue for requests to process.

Service Instance

Each service controlled by service management may have multiple service instances. Each instance may consist of one or more processes.

Network Failure Recovery

As part of its shutdown process, the ICM determines if it's being forced to shutdown due to losing its database connection. This is done by looking for specific error messages ORA-3113, ORA-3114, or ORA-1041. If one of these error messages is detected, the ICM spawns the reviver process, which attempts to make a database connection. If unsuccessful, it sleeps for a period before trying again. This continues until either a successful connection is made or it receives a signal to shut itself down.

When a successful connection is made, the process kills the old ICM database session,

and then starts a new ICM using the normal start manager script. Once the ICM is restarted, it starts up any other managers that had also shut down, and normal processing resumes.

Failover Sensitive Workshifts

Nodes can become overloaded when a middle-tier node fails and service instances on that node failover to their secondary nodes. The *Load Distribution* feature allows the System Administrator to control the allocation of resources during normal processing. The *Failover Sensitivity* feature allows Work Shifts to failover with fewer processes than on the original node. This lessens the impact on the existing resources allocated on the secondary node.

The number of failover processes is entered as part of the standard Work Shift settings in the *Service Instance Definition*. When failover occurs, the ICM uses the **Failover Processes** value in place of the normal running processes value as it iterates through service instances to perform queue sizing.

Overview of Concurrent Programs and Requests

A concurrent program is an executable file that runs simultaneously with other concurrent programs and with online operations, fully utilizing your hardware capacity. Typically, a concurrent program is a long-running, data-intensive task, such as posting a journal or generating a report.

Standard Request Submission (SRS) is an Oracle E-Business Suite feature that allows you to select and run your concurrent programs from a single, standard form (*Submit Request*) or window (*Schedule Request*). Requests to run concurrent programs are called concurrent requests.

There are two main ways to group concurrent programs. Request sets are defined to run several concurrent program in a single request. Request groups are used to control access to concurrent programs via responsibilities. Both request sets and request groups are discussed in later sections

Related Topics

Organizing Programs into Request Sets, page 4-11

System Administrator Request Set Privileges, page 4-24

Organizing Programs into Request Groups, page 4-28

Copying and Modifying Program Definitions, page 4-50

Running Reports and Programs, *Oracle E-Business Suite User's Guide*

for a User

You can restrict users' access to submit and view concurrent requests.

Controlling Access to Concurrent Programs with Request Security Groups

Note: This method is used in releases prior to Release 12.

A *request security group* is a collection of reports or concurrent programs. A System Administrator defines request security groups in order to control user access to reports and concurrent programs. Only a System Administrator can create a request security group.

- The reports and concurrent programs that may be selected by a user in Standard Request Submission belong to a *request security group*, which is a request group assigned to a responsibility.

  Important: The use of request security groups is for backward compatibility only.

- The reports and concurrent programs that may be selected from a customized SRS form or window belong to a request security group that uses a code.

  See: Customizing the Submit Request Window using Codes, page 4-32.

Controlling Access to Concurrent Programs using Role-Based Access Control (RBAC)

RBAC allows administrators to have more granular control in securing data related on concurrent programs and requests.

Submitting Requests

Administrators can assign individual programs/sets, all programs/sets in a request group, programs/sets belonging to one or more applications, and so on, either to the user directly or to a role that can then be assigned to one or more users. For more information on RBAC, see: Overview of Access Control with Oracle User Management, *Oracle E-Business Suite Security Guide*.

If applications are included in the request groups, all programs/requests sets that are created in these applications will also be automatically included. Please note that request submission applies to both programs and request sets.

The following types of "instance sets" can be used for assignment (but administrators can create new instance sets based on their needs):

- All programs in a particular request security group
- All request sets in a particular request security group

To enable this functionality, the following are seeded:

- Permission "Submit Request"
- Permission "View Request"
- Permission Set "Request Operations" containing the permissions "Submit Request" and "View Request"
- Object "Concurrent Programs"
- Object Instance Set "Programs that can be accessed"
- Object Instance Set "Request sets that can be accessed"

To grant access to a request security group to a role, follow these steps:

1. Define your role (User Management responsibility).
2. Define your request security group (System Administrator responsibility).
3. Define your grant (Functional Administrator responsibility).
    1. Enter a Name and Description (optional) for the grant.
    2. Enter the Security Context for the grant.
    3. Under Data Security, choose "Concurrent Programs" or "Request Sets" as the object. Click Next.
    4. For the Object Data Context, select "Instance Set" for the Data Context Type. Choose either "Programs that can be accessed" or Request Sets that can be accessed" as appropriate. Click Next.
    5. Review the Instance Set information. Under Instance Set Details, enter the request group and its application. Specifically, enter the request group name as Parameter 1 and the application short name as Parameter 2.
    6. Choose "Request Operations" as the permission set under "Set". Click Next.
    7. Review the grant information and save your work.

Note that there are two seeded grants for all users to account for request group assignments that already exist for legacy responsibilities. These are:

- Programs - Grant Defaults
- Request Sets - Grant Defaults

Viewing Requests

You can control users' access to viewing requests with RBAC.

> **Note:** In previous releases, the Concurrent: Report Access Level profile was used to control privileges to report output files and log files generated by a concurrent program. This profile is no longer used.

Seeded "instance sets" allow administrators to grant:

- All requests submitted by a user
- All requests submitted by a user for a given application
- All requests belonging to a program submitted by a user
- All requests belonging to a request set submitted by a user (irrespective of the constituent programs' owning application)

to another user (or a group of users - via a role).

System administrators can create new "instance sets" based on their needs. They can grant access to requests (of a particular program/set) submitted by all users to a specific set of users. For example, say a given application's administrators group want to track all requests of a particular type or program submitted by business users. Then the following approach, to grant specific programs' requests to a group of users, can be used:

1. Create an instance set that selects all the requests belonging to the particular program irrespective of which user submitted it.

   For example,

   ```
   &TABLE_ALIAS.request_id in
       ( select cr.request_id
         from fnd_concurrent_requests cr, fnd_concurrent_programs cp
         where cr.concurrent_program_id = cp.concurrent_program_id
         and cr.program_application_id = cp.application_id
         and cp.concurrent_program_name = &GRANT_ALIAS.PARAMETER1)
   ```

   If you want to grant access to a set of programs instead of a single program, '&GRANT_ALIAS.PARAMETER1' can be replaced with a subquery that returns all the programs in a particular request group.

2. Create a grant to grant this instance set to (an existing) role, for example, "<Application> Administrator" role, and assign the program name to grant. Use the "Concurrent Requests" data object in creating this grant.

3. Ensure that the role is assigned to all users that should have access to these requests.

Limiting Active Requests by a User

As System Administrator you can limit the number of requests that may be active (status of Running) for an individual user. This ensures that a user cannot monopolize the request queue. For example, if a user with an Active Request Limit of 5 submits 20 requests, only 5 requests will be run at the same time. The remaining requests will be run when the number of active requests for the user drops below 5. Use the Profile Options window to set the *Concurrent: Active Request Limit* profile. To set a global limit for all users, set this option at the site level. You can then modify limits for individual users by setting this profile option at the User level.

Organizing Programs into Request Sets

Reports and concurrent programs can be assembled into request sets.

Request sets define run and print options, and possibly, parameter values, for a collection of reports or concurrent program. End users, with the appropriate privileges, and System Administrators can define request sets. A System Administrator has request set privileges beyond those of an end user. Request sets can be run from Forms-based applications and HTML-based applications.

Request sets are a quick and convenient way to run several reports and concurrent programs with predefined print options and parameter values. Request sets group requests into stages that are submitted by the set. The order in which the stages are submitted is determined by the status of previous stages.

Request sets can also be used by a System Administrator to customize access to reports and concurrent programs. Using request sets, a System Administrator can:

- grant users of a responsibility the ability to run selected reports and concurrent programs that are outside their request security group.
- grant access to requests and other concurrent programs on a user-by-user basis.
- guarantee that reports in the set run with print options and parameter values that cannot be edited by end users.

As System Administrator, you have privileges beyond those of your application users, including a privileged version of the Request Set window.

Defining Request Sets

You can run the same set of concurrent requests regularly by defining a request set, and then submitting the request set from the Submit Requests form.

As System Administrator, you can include *any* Standard Request Submission report or concurrent program in the request sets you define. When end users define a request set, they can only select from reports and programs that belong to their responsibility's request security group.

Use the Request Set form to create and edit request sets.

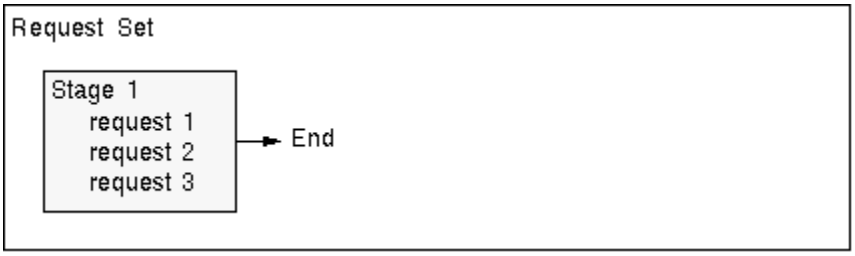
Request Set Stages

This section describes how request set stages are defined and organized.

Organizing Request Sets into Stages

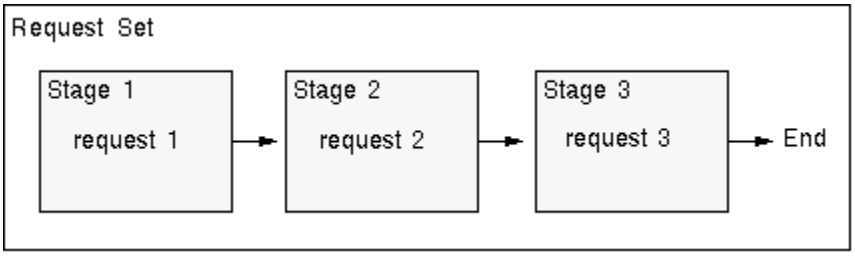
Request sets are divided into one or more "stages" which are linked to determine the sequence in which requests are run. Each stage consists of one or more requests that you want to run in parallel (at the same time in any order). For example, in the simplest request set structure, all requests are assigned to a single stage. This allows all of the requests to run in parallel.

Request Set with a Single Stage



To run requests in sequence, you assign requests to different stages, and then link the stages in the order you want the requests to run.

Request Set with a Sequence of Stages

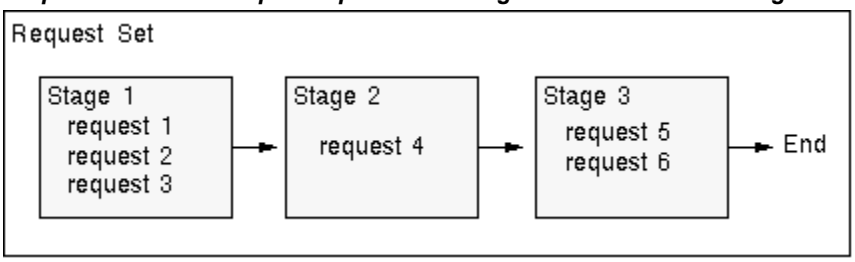


The concurrent manager allows only one stage in a request set to run at a time. When one stage is complete, the following stage is submitted. A stage is not considered to be complete until all of the requests in the stage are complete.

One advantage of using stages is the ability to run several requests in parallel and then move sequentially to the next stage. This allows for a more versatile and efficient

request set.

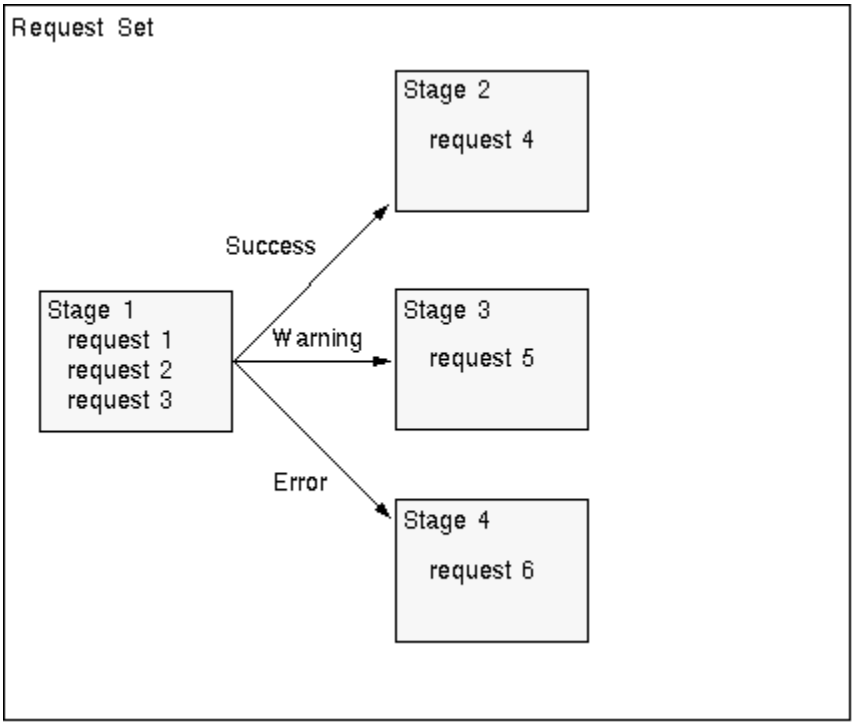
Request Set with Multiple Requests Running in Parallel within a Stage



Using Stage Status

Like request sets and concurrent requests, stages can complete with different statuses. Each stage can complete with a status of Success, Warning, or Error. You can use these completion statuses to structure your request set, by defining which stage will follow the current stage based on its completion status. For example: a request set always begins with Stage 1. If Stage 1 completes with the status Success, then the Success link is followed, and Stage 2 is submitted. After Stage 2 completes, the set ends. If Stage 1 completes with Warning, then the Warning link is followed, and Stage 3 is submitted. After Stage 3 completes, the set ends. If Stage 1 completes with Error, then the Error link is followed, and Stage 4 is submitted. After Stage 4 completes, the request set ends.

Request Set Using Stage Statuses

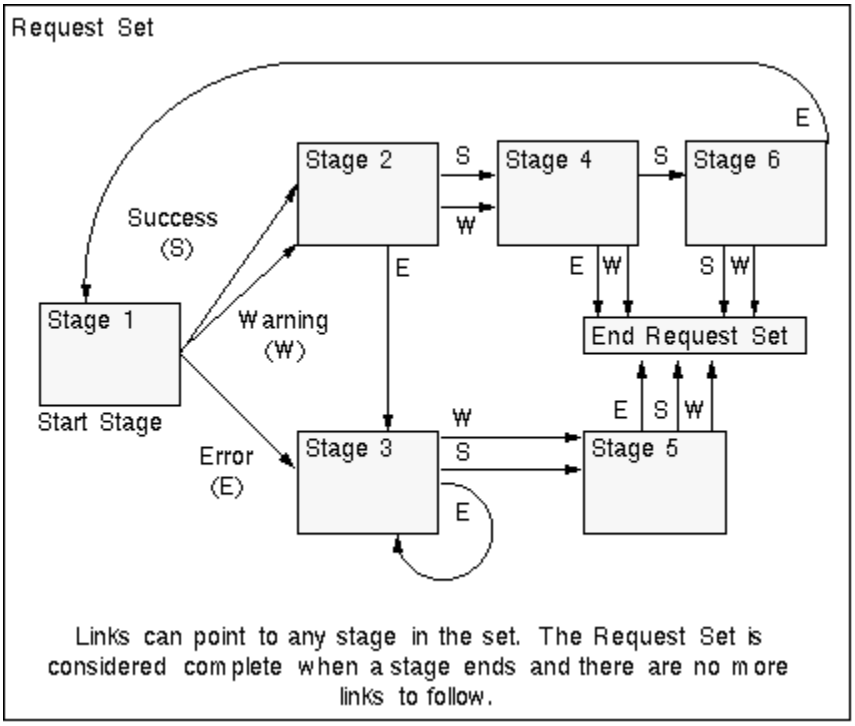


In this example, the stage status is determined using the Standard stage function. The Standard stage function uses the statuses of the requests within the stage to calculate the status for the stage. If all of the requests in a stage complete with a status of Success, then the status for the stage is Success. If one or more requests complete with a status of Error, then the status of the stage is Error. For a stage's status to be Warning, one or more of the requests must have a status of Warning, and no request may have a status of Error.

Linking of Stages

There are no restrictions on linking stages within a request set. Any stage may be linked to any other stage, including itself. Two or more links can point to the same stage. For example, Stage 1 can link to Stage 2 if the completion status of Stage 1 is Success or Warning, and link to Stage 3 if the status is Error.

Request Set with Multiple Links to Stages



You determine the end of a request set by not specifying a followup stage for each completion status. You can end a request set after any stage in the request set. When any stage completes with a status that does not link to another stage, the request set ends.

You can use the linking of stages to control your request set. In previous releases you had three options: run in parallel, run sequentially, and run sequentially but abort on Error. All of these are easy to recreate using the request set wizard. You can use the Request Set Wizard button in the Request Set window to start the wizard. The wizard takes your input and creates the request set as follows:

Run in Parallel — Creates one stage containing all of the requests you wish to run in parallel.

Run Sequentially — Creates a separate stage containing the request or requests for each step in the sequence and link in the appropriate order.

Run Sequentially but abort on Error — Sets up your sequence the same as it did for Run Sequentially, but when it links the stages, it does not enter a follow up stage as a link in the Error completion status field.

Stage Evaluation Function

The completion status of a stage is determined by a predefined function. The Oracle E-Business Suite Standard Stage Evaluation function uses the completion status of the requests it contains. Use this function to determine the status of a stage.

Request Set Completion Status

When a stage completes with a status for which there is no link defined, the request set ends. The completion status for the request set is determined by one of the following methods:

- Using the completion status of the last stage run in the request set. This method is used by default.
- The user can override the default behavior by defining a specific stage within the set to be "critical". If the request set has a single critical stage defined, and then runs this critical stage, then the completion status of the set will be the same as the completion status of the critical stage. This can be useful if the final stage of the set is a "clean up" stage and is not considered important to the overall status of the set.
- If a request set has more than one critical stages defined, the "worst" completion status of all the critical stages is considered the completion status of the set.

Printing Request Sets

On a report-by-report basis, you can select a different printer for each report in a request set. When you define a request set, print options, such as the printer a report is sent to, are saved so you do not have to specify them again when you run the request set.

> **Important:** If a printer is defined for a concurrent program using the Concurrent Programs form, then that value cannot be updated, either by a user profile option setting, a request set definition, or when running the program or request set.

> **Note:** Defining a printer for a request set concurrent program (for example, Request Set Payables Aging Reports) in the Concurrent Programs form has no effect; the printer definition is not referred to.

Holding Request Sets

In some circumstances, such as when a request set has a large number of stages and takes a long time to execute, administrators may want to yield a request set to higher priority requests. By utilizing the *Hold Request Set* feature, users can place a running

request set on hold and effectively control the execution of request set stages.

The **Hold** and **Remove Hold** buttons are available on the OAM View Running Requests page. To hold a request set, simply select the request set and click the **Hold** button. Click **Remove Hold** when you want the request set to continue executing.

Request Sets as Concurrent Programs

When you define a request set or a stage within a request set that allows incompatibilities, a concurrent program is created to run the requests in your request set according to the instructions you enter.

All concurrent programs that run request sets are titled *Request Set <name of request set>*, and programs that run request set stages are titled *Request Set Stage <name of request set stage>*. In the Concurrent Programs form, to query request set or request set stage concurrent programs on the basis of a program's name, enter the following in the Name field:

- "Request Set" or "Request Set Stage" before the name of the concurrent program
- "Request Set %" or "Request Set Stage %" to perform a query on all request set programs

Request set and request set stage concurrent programs create log files documenting the execution of the request set or stage. Each report or concurrent program within a request set or stage also creates its own log file.

> **Note:** The recorded actual start time for a request is the start time the request's status is changed to "Running". A request may need to wait to start until it has necessary information from child process(es).

When you run a request set that allows incompatibilities, you submit a request to run the concurrent program that defines the request set. The request set concurrent program submits a request set stage concurrent program. The request set stage concurrent program submits the requests for the individual programs and reports within the stage. A request to run the request set concurrent program or the request set stage concurrent program is a *Parent* request, while the requests to run the programs and reports are *Child* requests.

You can review the status of a request set and the programs it contains using the Concurrent Requests form. The following table displays information pertaining to request sets in the Running phase.

Status	Description
Paused	Parent request pauses for all its Child requests to complete. For example, a request set stage pauses for all reports in the stage to complete.
Resuming	All requests submitted by the same Parent request have completed running. The Parent request resumes running.

Modifying Request Sets

A request set can only be modified by its owner or by a System Administrator. To make modifications, query the request set you want to modify in the Request Set window.

> **Note:** If you wish to retain modifications to request sets provided by your Oracle application during upgrades, you must rename or recreate the request set using a different name before you upgrade. If you modify a predefined request set without changing the name, your modifications are overwritten when you upgrade your Oracle E-Business Suite.

Creating Request Sets

Follow this procedure to create a request set:

1. Navigate to the Request Set window.
2. Enter a Name for your request set.
3. Enter a Set Code for your request set. This name is used internally to reference your request set.
4. Enter the Application with which you want to associate your request set.
5. Enter a Description of your request set if you like.
6. The Owner field defaults to your username and can only be changed by your system administrator.
7. Enter values for the Active Dates "From" and "To" fields to define an effective period when you and others can run the request set. If the current date is outside the range you define, the request set will not be available in the Submit Requests window. Note that the request set will not be available on the date specified in the

"To" field.

8. Check the Print Together check box to send all your requests to the printer together when they complete, or uncheck the check box to send each request one at a time to the printer as it completes.

9. Check the Allow Incompatibility check box to allow your system administrator to specify programs that this request is incompatible with (may not run with). Leave Allow Incompatibility unchecked to specify that this request set may run with all other concurrent requests or request sets.

10. Choose **Define Stages** or **Link Stages** if you have finished defining your stages.

Defining Stages

Follow this procedure to define stages:

1. The value for the Display Sequence is defaulted in sequence as you enter your stages. You may change the display order of the stages by modifying this field.

2. Enter a Name for the stage.

3. Enter a Description of your stage if you like.

4. Enter a Stage Code for the stage. This code is used internally to reference the stage.

5. In the Function field of the Function region, use the List of Values to select a function. The default value for this field is the Standard Stage Evaluation function. This function bases its completion status on the normal completion status of the requests it contains. Other functions may be provided by your Oracle product. For a description of these functions, refer to the user's guide for that product.

6. Use the "The Return Value of this Stage Affects the Set Outcome" check box if you want to ensure that the request set's completion status is equal to the completion status of this stage.

   Note: If you choose this check box for more than one stage, the completion status of the request set will equal the completion status of the last of these stages to run within the set.

7. Use the Allow Incompatibility check box to allow your system administrator to specify programs that this stage is incompatible with (may not run with). Leave Allow Incompatibility unchecked to specify that this stage of the request set may run with all other concurrent requests or request sets.

8. Choose **Requests**.

Stage Requests Window

In the Stage Requests window you define which requests you want to include in the stage.

1. Select the report or program you want to include in your request set. A description of the request you choose and its associated application appears in the Description and Application fields.

   The list of requests you can choose includes the requests that your responsibility's request group lets you access from the Submit Requests form.

2. Use the Allow Stage Function to Use This Program's Results check box to indicate which programs or reports should be included.

3. The Print Options region reflects the options for the current request. Specify the number of Copies of output to print, the Style to print, the Printer to print to, and whether to save the output to an operating system file.

   Standard Request Submission saves these options so you do not have to specify them again when you run the request set. If you do not wish to specify these options for each request when you define the set, Standard Request Submission uses the values of your personal profile options as the default when you submit the request set. See: Concurrent Processing User Profile Settings, *Oracle E-Business Suite Maintenance Guide*.

   Note: Some requests may have a required Style or Printer that you cannot change.

4. When you are done with the Print Options, choose Parameters to display the Request Parameters window.

Request Parameters Window

The Request Parameters window lets you customize the parameter values of a specific request in a request set. The fields at the top of the Request Parameters window list general information about the current request set and the request for which you can customize the parameter values. The multi-row portion of the window lists the parameters for that request.

1. The Sequence field displays the order in which each request parameter appears when you run the request in the Submit Requests window (lower numbers appear before higher numbers). Only your system administrator can change a parameter's order.

2. The Prompt field is a display-only field that shows the request parameter's prompt.

3. Check the Display check box to specify that you can see a request parameter at submission time, or uncheck the check box to specify that a parameter should not be displayed at submission time.

4. Check the Modify check box to specify that you can insert or change the value for a request parameter at submission time, or uncheck the check box to specify that a parameter cannot be changed at submission time.

5. Use the Shared Parameter field to set a default value for a parameter that occurs in more than one report or program of a request set. Once you enter the same parameter label in the Shared Parameter field for each occurrence of the same parameter, the value that you assign to the first occurrence of the parameter becomes the default value for all subsequent occurrences of the parameter. The shared parameter label simply enables you to set an initial default value for all occurrences of the same parameter so you can avoid typing the same value all over again for every occurrence of the parameter.

   For example, suppose you define a request set that includes three reports, and all reports include a parameter called "Set of Books". You want the "Set of Books" parameter to default to the same value in all reports. To accomplish this, enter a label called "Book" in the Shared Parameter field for the first occurrence of this parameter. You can also assign a value in the Default Value field of this parameter now, or wait until you run the request set to assign a default value when the parameter first appears. Enter the label "Book" in the Shared Parameter field of all other occurrences of the "Set of Books" parameter in your request set. When you submit this request set from the Submit Requests window, every parameter that you label "Book" defaults to the value you assign to the first occurrence of the "Set of Books" parameter.

   > **Important:** Note that if you later change the value of a parameter that contains a shared parameter label, you change only the value for that instance of the parameter, and not the value for all other occurrences of that labelled parameter.
   >
   > We recommend that if you make a parameter with a shared parameter label modifiable, that you also display the parameter so you can always see what the parameter's current value is. This helps reinforce the understanding that a later value change to one labelled parameter cannot propagate a value change to all other similarly labelled parameters.

6. Optionally enter a Default Type and Value for the parameter.

7. Save your work.

8. Go back to the Stage Requests window and repeat Steps 9 through 11 to add more

requests to the request set stage.

You can select a request more than once if you want to run the same request with different default parameter values.

9. To start a new stage, return to the Stage window and choose New Record from the File Menu.

Linking Stages

Follow this procedure to link stages:

1. Enter the Start Stage. The stage you enter here is the first stage submitted for the request set.

2. Enter the stages you want to run following the first stage in the Success, Warning, and Error columns. To ensure that a particular stage follows the preceding stage regardless of the completion status, enter the desired stage in all three columns. To stop the request set if a stage ends in Error, leave the Error column blank. Any time you do not specifically indicate which stage should follow for a completion status, the request set will exit on that completion status.

   In the example shown in the table below, the request set will always exit if any stage returns a completion status of error. In addition, stages C and D will terminate the request set regardless of their completion status. If Stage A returns a status other than Error, Stage B will be submitted. Finally, when Stage B completes with a status of Success, it is followed by Stage C, or if the status is Warning, Stage D will follow.

3. Choose **Done.**

The following table shows an example of linking stages as in step 2 above:

Display Sequence	Name	Success	Warning	Error
1	Stage-A	Stage-B	Stage-B	
2	Stage-B	Stage-C	Stage-D	
3	Stage-C			
4	Stage-D			

Restarting Request Sets

If a request set completes with a status of *Error*, the **Restart** button, on the Oracle

Applications Manager - View Completed Requests page is enabled. The system also automatically captures, records, and saves the information of the first stage that fails so that when the user clicks on the **Restart** button the request set can restart from that point.

Once the stage has been identified, the request set program submits the stage program in resubmit mode. In this mode, the program looks at the same stage from the previous run and determines which programs need to be rerun, (only those that ended in error), and runs those programs. If this stage completes successfully or has a *Warning* status, the system proceeds to the next stage using the normal mechanism of restarting the request set program.

> **Note:** Users may restart a request set multiple times. The logs for each stage and individual programs are maintained independent of the number of runs as each stage and program submission generates a new request. However, the logs and associated files for a request set are rewritten each time the set is restarted.

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Request Sets and Owners

There are significant differences between end user and System Administrator privileges when defining or editing request sets.

End users own the request sets they create

An end user can create a request set by selecting reports, other request sets, or concurrent programs that are part of the report security group assigned to his or her responsibility.

When an end user creates a request set, the user automatically becomes the "owner" of the request set. Ownership is identified by the person's application username.

End users use the Request Set form to create a new request set, or to query and update any request sets they own. End users can only edit request sets they own.

We sometimes refer to a request set that an end user owns as a *private* request set. Private request sets are not automatically added to a request security group. That is, other users cannot access your private request sets using the Submit Requests window unless the System Administrator assigns that request set to a request security group.

Request sets owned by an end user are always available to that user, regardless of what responsibility the user is operating under. However, a standard submission form customized to display only reports in a request group using a code does not display private request sets.

When a user signs on to Oracle E-Business Suite, the user can run requests, request sets, and concurrent programs included in:

- his or her responsibility's request security group
- any request sets they own.

End User Benefits from Private Request Sets

Private request sets offer two main benefits to end users:

1. The request sets that users own are always available to them, regardless of which responsibility they are working under.
2. Users can create as many request sets as they want without adding request set choices to the list of standard submission concurrent programs that other users must select from.

System Administrator Request Set Privileges

As System Administrator, you can:

- create request sets that include *any* reports or concurrent program.
- query and edit *all* request sets using the Request Set form.
- permit and define incompatibility rules for individual request sets. See: Request Set Incompatibilities, page 4-25.

After you define a request set, you can assign a user to be its owner if you want the user to be able to run or edit this request set from any responsibility. Request sets without an owner cannot be edited or updated by any end users. In this way, you can guarantee print options and report parameters for a request set. You can also later edit the request set to remove or change its ownership properties.

Other users can also run a request set if you, as System Administrator, assign the request set to their responsibility's request security group. If you do not assign a request set to a request security group, then only the owner can run the request set. In this way, you can grant access to reports and concurrent programs on a user-by-user basis.

Request Security Groups, Request Sets, and Reports

As System Administrator you can add any request set, including private request sets, to a request security group. This allows you to provide members of a responsibility access to reports and programs outside their request security group.

Request set editing and report viewing privileges are different for reports that belong to a user's request security group than they are for reports that are not in the user's request security group.

- cannot edit the request set.
- cannot run an individual report by itself, but can only run the entire request set.
- can add any other requests in their request security group to the request set.
- can delete any request from the request set, regardless of whether that report is in their request security group.
- can update print options or parameters for an individual report in the request set, if the report is in their request security group.
- cannot run an individual report by itself, but can only run the entire request set.

System Administrator Benefits from Request Sets

Request sets offer three main benefits to System Administrators:

1. Request sets offer a means of controlling access to concurrent programs on a user-by-user basis.

   By defining a request set, assigning it an owner, and then not assigning the request set to any request security group, the reports and programs in the request set are only available to the owner.

2. By leaving the Owner field blank, System Administrators can create request sets whose individual programs and parameters cannot be edited or updated by end users.

   Only a System Administrator can edit a request set that has no owner.

3. System Administrators can provide members of a responsibility access to reports and programs outside their request security group.

   By defining a request set that contains reports or programs not in a request security group, and assigning that request set to the request security group, users can be granted run, but not edit privileges for selected reports or programs.

Request Set Incompatibilities

A request set is actually a concurrent program that submits requests to run each program in the request set. You can allow incompatibility rules to govern your request set so that the request set does not run at the same time as other reports or concurrent programs. You can also apply these rules to the stages that make up the request set.

Use the Concurrent Programs form to query the request set concurrent program and list those programs, and/or stages you want to define as incompatible with your request set. See: Concurrent Programs, page 4-71.

All concurrent programs that run request sets are titled *Request Set <name of request set>*.

In the Concurrent Programs form, if you query a request set concurrent program on the basis of the program's name, you must enter in the Name field the words:

"Request Set" before the name of a concurrent program

"Request Set %" to perform a query on all request set programs

When you list a program as incompatible with your request set, the program will not run simultaneously within the same conflict domain as the request set or any of the reports within the set. See: Defining Program Incompatibility Rules, page 4-34.

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Defining Program Incompatibility Rules, page 4-34

Concurrent Programs, page 4-71

Sharing Parameters in a Request Set

Parameters, also referred to as *arguments,* are values that define aspects of a program's execution. You can share a parameter and its entered value among some or all of the requests in your request set.

You identify a parameter as shared by giving it a label. Then, for each concurrent program in your request set, you can assign the same label to a parameter for that program. Among the programs in your request set, the parameters for each program share or accept a common value.

The *first time* you enter a value for any of the shared parameters, that value becomes the shared parameter's value. This is useful, because you only have to enter a value once, rather than for each program in the request set.

Behavior of Shared Parameters

Selecting a value for a shared parameter provides a default for subsequent occurrences of the parameter. Changing a shared parameter's value provides a new default for subsequent occurrences of the parameter, but does not affect prior requests in the set.

Once all the shared parameters contain values, changing the value for a shared parameter has no effect on the other shared parameters.

> **Important:** Do not hide shared parameters. Do not set shared parameters to Display = No (which prevents modifying the value) or Modify = No. This prevents updates to shared parameters, which are not propagated to other reports in the set, from generating unwanted inconsistencies.

Example - Shared Parameter Value

We've created a request set containing two reports, a *Concurrent Programs Report* and the *Concurrent Program Details Report*. The two reports and their parameters are listed in the table below:

REPORT	PARAMETERS
Concurrent Programs Report	Application Name
Concurrent Program Detail Report	Application Name, Program

We identify the parameter Application Name as a parameter shared between the two reports. We want to enter a value only once, that is, when the Report Parameters window appears for the first report in the set, requiring us to enter Application Name.

To identify a shared parameter, we give it a name, in this example, *applname*, and enter it as a Shared Parameter for each report.

Report Parameters

Sequence	Prompt	Display	Modify	Shared Parameter	Type	Default Value
1	**Application Name**	Yes	Yes	**applname**		

Report Parameters

Sequence	Prompt	Display	Modify	Shared Parameter	Type	Default Value
1	**Application Name**	Yes	Yes	**applname**		
2	**Program**	Yes	Yes			

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Control the Behavior of Request Parameters, page 4-54

Concurrent Programs, page 4-71

Request Sets Report

This report documents request set definitions, including the set's owner, program incompatibilities, as well as printer and print style information. Use this report when

defining or editing request set definitions.

Report Parameters

None.

Report Headings

The report headings provide you with general information about the contents of the report.

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Organizing Programs into Request Sets, page 4-11

Concurrent Programs Report, page 4-63

Organizing Programs into Request Groups

This essay explains how you can organize your applications programs and reports into request groups. It presents the following topics:

- Request Security Groups
- Using Codes with Request Groups
- Customizing the Submit Requests Window using Codes
- Report Group Responsibilities report

Defining a Request Group

When defining a request group, you can include:

- all the reports and concurrent programs owned by an application
- individual reports and concurrent programs
- request sets, which are collections of reports and concurrent programs that may be selected from an application user's request security groups
- request set stage functions, which are used to calculate the status of stages within a request set.

Two Types of Request Groups

A request group is used by Oracle E-Business Suite at two different levels:

1. Responsibility level

   When a request group is assigned to a **responsibility,** it is referred to as a request security group, and it defines the reports, request sets, and concurrent programs that a user, operating under that responsibility, can select from the Submit Requests Window.

2. Form level

   When a request group is assigned a **code**, that code can be passed as a parameter to the Submit Requests Window. The code helps define the function that calls the Submit Requests Window.

   The list of values for that unique Submit Requests Window lists the reports, request sets, and concurrent programs in the request group.

Request Security Groups

When a request group is assigned to a responsibility, the request group is referred to as a *request security group*. Any user signed on under a responsibility can run the reports and concurrent programs contained in their responsibility's request security group.

The Submit Requests standard submission form displays a list of all the reports and programs in the current responsibility's request security group.

Related Topics

Using Codes with Request Groups, page 4-29

Customizing the Submit Requests Window using Codes, page 4-32

Report Group Responsibilities Report, page 4-33

Request Groups, page 4-65

Using Codes with Request Groups

Normally, when a menu calls the standard request submission form, that form can list the reports and concurrent programs contained in the report security group for the current responsibility.

Alternatively, you can assign a code to a request group so that a customized standard submission form only displays a list of concurrent programs contained in that particular request group. A request group code is simply an argument that is passed from a menu to a customized standard submission form. To summarize:

- Request group codes provide a form-based method of controlling user access to

concurrent programs and reports.

- A code can be assigned to a request group.
- You can use the code as an argument passed from a menu to the standard submission form.
- When a menu that calls the standard submission form uses the code, that form lists only those programs in the request group identified by the code.

Related Topics

Organizing Programs into Request Groups, page 4-28

Customizing the Submit Requests Window, page 4-30

Customizing the Submit Requests Window using Codes, page 4-32

Request Groups, page 4-65

Customizing the Submit Requests Window

You can customize the Submit Request window in several ways.

Rename the Window Title

You can change the title to reflect the requests available in the window. See: Customizing the Submit Requests Window using Codes, page 4-32.

Restrict Requests Available to A Request Group

You can restrict the reports and programs available to those in a specified request group. See: Customizing the Submit Requests Window using Codes, page 4-32.

Restrict Requests to a Single Request

You can call Submit Requests form for a single request submission by passing the program/set name as parameters

The parameters window pops up on navigation to the form when called with a program/report_set name. The form exits after the user acknowledges the displayed request ID for the submitted request.

Restrict Requests To A List of Requests

You can call Submit Requests form to submit one or more requests for a single program/set by passing the program/set name as parameters

The parameters window pops up on navigation to the form and the user can submit one or more requests for the program that was passed as a parameter. Requests for other

programs cannnot be submitted in this case.

Pass Parameters Used in Value Set Parameters

You can pass additional parameters to the Submit Requests form that can be referenced in the value sets to validate the request parameters.

Pass Manufacturing "ORG" Parameters

You can pass 5 ORG related parameters and refer to them in the value set. Alternatively, you can bring up a ORG LOV on navigation to the Submit Requests form that populates the ORG parameters which can be referenced in the value sets.

Complete List of All Submit Request Paramters

Below is the comprehensive list of parameters supported by the "Run Requests"/SRS form and additional information about their usage.

- REQUEST_GROUP_CODE
- REQUEST_GROUP_APPL_SHORT_NAME (used with REQUEST_GROUP_CODE)
- CONCURRENT_PROGRAM_NAME
- PROGRAM_APPL_SHORT_NAME (used with CONCURRENT_PROGRAM_NAME)
- REQUEST_SET_NAME
- SET_APPL_SHORT_NAME (used with REQUEST_SET_NAME)
- SUBMIT_ONCE (default 'N').

  SUBMIT_ONCE can be set to either Y or N (N is the default).

  SUBMIT_ONCE is used in conjunction with CONCURRENT_PROGRAM_NAME or REQUEST_SET_NAME.

  If SUBMIT_ONCE is set to Y, then the form will exit after the Submit button is clicked.
- TITLE
- LOOKUP (default 'N')
- USE_ORG, ORG_ID, ORG_NAME, ORG_CODE, CHART_OF_ACCOUNTS_ID (five parameters)

  If USE_ORG is set to 'Y' (default is 'N') then the Submit Requests form checks to see if the other ORG parameters are set. If the parameters are not set, then it attempts to

populate the parameters from the globals (GLOBAL.FND_ORG_ID, GLOBAL.FND_ORG_NAME, etc.). If the globals have not yet been set, the an ORG LOV shows, and both the parameters and the globals are populated from the LOV.

Values sets should always reference the parameters, not the globals.

- CHAR1, CHAR2, CHAR3, CHAR4, CHAR5
- DATE1, DATE2, DATE3, DATE4, DATE5
- NUMBER1, NUMBER2, NUMBER3, NUMBER4, NUMBER5

  In your value sets, refer to these parameters as:

  ```
  :PARAMETER.CHAR1, :PARAMETER.DATE1, :PARAMETER.NUMBER1  etc.
  ```

Customizing the Submit Requests Window using Codes

You can give the Submit Requests Window a different title, and define the form so that it allows users to select only those reports or concurrent programs belonging to a request group that you have assigned a code to. To do this, you register a form function that references the Submit Requests Window, and you pass certain arguments to the function. Then you construct your menu to include this form function. For more information, see *Oracle E-Business Suite System Security Guide*.

Using a Request Group Code as an argument

The following table describes the parameters passed to associate a request group with the Submit Requests Window and to customize the title of that form. Text is entered in the Parameters field of the Form Functions form.

Parameter Syntax followed by Example	Explanation
REQUEST_GROUP_CODE ="*Request Group Code*" REQUEST_GROUP_CODE = "OE_CONC_PROGRAMS"	This parameter passes the request group's code. (Required)
REQUEST_GROUP_APPL_SHORT_NAME = "*Application short name*" REQUEST_GROUP_APPL_SHORT_NAME = "OE"	This parameter identifies the short name for the application associated with the request group. (Required)
TITLE ="*Application_short_name:Message_Name* " TITLE = "FND:SRS_NEWTITLE"	This parameter identifies a message whose contents define the title, as well as the application short name of that message. (Optional)

Parameter Syntax followed by Example	Explanation
LOOKUP = "Y|N" LOOKUP = "Y"	This parameter indicates whether the TITLE parameter is a message name or a hardcoded string. The default value is "Y", which indicates that TITLE is a message name. (Optional)

Related Topics

Customizing the Submit Requests Window, page 4-30

Organizing Programs into Request Groups, page 4-28

Using Codes with Request Groups, page 4-29

Report Group Responsibilities Report, page 4-33

Request Groups, page 4-65

Report Group Responsibilities Report

This report lists those responsibilities which have access to a report or a request set. Use this report when granting access privileges to reports and request sets, either by assigning reports and request sets to request security groups, or when assigning owners to a request set.

Report Parameters

Application Name

Choose the application name associated with the report or request set.

Report Name/Request Set Name

Either choose the name of a report or request set.

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Organizing Programs into Request Groups, page 4-28

Request Groups, page 4-65

Defining Program Incompatibility Rules

This essay explains how you can define incompatibility rules for your concurrent programs and reports.

Incompatible and Run Alone Programs

When a concurrent program is incompatible with another program, the two programs cannot access or update the same data simultaneously.

When you define a concurrent program, you can list those programs you want it to be incompatible with. You can also list the program as incompatible with itself, which means that two instances of the program cannot run simultaneously.

You can also make a program incompatible with *all other* concurrent programs by defining the program to be run-alone.

There are two types of program incompatibilities, "Global" incompatibilities, and "Domain-specific" incompatibilities.

You can define a concurrent program to be globally incompatible with another program -- that is, the two programs cannot be run simultaneously at all; or you can define a concurrent program to be incompatible with another program in a Conflict Domain. Conflict domains are abstract representations of groups of data. They can correspond to other group identifiers, such as sets of books, or they can be arbitrary.

You define a concurrent program to be run-alone or to be incompatible with specific concurrent programs by editing the concurrent program's definition using the Concurrent Programs window. See: Concurrent Programs, page 4-71.

Program incompatibility and run-alone program definitions are enforced by the Conflict Resolution Manager (CRM).

> **Note:** The concept of "Global" incompatibilities was introduced with Patch 2364876.
>
> With this patch, all pre-existing incompatibilities are converted to the type Global, unless both of the programs have a conflict domain parameter registered. This may mean that if you have been using the Concurrent:Conflicts Domain profile option for your custom programs, you may need to switch the incompatibility type to "Domain-specific" to keep the expected behavior.
>
> Also, the two user-level constraints, set by the Concurrent:Active Request Limit and Concurrent:Sequential Requests profile options, are now enforced globally across all conflict domains.

Request Sets - Incompatibilities Allowed

When you define a request set or request set stage that allows incompatabilities, you create a concurrent program that runs the reports in your request set or stage according to the instructions you entered. Using the Concurrent Programs window, when you list programs as incompatible with a request set, those programs are prevented from starting until all the reports in the set or stage have completed running.

To define incompatibility rules for a request set and request set stage:

- For a request set check the Allow Incompatibility check box on the Request Set window.
- For a request set stage check the Allow Incompatibility check box on the Stages window.
- Navigate to the Incompatible Programs block in the Concurrent Programs form and list those programs that your request set or stage is incompatible with.

All concurrent programs that run request sets are titled *Request Set <name of request set>* while all concurrent programs that run request set stages are titled *Request Set Stage <name of stage>-Request Set <name of request set>*. In the Concurrent Programs form, if you query a request set or stage concurrent program on the basis of the program's name, you must enter in the Name field the words:

- "Request Set" or "Request Set Stage" before the name of a concurrent program
- "Request Set %" to perform a query on all request set and stage programs

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Request Set Incompatibilities, page 4-25

Modifying an Incompatible Programs list, page 4-53

Data Groups, page 4-86

Concurrent Programs, page 4-71

Concurrent Conflict Domains

If two programs are defined as incompatible with one another, the data these programs cannot access simultaneously must also be identified.

In other words, to prevent two programs from concurrently accessing or updating the same data, you have to know *where*, in terms of data, they are incompatible. A Conflict Domain identifies the data where two incompatible programs cannot run simultaneously.

Conflict Domains

In Oracle E-Business Suite, data is stored in database tables that belong to a particular application. Each table may also contain information used to determine what conditions need to be met to access the individual records. These conditions may consist of one or more of the following data groupings:

- SOB - based on the profile option GL_SET_OF_BOOKS
- Multiple installations (referred to as MSOB)
- Multiple Operating units (determined by profile option MO_OPERATING_UNIT) (referred as MULTIORG).
- Multiple Orgs (determined by profile option INV_ORGANIZATION_ID, Used by Manufacturing Applications)
- HR may use business group as a conflict resolution domain
- FA may use FA book
- etc...

A conflict domain is an abstract representation of the groupings used to partition your data. There is no limit to the number of domains that can be defined, but excessive domains may hurt performance.

All programs are assigned a conflict domain when they are submitted. If a domain is defined as part of a parameter the concurrent manager will use it to resolve incompatibilities. If the domain is not defined by a parameter the concurrent manager uses the value defined for the profile option Concurrent:Conflicts Domain. Lastly, if the domain is not provided by a program parameter and the Concurrent:Conflicts Domain profile option has not been defined the 'Standard' domain is used. The Standard domain is the default for all requests.

All programs use the Standard conflict domain unless a value is defined for the profile option Concurrent:Conflicts Domain or a conflict domain is defined through a program parameter.

Each request submitted uses parameters which identify the records that it will access. For programs that are defined with incompatability rules an additional parameter (conflict domain parameter) is used. The conflict domain may be set automatically based on such variables as a login ID, set of books, or the organization the user is working in. The conflict domain parameter may in some cases be selected in the parameters field of the Submit Requests form. Once the parameter is determined the Conflict Resolution Manager (CRM) uses the domain to ensure that incompatible programs do not run simultaneously in the same domain.

Enforcing Incompatibility Rules

Concurrent managers read requests to start concurrent programs running. The Conflict Resolution Manager checks concurrent program definitions for incompatibility rules.

If a program is identified as Run Alone, then the Conflict Resolution Manager prevents the concurrent managers from starting other programs in the same conflict domain.

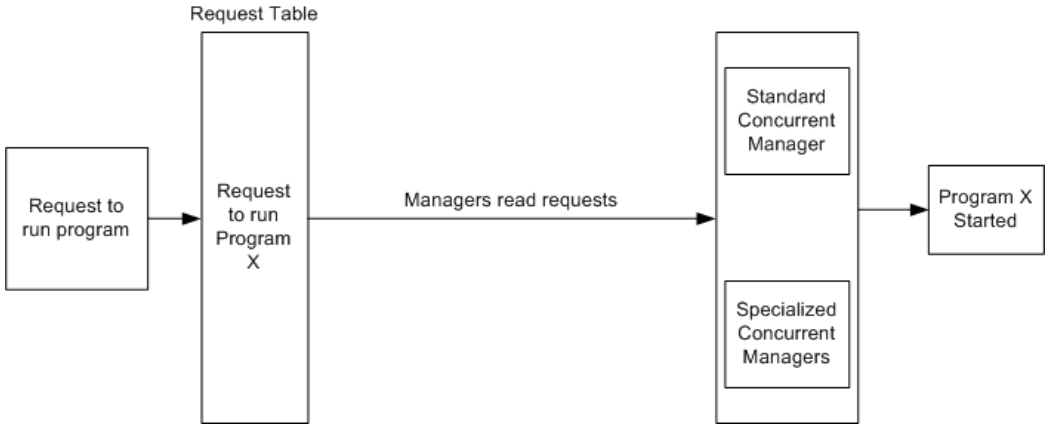
When a program lists other programs as being incompatible with it, the Conflict Resolution Manager prevents the program from starting until any incompatible programs in the same domain have completed running.

The figures below illustrate the role of the Conflict Resolution Manager when enforcing program incompatibility rules.

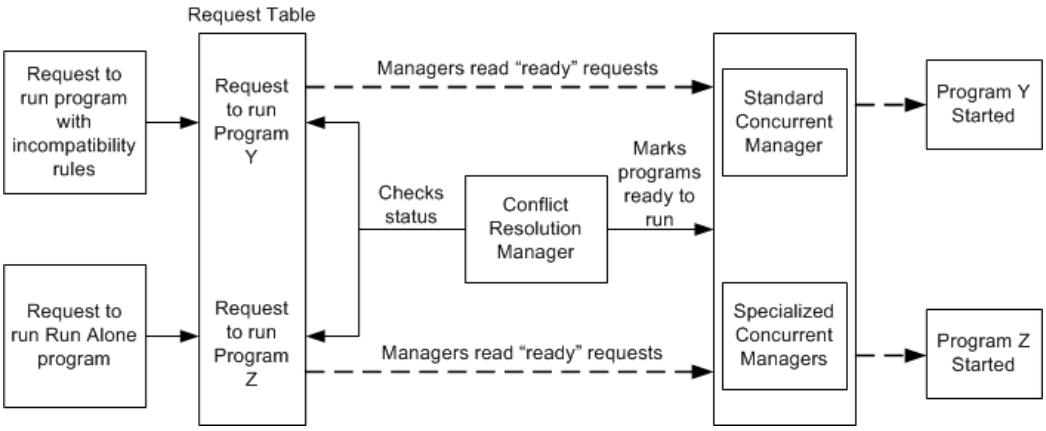
In a simple example without incompatibilities, a user submits a request to run a program. This request is then added to the request table which contains a list of requests. Managers then read requests from this table and start the associated concurrent programs.

A more complex example users may have submitted one request with incompatibility rules and another request to run a program that must be run alone. In this case these requests are added to the request table, but the Conflict Resolution Manager then checks the statuses of the requests in the table and marks which requests are ready to be run. The concurrent managers then read only the "ready" requests and start their concurrent programs.

Simple Run Program Request



Complex Run Program Request



Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Copying and Modifying Program Definitions, page 4-50

Modifying an Incompatible Programs list, page 4-53

Concurrent Programs, page 4-71

Custom Concurrent Programs

This section provides information for system administrators on custom concurrent programs. It explains certain procedures and conventions for creating customized concurrent programs:

- Log and Output Filenames, page 4-39
- Oracle Tool Concurrent Programs, page 4-40
- Pro*C Concurrent Programs, page 4-41
- Host Language Concurrent Programs, page 4-44
- Submitting Concurrent Requests (CONCSUB), page 4-45

For information on creating custom concurrent programs, see the *Oracle E-Business Suite Developer's Guide*.

For information on setting up the development environment , see the *Oracle E-Business Suite Concepts Guide*.

Log and Output Filenames

Log and output files must have specific names and locations for users to review the files online.

If you use the Oracle Application Object Library routine fdpwrt() to write to files, the concurrent managers automatically name the files according to the operating system's naming conventions. This method of writing to files is completely portable. You do not have to rewrite your programs to name your log and output files differently if you port your application to another platform.

Standard Names

Standard names for log and output files are listed in the following table:

File Type	Location	Filename
Log	*Default:* $APPLCSF/$APPLLOG. But if the variable $APPLLDM is set to "product", log files go under $APPLCSF/$<PROD>/$APPLLOG.	l<request ID>.req
Output	*Default:* $<PROD>_TOP/$APPLOUT with Common Directory: $APPLCSF/$APPLOUT	*Default:* <USERNAME>.<request ID> or O<request ID>.out *or* user.out based on value of APPCPNAM

The variable parameters shown in this table have the following values:

- <PROD>_TOP - The application's top environment variable.
- <Request ID> - The number that identifies the concurrent request.
- <USERNAME> - Up to eight characters (uppercase) of the application username of the person who requested the concurrent process.

On UNIX platforms, the filenaming works as described in the following table:

APPCPNAM Variable Assignment in UNIX	Output File Format
APPCPNAM = "REQID"	o999999.out

APPCPNAM Variable Assignment in UNIX	Output File Format
APPCPNAM = "USER"	<Applications user>.out
APPCPNAM = "USER.REQID"	<Applications user>.999999
(unset or unrecognized syntax)	o999999.out

In the above table, <Applications user> refers to an Oracle E-Business Suite user name and '999999' stands for a concurrent request ID.

On Windows platforms, the default format is o<request ID>.out. Setting APPCPNAM to USER on a Windows platform results in the output files having the format <user ID>.out.

Oracle Tool Concurrent Programs

If you write concurrent programs in PL/SQL, SQL*Plus, or Oracle Reports, name the program exactly as you identified it in the Execution File field of the Concurrent Program Executable window, plus an extension if necessary.

The following table lists the file extensions used for these programs and the directories where the programs should reside. (This does not apply to PL/SQL stored procedures, which are stored in the database.) The directories are under your custom application's TOP directory, $<PROD>_TOP.

If you use shared PL/SQL libraries with your Oracle Reports programs, and you want to include the libraries you write for your custom application, place the libraries in the $APPLPLS directory under your custom application's TOP directory.

Tool	Extension	Directory	Comments
SQL*Plus and PL/SQL	.sql	$APPLSQL	The program name is case sensitive and must exactly match the Execution file you defined with Oracle Application Object Library.

Tool	Extension	Directory	Comments
Oracle Reports	.rdf	$APPLREP	Oracle Application Object Library looks for the .rdf file first. It uses the .rex file if it does not find the .rdf file. The program name is case sensitive and must exactly match the execution file name you defined with Oracle Application Object Library.
SQL*Loader	.ctl	$APPLBIN	

Pro*C Concurrent Programs

When you write a concurrent program in Pro*C, copy the skeleton programs EXMAIN.c and EXPROG.c from the directory $FND_TOP/$APPLUSR. Rename the files and globally replace SUBROUTINE_NAME with the name of your subroutine.

EXMAIN.c is the skeleton used for your spawned programs. EXPROG.c is the skeleton used for your program's logic. This module can be used to create a spawned or an immediate program. For immediate programs, you must include your copy of EXPROG.c in a program library. See below for information on building a program library.

You can use programs written with these skeleton programs as spawned or immediate concurrent programs. Spawned programs run as a separate process while immediate programs run linked in with a concurrent manager.

> **Important:** Oracle provides information on immediate concurrent programs for backwards compatibility only. We strongly recommend that you do not create any new immediate concurrent programs. You should define your new Pro*C concurrent program executables as spawned.

Naming Your Executable File

Name your program's executable file exactly as you identified it in the Execution File field of the Concurrent Program Executable window. Put your executable file in the $APPLBIN directory under your application's TOP directory.

Building Your Program Library

Register a new program library with the Register Concurrent Program Library form and register all the programs you want to include in this library. Then enter "Yes" in the Rebuild field and commit. This creates a request to build a new catalog file called <Library Name>.c under $<PROD>_TOP/$APPLLIB$. You should compile the <Library Name>.c file after the request completes.

Sample program libraries such as prgcat.c and prglib.c are located under $FND_TOP/$APPLUSR.

> **Tip:** For ease of maintenance, define your concurrent program executables as spawned.

Compiling C and Pro*C Programs

Your environment for compiling custom code depends on the file $FND_TOP/usrxit/devenv. If you change this file, you should reread it by logging in again so that the changes take effect.

You compile your C or Pro*C programs into object modules using $FND_TOP/usrxit/Makefile. You then link your programs using adrelink. We do not support both compiling and linking executables using a single makefile or utility.

To compile the C program example.c, use the following syntax. In all the examples, you should run the commands from the directory in which your files are located.

```
$ make -f $FND_TOP/usrxit/Makefile example.o
```

To compile the Pro*C program proexamp.pc, use the following syntax:

```
$ make -f $FND_TOP/usrxit/Makefile proexamp.o
```

To compile the four C and Pro*C programs a.c, b.c, c.pc, d.pc (all of which are in the current directory), use the following syntax:

```
$ make -f $FND_TOP/usrxit/Makefile a.o b.o c.o d.o
```

Linking Spawned Concurrent Programs as Stand-alone Programs

If you want your spawned concurrent program to run as a stand-alone program, perform the following steps before compiling your stand-alone executable.

For custom concurrent programs you define under your custom application (as recommended), you should copy the sample.mk file from $FND_TOP/usrxit to your $<PROD>_TOP/$APPLLIB directory. Modify your copy according to the instructions contained in the file. This is the file adrelink uses to link your stand-alone executables.

Then enter the following commands.

```
$ . $FND_TOP/fndenv
```

Move to the directory in which your source files are kept.

```
$ cd <source_directory>
$ make -f $FND_TOP/$APPLLIB/Makefile <source file>.o
```

Here, <source file> is the name of the file containing your program and <directory> is the directory where the source file is located.

You can then link your stand-alone executable and place the executable in the $APPLBIN directory under the TOP directory for your custom application:

```
$ adrelink force=y "<appl_short_name> <program name>"
```

In this relink command, <appl_short_name> is the application short name of the application your program belongs to, and <program name> is the program name.

Linking your Immediate Concurrent Program

To create a program library, you link your compiled library catalog with your program object files using an Oracle Application Object Library link procedure.

> **Important:** Oracle provides information on immediate concurrent programs for backwards compatibility only. We strongly recommend that you do not create any new immediate concurrent programs. You should define your new Pro*C concurrent program executables as spawned.

Make sure the environment variable $LUSRLIB includes the modules that define the functions for the immediate concurrent programs and your program library. Set the $LUSRPRG variable to include the object modules of your library catalog. The file devenv in the directory $FND_TOP/$APPLUSR defines the variables $LUSRLIB and $LUSRPRG. The file fndenv executes devenv.

The files devenv and fndenv are UNIX shell scripts that set up the necessary environment variables.

We recommend that you make a copy of the working program library before linking your new immediate concurrent program library in case your new program library does not function as expected. To link your program library, execute this command from the operating system:

```
$ adrelink force=y "fnd UFNDLIBR"
```

This creates your new program library as UFNDLIBR. You can rename it, but the name of your new program library must be eight characters or less.

Testing Pro*C Concurrent Programs

You can use the following method to test your program. You must pass each argument needed by your program. To pass parameters, enter the following at the operating system prompt:

```
$ <program name> <ORACLE username>/<ORACLE password> 0 Y \
[<parameter 1> <parameter 2>... ]
```

Instead of the Oracle username and password, you can use an Oracle E-Business Suite username and password, if the corresponding user has the System Administrator responsibility.

The program name must be uppercase and the same name that you entered in the Execution File field of the Concurrent Program Executable window. The 0 and Y arguments are required.

If any of your program-specific parameters includes spaces, enclose that parameter in double quotes. If a parameter contains a literal double quote, precede that mark with a backslash [\].

Host Language Concurrent Programs

Name your program `<name>.prog`, where `<name>` is the value you enter in the Execution File field of the Concurrent Executable window. Then make a symbolic link using your execution file name (without an extension) to `fndcpesr` , which is located in the $FND_TOP/$APPLBIN directory. Put your executable file and the linked file in the $APPLBIN directory under your application's TOP directory.

For example, name your custom shell script `CUSTOM.prog`. Create a symbolic link to `fndcpesr` named CUSTOM. Place both files in your $APPLBIN directory. Create your concurrent program executable using the execution file CUSTOM.

Host Program Parameters

The concurrent manager running your program puts your program name in `$0`, the four arguments orauser/pwd, userid, username, and request_id in `$1` to `$4`, and your program specific parameters in `$5` and beyond. Each of these arguments can be at most 50 characters.

For example, if you pass two parameters into your program, you use `$5` to refer to the first parameter and `$6` to refer to the second parameter.

Protecting Your Oracle User Password

In some cases, there are security concerns with passing your Oracle username and password on the command line to your HOST program. If you do not want the concurrent manager to pass your username/password to your program, you can have the manager pass it as an environment variable instead. Alternatively, you can not pass it at all.

> **Note:** Passing the username/password in an environment variable is not any more secure than passing it on the command line.

First, define your concurrent program executable as a HOST program in the Concurrent Program Executable form.

To have the username/password passed as an environment variable, enter the term

'ENCRYPT' in the Execution Options field of the Concurrent Programs window when defining a concurrent program using this executable. 'ENCRYPT' signals the concurrent manager to pass the username/password in the environment variable fcp_login. The argument `$1` is left blank.

If you do not want the username/password passed to the program at all, enter 'SECURE' in the Execution Options field. The concurrent manager will not pass the username/password to the program.

Success Codes

By default, a shell script returns success (status code 0). If your script traps an error, use the UNIX exit command "exit 1" to return failure (status code 1) to the concurrent manager running the program.

Log and Out Files

Use names in FCP_LOG and FCP_OUT. In this way the log and output/report files can be viewed online.

Testing Your Program

You should test using the `<name>.prog` file to make sure your script behaves correctly.

Submitting Concurrent Requests (CONCSUB)

You can test your concurrent program by submitting the program using the CONCSUB utility from the operating system.

Syntax

You can submit a concurrent request to run any concurrent program by running the CONCSUB program with the following syntax:

```
$ CONCSUB <APPS username>[/<APPS password>] \
<responsibility application short name> \
<responsibility name> \
<username> \
[WAIT=N|Y|<n seconds>] \
CONCURRENT \
<program application short name> \
<program name> \
[PROGRAM_NAME="<description>"] \
[REPEAT_TIME=<resubmission time>] \
[REPEAT_INTERVAL= <number>] \
[REPEAT_INTERVAL_UNIT=< resubmission unit>] \
[REPEAT_INTERVAL_TYPE=< resubmission type>] \
[REPEAT_END=<resubmission end date and time>] \
[NLS_LANGUAGE=<language of the request>] \
[NLS_TERRITORY=<territory of the request>] \
[START=<date>] \
[IMPLICIT=< type of concurrent request> \
[<parameter 1> ... <parameter n>]
```

> **Note:** Passing the APPS password above is optional. It can be passed using standard I/O.

For parameters that follow the CONCURRENT parameter and include spaces, enclose the parameter argument in double quotes, then again in single quotes. Oracle Application Object Library requires this syntax because it parses the argument string twice. For example, to pass this argument to a program:

```
This is an example
```

pass this argument through CONCSUB:

```
'"This is an example"'
```

Example

Here is an example of the command to run CONCSUB:

```
echo "$password" | CONCSUB APPS \
SYSADMIN \
"System Administrator" \
SYSADMIN \
WAIT=N \
CONCURRENT \
FND \
FNDFMRTC \
PROGRAM_NAME='"Register Custom Tables Weekly"' \
REPEAT_INTERVAL=7 \
REPEAT_INTERVAL_UNIT="DAYS" \
REPEAT_INTERVAL_TYPE="START" \
START='"08-JUN-14 23:55:00"'
CGL
APPLSYS
ALL
CGL
```

Parameters

The following entries explain the required and optional parameters for submitting a concurrent program with CONCSUB. Default values are listed to the right.

username/ password
Required. The ORACLE username and password that provides access to the data that your program uses. Alternatively, an Oracle E-Business Suite username and password for a user with the System Administrator responsibility. Another option is "Apps:User" which then requires the password of the FND_USER in the parameter *username*.

responsibility application short name
Required. The application short name of the responsibility whose concurrent processing options you want to use.

responsibility name
Required. The name of your responsibility. If the name of

your responsibility includes spaces, enclose that name in double quotes.

username

Required. The uppercase username of the application user whose concurrent processing options you want to use.

WAIT

Optional. A flag that indicates whether to wait for the submitted request to complete. If you leave this parameter out, the default value of N makes CONCSUB return you to the operating system prompt without waiting for your request to complete.

Set WAIT=Y to have CONCSUB check the request status every 60 seconds and return you to the operating system prompt when your request is completed. You can also enter an integer value for a number of seconds, as in WAIT=30, for CONCSUB to check for request completion every <number> seconds.

> **Important:** Using WAIT=Y or WAIT=<number> requires that your request completes before CONCSUB returns you to the operating system. If the concurrent manager is down, your CONCSUB process waits indefinitely until the concurrent manager is started and the request completes.

CONCURRENT

Required. A flag that separates the program-specific parameters from the operating system parameters.

program application short name

Required. The application short name of your concurrent program.

program name

Required. The uppercase name of your program. It must be the short name that you enter in the Concurrent Programs window when defining a concurrent program.

PROGRAM_ NAME

Optional. A descriptive name for your program. The program field on the View Requests form displays this as the user-friendly program name. The concurrent program short name passed to CONCSUB is often hard for end users to understand, so the PROGRAM_NAME parameter allows you to pass a more easily remembered name for your concurrent program. If you do not specify a PROGRAM_NAME, the View Requests form displays the

user-friendly program name specified in the Concurrent Programs window.

You may also use the PROGRAM_NAME parameter to indicate the batch that your request processes for programs that process a set of data, where there could be several requests for a given program that are active at the same time.

REPEAT_TIME Optional. The time of day to resubmit the request. The format for the time is HH24:MI or HH24:MI:SS. For example, REPEAT_TIME=14:30 resubmits your request daily at 2:30 p.m.

> **Important:** Do not use REPEAT_TIME with other resubmission parameters except for the optional parameters REPEAT_END and START.

REPEAT_ INTERVAL Optional. The interval between resubmission (a positive integer or real number). Use this parameter along with REPEAT_INTERVAL_UNIT to specify the time between resubmissions.

REPEAT_ INTERVAL_ UNIT Optional. The unit of time used for the interval between resubmissions. The available units are MINUTES, HOURS, DAYS or MONTHS. Use this parameter along with REPEAT_INTERVAL to specify the time between resubmissions. For example, setting REPEAT_INTERVAL=12 and REPEAT_INTERVAL_UNIT=HOURS resubmits your request every twelve hours. The default value is DAYS.

> **Important:** Do not use REPEAT_INTERVAL and REPEAT_INTERVAL_UNIT with REPEAT_TIME.

REPEAT_ INTERVAL_ TYPE Optional. Whether to time the resubmission interval from the requested start time of the request or from its completion. Set this parameter either to START or END. The default value is START.

> **Important:** Use

REPEAT_INTERVAL_TYPE only if you use REPEAT_INTERVAL.

REPEAT_END

Optional. The date and time to stop resubmitting the concurrent request. Use one of the following for the format of the end date:

'"DD-MON-RR HH24:MI:SS"' (as in '"07-APR-14 18:32:05"')

or

'"DD-MON-RRRR HH24:MI:SS"' (as in '"07-APR-2014 18:32:05"')

Note that because this date format includes a space, you must enclose the date in double quotation marks and single quotation marks. You can also specify just the date:

'DD-MON-RR'

or

'DD-MON-RRRR'

NLS_LANGUAGE

Optional. The NLS language for the request.

NLS_TERRITORY

Optional. The NLS territory for the request.

START

Optional. A start date and time for your program in this format:

'"DD-MON-RR HH24:MI:SS"' (as in '"07-APR-02 18:32:05"')

Because this date format includes a space, you must enclose the date in double quotation marks and single quotation marks. If you do not specify a start time, your program submits immediately and is processed by the next available concurrent manager. The default value is the current time.

IMPLICIT

Optional. Whether to show this concurrent request on the View Requests form. Specify NO, YES, ERROR or WARNING. The value IMPLICIT=NO allows the request to appear on the View Request form. The default value is NO.

The value IMPLICIT=YES means that only the System Administrator's privileged View Concurrent Requests form displays this request. Use this value if the request is not interesting to the user.

Specify IMPLICIT=ERROR or IMPLICIT=WARNING,

respectively, if you want the request to appear only if it fails or completes with warnings.

REPEAT_DAYS

Optional. The number of days after which to repeat the concurrent request, calculated from the last requested start date. The number can be a positive integer or real number. For example, REPEAT_DAYS=1.5 resubmits your request every 36 hours.

> **Important:** Do not use REPEAT_DAYS with other resubmission parameters except for the optional parameters REPEAT_END and START.

> **Tip:** REPEAT_DAYS will become obsolete in a future release. You may therefore want to use REPEAT_INTERVAL, REPEAT_INTERVAL_TYPE and REPEAT_INTERVAL_UNIT instead of REPEAT_DAYS.

parameter 1 ... parameter *n*

Optional. Your program-specific parameters. If a parameter includes spaces, enclose that parameter in double quotes, then in single quotes. If a parameter contains a double quotation mark as part of the argument, precede that mark with a backslash [\].

Copying and Modifying Program Definitions

These sections explain how you can copy and modify concurrent program definitions.

> **Warning:** Do not overwrite program definitions for existing concurrent programs. Copy the program, rename it, then make any desired modifications to the new program.

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Copying and Renaming a concurrent program, page 4-51

Defining Program Incompatibility Rules, page 4-34

Alter Program Priority, page 4-52

Modifying an Incompatible Programs list, page 4-53

Concurrent Program Parameters, page 4-53

Example of modifying a program's parameters, page 4-60

Concurrent Programs, page 4-71

Warnings for Modifying Program Definitions, page 4-58

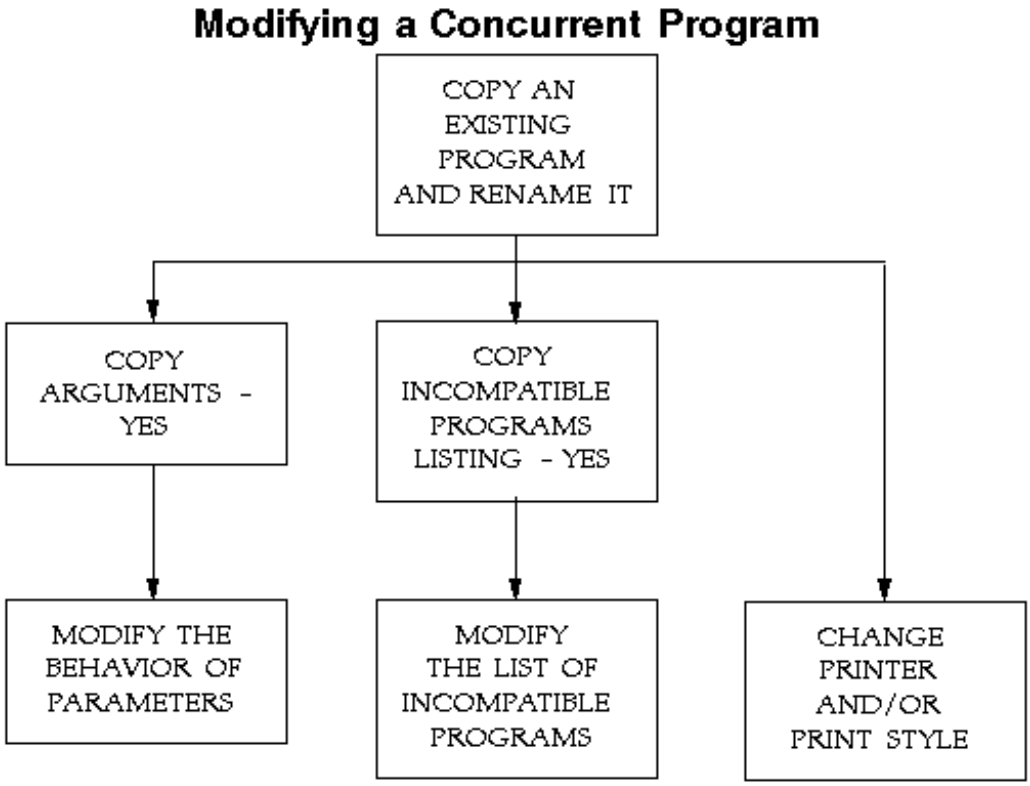
Copying and Renaming a concurrent program

You can copy your concurrent programs and modify them to create new programs with definitions that meet your needs. You can modify how a concurrent program operates by changing the program's definition of:

- incompatible programs
- parameters (arguments)
  - parameter value sets
- printer, print style, etc.

Rather than overwrite a concurrent program's definition, you should customize a program by copying and renaming an existing program, then modifying the new program to suit your needs. The figure below illustrates the basic steps in copying and modifying a new concurrent program.

As the figure illustrates, you can copy parameters, and then modify the behavior of the parameters. Or you can copy the list of incompatible programs, and then modify the list. Finally , you can change the associated printer and/or print style.



Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Copying and Modifying Program Definitions, page 4-50

Example of modifying a program's parameters, page 4-60

Concurrent Programs, page 4-71

Alter Program Priority

You may wish to control the priority of some requests on a program level rather than at the user level.

Setting the priority for a program allows any request to run that concurrent program to use your selected priority rather than the priority of the user submitting the request.

For example, a user can submit a variety of requests at the standard priority determined by the value of the user profile Concurrent:Priority. However, when the user submits a request for a particular concurrent program, you want that request to have a higher priority.

You assign that program a priority of 10. When the user requests that program to run, it receives the higher priority defined on the Concurrent Program window rather than the

user's standard priority and is processed ahead of other requests. When the users requests other concurrent programs that do not have a specified priority, those requests use the user's Concurrent:Priority profile value.

Related Topics

Copying and Modifying Program Definitions, page 4-50

Concurrent Programs, page 4-71

Modifying an Incompatible Programs List

A concurrent program's definition may include a list of incompatible programs. When a program is listed as incompatible with another program, the two programs cannot run simultaneously in the same conflict domain. See: Defining Program Incompatibility Rules, page 4-34.

You can view which programs are incompatible with a concurrent program from the Incompatible Programs block on the Concurrent Programs window. The programs listed cannot run simultaneously within the same conflict domain as the concurrent program whose definition you are viewing.

To modify the list of incompatible programs you can either:

Add new programs to the list.

The *Scope* field refers to whether you want the program by itself to be incompatible, or whether you want the program and all child requests, that is, concurrent programs started by the program as part of a request set, to be incompatible.

- Delete programs from the list.

> **Important:** To immediately effect any changes you make in the Incompatible Programs zone, you must navigate to the Administer Concurrent Managers window and choose *Verify* for the Internal Concurrent Manager.

Related Topics

Copying and Modifying Program Definitions, page 4-50

Concurrent Programs, page 4-71

Administer Concurrent Managers, page 4-151

Concurrent Program Parameters

Parameters, also referred to as *arguments*, are assigned to standard submission concurrent programs. To define a program as standard submission, set the value of the Standard Submission field in the Concurrent Programs form to Yes.

Important: All the mechanisms for parameter defaulting (including references to values of other parameters, user profiles, etc.) are evaluated only at submission time.

There are two aspects to a parameter associated with a concurrent program: its value set and its behavior.

Parameter value set The valid values the parameter can accept. The set of valid values is referred to as a *value set*.

Parameter behavior How the parameter behaves within an application. For example, whether:

- an entry value for the parameter is required in order for the program to work
- the parameter is displayed to the end user
- a default value is automatically provided for the parameter

If you wish to define or modify a value set, you must first carefully plan your value set's purpose and implementation.

Using the Concurrent Programs form, you can see a concurrent program's parameters by choosing *Parameters*. Each parameter has a value set that defines what values are permissible for the parameter. To see the name of a parameter's value set, look at the Value Set field in the Argument Details block.

Related Topics

Copying and Modifying Program Definitions, page 4-50

Control the Behavior of Request Parameters, page 4-54

Example of modifying a program's parameters, page 4-60

Concurrent Programs, page 4-71

Control the Behavior of Request Parameters

The behavior of parameters in programs running individually may differ from when those programs are run as part of a request set.

You define how a program's parameters behave when you define the program using the Concurrent Programs form.

Using the Request Set form, you can also define how a program's parameters behave when the program is run as part of a request. In addition, you can define parameters in different programs in a request set to all share the same value by labeling them as

Shared Parameters. See: Sharing Parameters in a Request Set, page 4-26.

> **Warning:** Modifying a concurrent program's definition by adding new or deleting existing parameters, or changing a parameter's value set can prevent the program from running. See: Warnings for Modifying Program Definitions, page 4-58.

Not Displaying Parameters

Using the Concurrent Programs form or the Request Set form, you can set a parameter so it does not display to an end user. Because parameters that do not display cannot be modified, setting a parameter to not display:

- is a good security measure, guaranteeing a desired default value is used
- means you should enter a valid default type and value at either the program's definition, or if the program is part of a request set, at the request set's definition.

> **Warning:** Set defaults for required parameters before setting the Display field to No. Otherwise the Submit Requests form returns an error when attempting to submit the program.

If you define a parameter to not display, then the parameter does not appear when the program is run using the Submit Requests form, nor does it appear in the Request Set form.

If you define a parameter to not display, using the Request Set form, then the parameter does not appear on the Submit Requests form when the program is run as part of a request set.

Viewing displayed parameters after a request is submitted

After a request is submitted to run a concurrent program, the program's parameters may be displayed in the Details block of the Concurrent Requests form.

When a parameter is set to not display, it does not appear in the Details block of the Concurrent Requests form.

These displayed parameter values exactly match the values that the concurrent manager passes to the concurrent program, and may or may not correspond to the displayed value that the user chose.

For example, in the Submit Requests form, the user may choose "Oracle General Ledger" as a parameter, but the corresponding application ID displays in the Concurrent Requests form.

> **Tip:** If your users encounter errors when running a program, you can

look at the exact values that the concurrent program uses to help you diagnose the problem.

Setting Default Values for Parameters

Parameter default values can be changed by users when they submit a program or request set to run.

You can set a default value for a parameter using the:

- Default Type and Default Value fields in the Concurrent Programs form. These values cannot be changed on the Request Set form.
- Default Type and Default Value fields in the Request Set form.

  This default definition applies only when the program is run as part of a request set.
- Shared Parameter and Default Value fields in the Request Set form

  This default definition applies only when the program is run as part of a request set. All parameters labeled with the *same shared parameter label* default to the value you set in the Default Value field.

Entering erroneous default values

If the Default Type or Default Value for a parameter is incorrect, when the program is being set to run using the Submit Requests form, a window displays along with an error message.

If the parameter is not displayed, you receive an error message. You cannot update a field that is not displayed.

> **Warning:** Be careful when entering the default type and default value, because these values are not validated with the value sets for your parameters. If you enter incorrect values, they do not appear as defaults when you run this request set using the Submit Requests form.

Preventing modification of parameter values in a Request Set

If a parameter is displayed in the Request Set form and there is no default value provided by the program's definition, you can define a default value or have the parameter inherit a shared value, and then prevent end users from modifying that value.

Set the Modify field in the Request Set form to No if you want to show the value for a parameter but not allow changing it when the request set is run using a Standard Submission form. You can set a value for the parameter using a default value or a

shared parameter.

If the Display field is set to No, the Modify field automatically defaults to No, and you cannot update it.

> **Caution:** Set defaults for required parameters before turning Modify to No. Otherwise the Submit Requests form returns an error when attempting to submit this report.

Changing responsibility to see changes take effect

Modifying parameter behavior, for example, changing whether a parameter is displayed to the end user, takes effect immediately after you commit your change. However, some changes do not appear to you unless you change responsibility or select your current responsibility again.

Behavior of Program Parameters

The following table describes how a parameter's details affect its behavior in the Concurrent Programs form and the Run Requests form.

Parameter Details	Concurrent Programs form	Run Requests form
Required	Yes	Parameter requires a value (entered by user or a default).
Display	Yes	Parameter is displayed.
Display	No	Parameter is not displayed, and cannot be modified.
Default Type & Value	Yes - Default Type and Value entered.	A default value displays, and can be changed by the user.
Default Type and Value	No default entered.	No default value is displayed.

The following table describes how a parameter's details affect its behavior in the Request Sets form and Run Requests form.

Parameter Details	Concurrent Programs form	Request Set form	Run Requests form
Required	Yes	Parameter does not require a value.	Parameter requires a value.
Display	Yes	Parameter is displayed. - Display set to Yes.	Parameter is displayed.
Display	Yes	Parameter is displayed. - Display set to No.	Parameter is not displayed.
Display	No	Parameter not displayed.	Parameter not displayed.
Modify	n/a	Yes	Value can be modified.
Modify	n/a	No	Value cannot be modified.
Default Type & Value	Yes - Default Type and Value entered.	Default Type and Value cannot be modified.	Default values can be changed by the user.
Default Type & Value	No default entered.	Yes - a Default Type and Value can be entered.	Default values can be changed by the user.
Default Type & Value	No default entered.	No - Default Type and Value are not entered.	No default value is displayed.

Warnings for Modifying Program Definitions

The following table lists warnings for modifying program definitions:

Action	Form Used	Warning
Changing the number of columns or rows in a report program.	Concurrent Programs - Report Information region.	Some report programs are written to produce a precise output. Changing the output definition could prevent the program from running, or produce poor results.
Setting print style to Dynamic.	Concurrent Programs - Report Information region - Style field.	Dynamic print style informs the program to generate its output based on output dimensions that may vary. Special coding within a program is required to support the Dynamic print style.
Changing the number of parameters in a program definition.	Concurrent Programs - Parameters window.	Programs are defined to expect x number of parameters. If you add a new parameter (x + 1), the program will ignore it. Deleting a parameter can cause a program not to run.
Changing Value Sets.	Concurrent Programs - Argument Details region - Value Set field.	Programs expect values of a certain type and length. Programs may not operate if value set is changed.
Changing tokens.	Concurrent Programs - Argument Details region - Token field.	Programs expect values of a certain type and length. Program may not operate if expected token is not received.
Defining a concurrent executable or program's execution method as Immediate.	Concurrent Program Executables - Execution Method field. Concurrent Programs - Executable Information region - Method field.	Concurrent programs whose execution method is Immediate must be registered with the program library FNDLIBR. Application developers can register programs with program libraries, System Administrators cannot.

Related Topics

Copying and Modifying Program Definitions, page 4-50

Concurrent Program Parameters, page 4-53

Example of modifying a program's parameters, page 4-60

Concurrent Program Details Report, page 4-63

Example of modifying a program's parameters

Consider the following example of when and how to modify a concurrent program's parameters.

If one user submits a large number of concurrent requests on a daily basis, for example, an Oracle Bill of Materials or Oracle Purchasing supervisor, you can create a streamlined purge program that *only* purges that user's concurrent processing records.

You can run this program as System Administrator and have it automatically resubmitted on a specific time interval.

You could also create a request set containing this one program and define the user as the owner of the request set. Then, if you do not assign the request set to any report security group, only the user (owner) can run the program. This way, the user can be responsible for purging their own records.

The System Administrator's Purge Concurrent Request and/or Manager Data program contains twelve parameters. You can copy, rename, and modify the program so it displays only three parameters, with only one parameter requiring user entry. See: Purge Concurrent Request and/or Manager Data, *Oracle E-Business Suite Maintenance Guide*.

The table below summarizes the steps to follow in our example.

Form Used	Task
Concurrent Programs (Concurrent Programs Define)	Query the Application Object Library program named "Purge Concurrent Request and/or Manager Data" and press Copy. Select both Copy Arguments and Copy Incompatible Programs. Enter a new name for the program you are going to copy, for example, enter JSMITH PURGE.
Concurrent Programs	To modify the JSMITH PURGE program's parameters, select the Parameters button.

Form Used	Task
Concurrent Programs, Parameter Window	Modify the following seven parameters so they do not display (user JSMITH cannot see nor change the program's default values). - Oracle ID - Program Application - Program - Manager Application - Manager - Responsibility Application - Responsibility Modify the following three parameters so they do not display (user JSMITH cannot see nor change the default values you set). Set the parameters to the following (Type=Constant) defaults: - Entity = Request - Mode = Age - User Name = JSMITH Leave the following two parameters unchanged so they display. Mode Value will require JSMITH to enter a value, and Report is set to a default value of "Yes". - Mode Value - Report
Request Set (Reports Set)	Create a request set with one program in it, the JSMITH PURGE program. Enter JSMITH in the Owner field. If this request set is not assigned to any report security group, only JSMITH will be able to run the JSMITH PURGE program.
Standard Request Submission program form. For example, the Run Reports form (Reports Run)	When first submitting the JSMITH PURGE program to run, navigate to the Resubmission Options region and enter, for example, "5" and "Days" in the Interval field.

Related Topics

Copying and Modifying Program Definitions, page 4-50

Concurrent Program Parameters, page 4-53

Control the Behavior of Request Parameters, page 4-54

Concurrent Program Details Report, page 4-63

Conflict Domains

A conflict domain is a set of related data stored in one or more ORACLE schemas and linked by grants and synonyms. Do not confuse logical databases with your ORACLE database. The ORACLE database contains all your Oracle E-Business Suite data, with each application's data usually residing in one ORACLE schema. You can think of a

logical database as a line drawn around a set of related data for which you wish to define concurrent program incompatibilities. In other words, logical databases determine which concurrent programs cannot run at the same time.

Logical Databases and Program Incompatibilities

When an ORACLE schema is identified as belonging to a logical database, concurrent program incompatibility rules are enforced when concurrent programs connect to the ORACLE schema.

By checking for incompatibilities between programs running concurrently, accessing the same data, Oracle E-Business Suite ensures that data retrieved by one program is not incorrect or adversely affected when retrieved by another program.

Example - Program Incompatibilities

An example of a concurrent program that is incompatible with other concurrent programs is Oracle General Ledger's Posting program, used to post journal entries.

If the Posting program's incompatibility with other Oracle E-Business Suite concurrent programs were not enforced, other financial reports running simultaneously with the Posting program could contain incorrect account balance information. Logical databases ensure that this does not happen.

Defining Logical Databases

The installation process automatically defines logical databases and assigns ORACLE schemas to them.

A *Standard* logical database can be assigned to every Oracle E-Business Suite product so that every concurrent program, if incompatible with any other program, does not run concurrently with that program, regardless of which ORACLE schema those two programs connect to. Assigning every ORACLE schema to the same (e.g., Standard) logical database is a fail-safe method of enforcing program incompatibility rules.

You must define new logical databases only if you build a custom application whose data do not interact with data found in existing logical databases.

As a general rule, you should define a logical database for each custom application, and assign that application's ORACLE schema(s) to the corresponding logical database.

However, if a custom application's data interacts with another application's data, you should assign the two applications' ORACLE schemas to the same logical database.

Registering your custom application's tables ensures that the table names appear as QuickPick values in the Define Alerts form.

Concurrent Program Details Report

This report documents concurrent program definitions, including executable file information, execution method, incompatible program listings, and program parameters. If a concurrent program generates a report, column and row information, as well as print output and print style, are also documented.

Use this report when considering concurrent program modifications, such as modifying program incompatibility rules.

Report Parameters

Caution: If you do not enter any parameters, the report returns values for *all* concurrent programs, and may be very lengthy.

Application Name

Choose the application name associated with the concurrent program whose program definition details you wish to report on.

Choose only an application name, without a program name, if you wish to run a program definition details report on all concurrent programs associated with an application.

Program

Choose the name of a concurrent program whose program definition details you wish to report on. You must enter a value for Application Name before entering a value for Program.

Report Headings

The report headings display the specified report parameters and provide you with general information about the contents of the report.

Concurrent Programs Report, page 4-63

Concurrent Programs Report

This report shows which concurrent programs are currently enabled nand which programs are disabled.

Use this report to record the execution method, argument method, run alone status, standard submission status, request type, and print style information associated with your concurrent programs.

Report Parameters

Application Name

Choose the application name associated with the concurrent programs whose program information you wish to report on.

If you do not enter an application name, the report will return values for *all* concurrent programs.

Report Headings

The report headings display the specified report parameters and provide you with general information about the contents of the report.

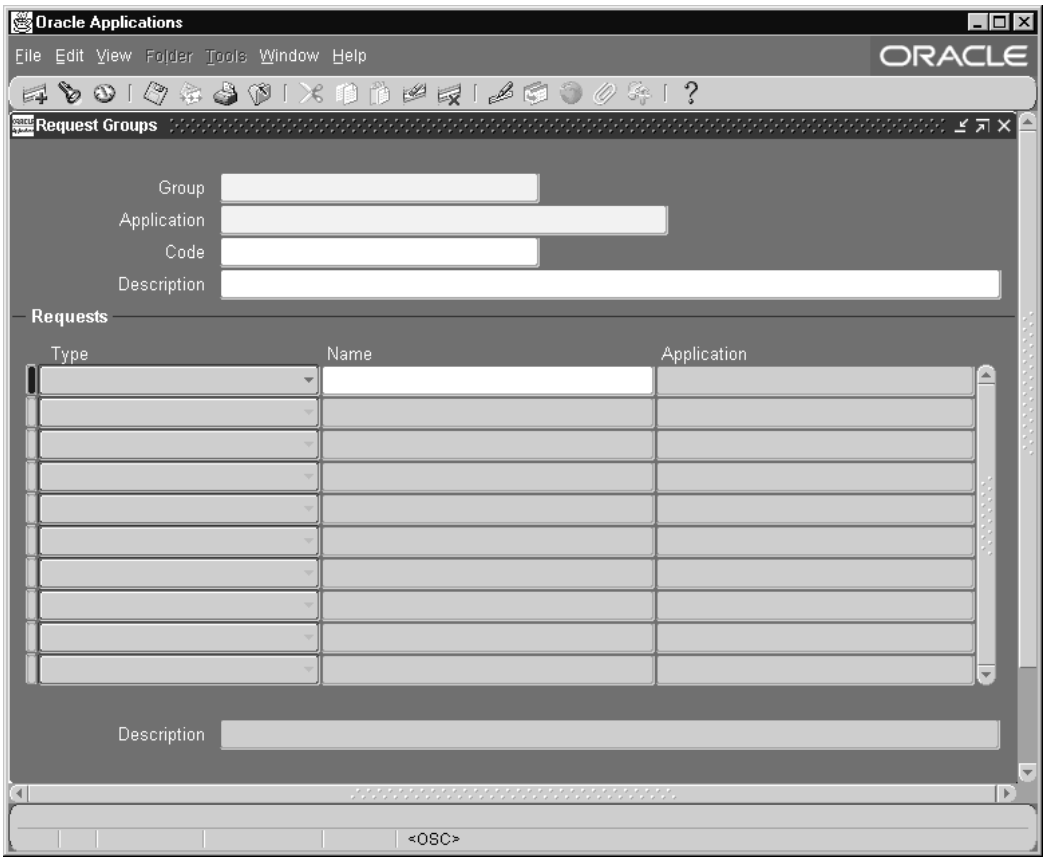
Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Concurrent Program Details Report, page 4-63

Concurrent Programs, page 4-71

Request Groups Window



Use this window to define a request group. A request *security* group is the collection of requests, request sets, and concurrent programs that a user, operating under a given responsibility, can select from the Submit Requests window.

System Administrators:

- Assign a request security group to a responsibility when defining that responsibility. A responsibility without a request security group cannot run any requests using the Submit Requests window.

- Can add *any* request set to a request security group. Adding a private request set to a request security group allows other users to run that request set using the Submit Requests window.

Users:

- Can create their own private request sets using the Request Sets window. In a private request set, users can include only the requests you assign to their request security group.

- Cannot update another user's private request set using the Request Sets window.
- Cannot delete a private request set if it is assigned to a request security group.

Request Groups Block

Enter the following fields.

Group

Use the request group's name to assign the request group to a responsibility on the Responsibilities window. An application name and request group name uniquely identify a request group.

Application

Select the name of the application you wish to associate with your request group. An application name and a request security group name uniquely identify a request security group. This application name does not prevent you from assigning requests and request sets from other applications to this request group.

Code

Assign a code to this request group. Some products use the request group code as a parameter that identifies the requests a customized standard submission form can select. See: Customizing the Submit Requests Window using Codes, page 4-32.

Requests Block

Specify the requests and request sets in the request group.

Type

Choose program or set to add one item, or choose application to include all requests in an application.

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

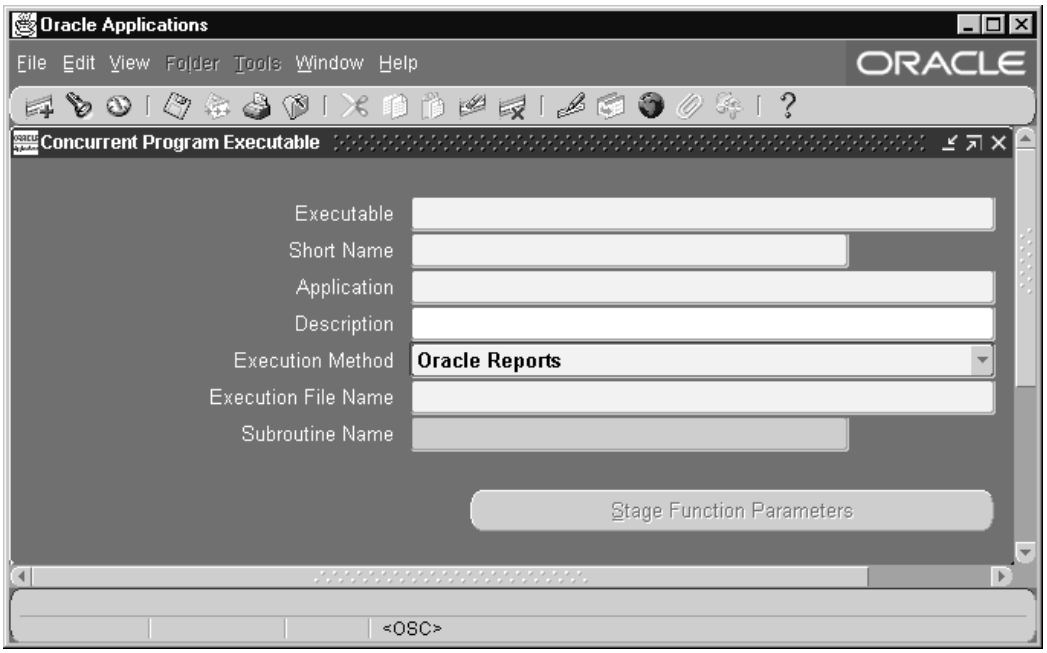
Organizing Programs into Request Groups, page 4-28

Using Codes with Request Groups, page 4-29

Customizing the Submit Requests Window using Codes, page 4-32

Report Group Responsibilities Report, page 4-33

Concurrent Program Executable Window



Define a concurrent program executable for each executable source file you want to use with concurrent programs. The concurrent program executable links your source file logic with the concurrent requests you and your users submit to the concurrent manager.

> **Important:** You cannot add new immediate programs to a concurrent manager program library. We recommend that you use spawned concurrent programs instead.

Concurrent Program Executable Block

The combination of application name plus program name uniquely identifies your concurrent program executable.

See: Concurrent Programs Window, page 4-71

Executable

Enter a name for your concurrent program executable. In the Concurrent Programs window, you assign this name to a concurrent program to associate your concurrent program with your executable logic.

Short Name

Enter a short name for your concurrent program executable.

Application

The concurrent managers use the application to determine in which directory structure to look for your execution file.

Execution Method

The execution method cannot be changed once the concurrent program executable has been assigned to one or more concurrent programs in the Concurrent Programs window.

The possible execution methods are:

Host	The execution file is a host script.
Oracle Reports	The execution file is an Oracle Reports file.
PL/SQL Stored Procedure	The execution file is a PL/SQL stored procedure.
Java Stored Procedure	The execution file is a Java stored procedure.
Java Concurrent Program	The execution file is a program written in Java.
Multi Language Function	The execution file is a function (MLS function) that supports running concurrent programs in multiple languages (as well as territories and numeric character settings).
SQL*Loader	The execution file is a SQL script.
SQL*Plus	The execution file is a SQL*Plus script.
Spawned	The execution file is a C or Pro*C program.
Immediate	The execution file is a program written to run as a subroutine of the concurrent manager. We recommend against defining new immediate concurrent programs, and suggest you use either a PL/SQL Stored Procedure or a Spawned C Program instead.
Request Set Stage Function	PL/SQL Stored Function that can be uesd to calculate the completion statuses of request set stages.

Execution File Name

Enter the operating system name of your execution file. Some operating systems are case sensitive, so the name entered here should match the file name exactly.

Do not include spaces or periods (.) in the execution file name, unless the execution method is PL/SQL stored procedure or Request Set Stage Function.

The maximum size of an execution file name is 60 characters.

Subroutine Name

Enter the name of your C or Pro*C program subroutine here. Do not use spaces or periods (.) in this field.

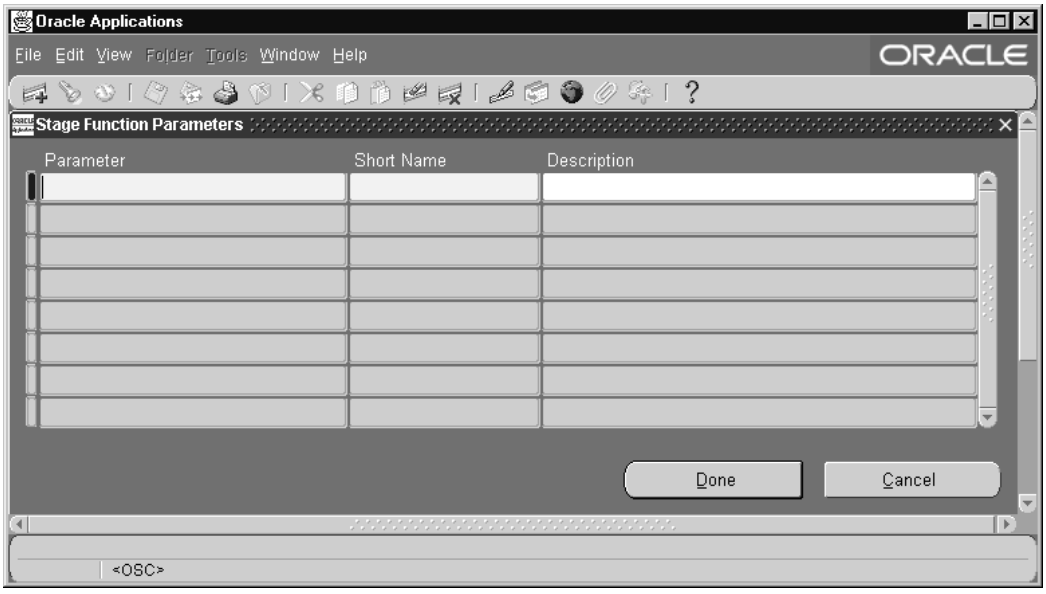
Only immediate programs or spawned programs using the Unified C API use the subroutine field.

We recommend against defining new immediate concurrent programs, and suggest you use either a PL/SQL Stored Procedure or a Spawned C Program instead.

Stage Function Parameters

The Stage Function Parameters button opens a window that allows you to enter parameters for the Request Set Stage Function. This button is only enabled when you select Request Set Stage Function as your Execution Method.

Stage Function Parameters Window



List the Parameters that your custom Stage Function uses.

Parameter

Enter a name for the Parameter. This name will be displayed in the Stage Functions Parameter window of the Request Set form.

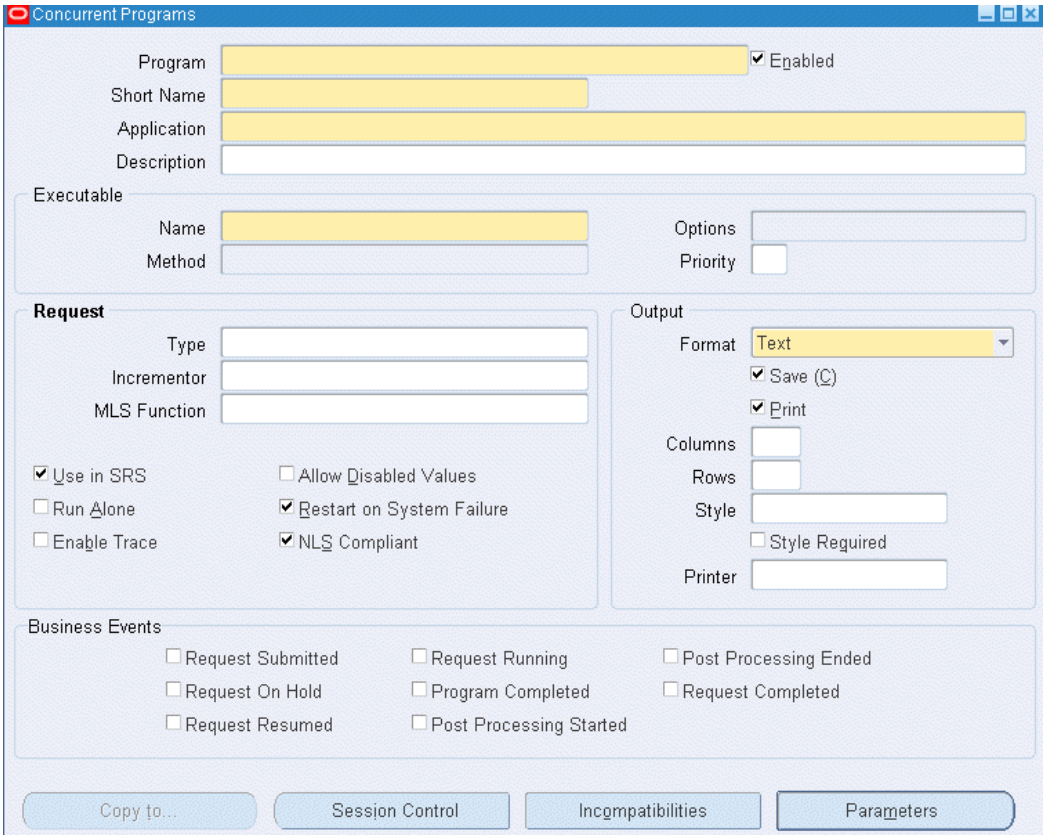
Short Name

Enter a short name that will be used by the function to reference the parameter.

Related Topics

Concurrent Programs, page 4-71

Concurrent Programs Window



Use this window to define and modify your concurrent programs.

Prerequisites

Build the execution file for your concurrent program.

Use the Concurrent Program Executables window to define a concurrent program executable for your operating system program.

Concurrent Programs Block

The combination of application name plus program name uniquely identifies your concurrent program.

Program

You see this longer, more descriptive name when you view your requests in the Requests window. If this concurrent program runs through Standard Request Submission, you see this name in the Submit Requests window when you run this

program.

Short Name

Enter a brief name that Oracle E-Business Suite can use to associate your concurrent program with a concurrent program executable.

Application

The program's application determines what ORACLE username your program runs in and where to place the log and output files.

Enabled

Indicate whether users should be able to submit requests to run this program and the concurrent managers should be able to run your program.

Disabled programs do not show up in users' lists, and do not appear in any concurrent manager queues. You cannot delete a concurrent program because its information helps to provide an audit trail.

(Executable) Executable: Name

Select the concurrent program executable that can run your program. You define the executable using the Concurrent Program Executables window. You can define multiple concurrent programs using the same concurrent program executable. See: Concurrent Program Executables, page 4-67.

(Executable) Executable: Options

Some execution methods, such as Oracle Reports, support additional execution options or parameters. You can enter such options in this field. The syntax varies depending on the execution method.

If you define a concurrent program with the bitmapped version of Oracle Reports, you can control the orientation of the bitmapped report by passing the ORIENTATION parameter or token. For example, to generate a report with landscape orientation, specify the following option in the Options field:

```
ORIENTATION=LANDSCAPE
```

Do not put spaces before or after the execution options values. The parameters should be separated by only a *single* space. You can also specify an orientation of PORTRAIT.

You can control the dimensions of the generated output with the PAGESIZE parameter. A specified <*width*>x<*height*> in the Options field overrides the values specified in the report definition. For example:

```
ORIENTATION=LANDSCAPE PAGESIZE=8x11.5
```

The units for your width and height are determined by your Oracle Reports definition. You set the units in your Oracle Reports menu under Report => Global Properties =>

Unit of Measurement.

If the page size you specify with the PAGESIZE parameter is smaller than what the report was designed for, your report fails with a "REP-1212" error.

(Executable) Executable: Method

The execution method your concurrent program uses appears here.

Valid values are:

Spawned	Your concurrent program is a stand-alone program in C or Pro*C.
Host	Your concurrent program is written in a script for your operating system.
Immediate	Your concurrent program is a subroutine written in C or Pro*C. Immediate programs are linked in with your concurrent manage and must be included in the manager's program library.
Oracle Reports	Your concurrent program is an Oracle Reports script.
PL/SQL Stored Procedure	Your concurrent program is a stored procedure written in PL/SQL.
Java Stored Procedure	Your concurrent program is a Java stored procedure.
Java Concurrent Program	Your concurrent program is a program written in Java.
Multi Language Function	A multi-language support function (MLS function) is a function that supports running concurrent programs in multiple languages (as well as territories and numeric character settings). You should not choose a multi-language function in the Executable: Name field. If you have an MLS function for your program (in addition to an appropriate concurrent program executable), you specify it in the MLS Function field.
SQL*Loader	Your concurrent program is a SQL*Loader program.
SQL*Plus	Your concurrent program is a SQL*Plus or PL/SQL script.
Request Set Stage Function	PL/SQL Stored Function that can be used to calculate the completion statuses of request set stages.

You can switch between Spawned and Immediate, overriding the execution method defined in the Concurrent Program Executable window, only if either method appears

when the executable is selected and both an execution file name and subroutine name have already been specified in the Concurrent Program Executable window. See: Concurrent Program Executables, page 4-67.

(Executable) Priority

You can assign this program its own priority. The concurrent managers process requests for this program at the priority you assign here.

If you do not assign a priority, the user's profile option Concurrent:Priority sets the request's priority at submission time.

(Request) Type

If you want to associate your program with a predefined request type, enter the name of the request type here. The request type can limit which concurrent managers can run your concurrent program.

(Request) Incrementor

For use by Oracle E-Business Suite internal developers only. The incrementor function is shown here.

(Request) MLS Function

The MLS function, if any, used by the program.

The Multilingual Concurrent Request feature allows a user to submit a request once to be run multiple times, each time in a different language. If this program utilizes this feature, the MLS function can be used to determine which installed languages are needed for the request.

Beginning with Release 12.1, MLS functions can support multiple territories and numeric character sets as well as multiple languages.

See:

Oracle E-Business Suite Developer's Guide

> **Note:** If your program has an MLS function associated with it and the "Use in SRS" box (below) is not checked, then the MLS function will be ignored.

(Request) Use in SRS

Check this box to indicate that users can submit a request to run this program from a Standard Request Submission window.

If you check this box, you must register your program parameters, if any, in the Parameters window accessed from the button at the bottom of this window.

(Request) Allow Disabled Values

If you check the Use in SRS box, you can also check this box to allow a user to enter disabled or outdated values as parameter values.

Many value sets use special table columns that indicate whether a particular value is enabled (using ENABLED_FLAG, START_DATE_ACTIVE, and END_DATE_ACTIVE columns). These value sets normally allow you to query disabled or outdated values but not enter them in new data. For Standard Request Submission, this means that a user would not normally be allowed to enter disabled values as report parameter values when submitting a report, even if the report is a query-only type report.

(Request) Run Alone

Indicate whether your program should run alone relative to all other programs in the same logical database. If the execution of your program interferes with the execution of all other programs in the same logical database (in other words, if your program is incompatible with all programs in its logical database, including itself), it should run alone.

You can enter any specific incompatible programs in the Incompatible Programs windows.

(Request) Enable Trace

Turns on SQL tracing when program runs.

(Request) Restart on System Failure

Use this option to indicate that this concurrent program should automatically be restarted when the concurrent manager is restored after a system failure.

(Request) NLS Compliant

This box is checked if the program allows for a user to submit a request of this program that will reflect a language and territory that are different from the language and territory that the users are operating in.

For example, users can enter orders in English in the United Kingdom, using the date and number formats appropriate in the United Kingdom, then generate invoices in German using the date and number formats appropriate to their German customers.

If this box is left blank then a user can associate any installed language with the request, but the territory will default to the territory of the concurrent manager environment.

Note that this option should be set only by the developer of the program. The program must be written as NLS Compliant to utilize this feature. See: the *Oracle E-Business Suite Developer's Guide.*

Note that this option should be set only by the developer of the program. The program must be written as NLS Compliant to utilize this feature.

(Output) Format

Select the output format from the following:

- HTML
- PCL (HP's Printer Control Language)
- PDF
- PS (Post Script)
- Text

> **Important:** If you choose HTML or PDF as the output type with Oracle Report programs, you must use an appropriate printer driver that handles HTML or PDF files.

(Output) Save

Indicate whether to automatically save the output from this program to an operating system file when it is run. This value becomes the default for all requests submitted for this program. The output of programs with Save set to No is deleted after printing.

If this is a Standard Request Submission program, users can override this value from the Submit Requests window.

(Output) Print

If you enter No, your concurrent program's output is never sent to the printer.

(Output) Columns / Rows

Enter the minimum column and row length for this program's report output. Oracle E-Business Suite uses this information to determine which print styles can accommodate your report.

(Output) Style

The print style you select depends on your system and printer setup. Print styles include:

- 132 columns and 66 lines (Landscape)
- 180 columns and 66 lines (Landwide)
- 80 columns and 66 lines (Portrait)

- 132 columns and 62 lines (A4)

Your list is limited to those styles that meet your program's columns and row length requirements.

(Output) Style Required

If your program requires a specific print style (for example, a checkwriting report), use this check box to enforce that print style.

(Output) Printer

If you want to restrict your program's output to a single printer, enter the name of the printer to which you want to send your output. If your program has minimum or maximum columns or rows defined, your list of values is limited to those printers that can support your program's requirements.

Users cannot override your choice of printer from the Submit Requests or Requests windows.

Business Events Region

Concurrent programs can be integrated with the Business Event System in Oracle Workflow. Business events can be raised at key points of the life cycle of a request to run a concurrent program. Users can subscribe to the business events and create their own business processes interacting with the concurrent programs.

Here you specify the points at which business events are enabled. The possible points are:

- Request Submitted
- Request On Hold
- Request Resumed
- Request Running
- Program Completed
- Post Processing Started
- Post Processing Ended
- Request Completed

Possible parameters for each event are:

```
REQUEST_ID
REQUESTED_BY
PROGRAM_APPLICATION_ID
CONCURRENT_PROGRAM_ID
STATUS
COMPLETION_TEXT
TIME_STAMP
```

Concurrent Programs Buttons

Copy to... Choose this button to create another concurrent program using the same executable, request and report information. You can elect to copy the incompatibility and parameter details as well.

Session Control Choose this window to specify options for the database session of the concurrent program when it is executed.

Incompatibilities Choose this button to open the Incompatible Programs window.

Parameters Choose this button to open the Concurrent Program Parameters window.

Copy to Window

Create another concurrent program using the same executable, request and report information as the current program. You can optionally copy the incompatibility and parameter details information as well.

Related Topics

See: Incompatible Programs Window, page 4-80

Concurrent Program Parameters Window, page 4-82

Concurrent Program Executables, page 4-67

Session Control Window

Use this window to specify options for the database session of the concurrent program when it is executed.

Consumer Group

Optionally specify the resource consumer group for the concurrent program.

See: Resource Consumer Groups in Oracle E-Business Suite, *Oracle E-Business Suite Maintenance Guide*.

Rollback Segment

Optionally specify a rollback segment to be used with the concurrent program. This rollback segment will be used instead of the default and will be used up until the first commit.

> **Important:** If you specify a rollback segment here, your concurrent program must use the APIs FND_CONCURRENT.AF_COMMIT and FND_CONCURRENT.AF_ROLLBACK to use the specified rollback segment. See: the *Oracle E-Business Suite Developer's Guide*.

Optimizer Mode

Optionally specify an optimizer mode. You can choose ALL_ROWS, FIRST_ROWS, Rules, or Choose. You would specify an optimizer mode only for a custom program that may not perform well with the default cost-based optimizer (CBO) and needs tuning. You can use a different optimizer mode until your program is tuned for CBO.

Target Node

If you are on a PCP/RAC environment, optionally specify the target node on which requests for this program will run. When requests for this program are submitted, they run on this node if possible.

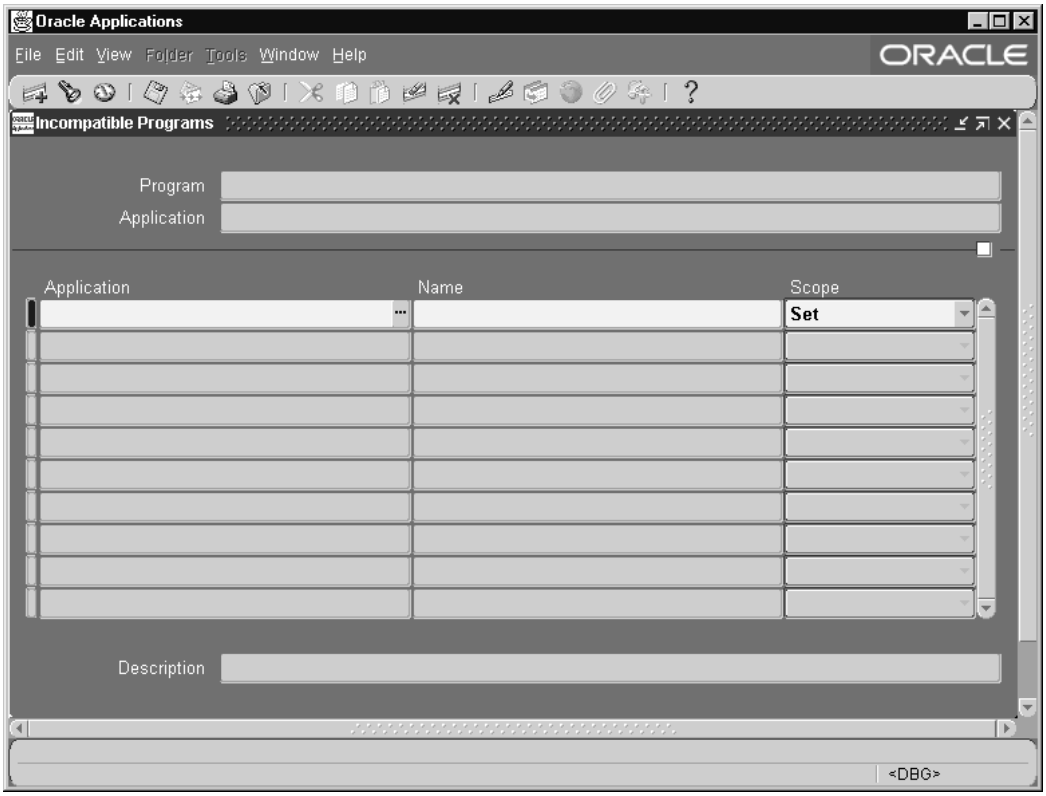
If no specification is made for the concurrent program, a request will be picked up by any manager able to run it.

If the target node is down, any available manager will pick up the request for processing and log a message to FND_LOG_MESSAGES.

Target Instance

Optionally specify a Real Application Cluster (RAC) instance on which the program will run. When requests for this program are submitted, they run on this instance if possible.

Incompatible Programs Window



Identify programs that should not run simultaneously with your concurrent program because they might interfere with its execution. You can specify your program as being incompatible with itself.

Application

Although the default for this field is the application of your concurrent program, you can enter any valid application name.

Name

The program name and application you specify must uniquely identify a concurrent program.

Your list displays the user-friendly name of the program, the short name, and the description of the program.

Scope

Enter Set or Program Only to specify whether your concurrent program is incompatible with this program and all its child requests (Set) or only with this program (Program Only).

Type

Enter Domain or Global. If you choose Domain, the incompatibility is resolved at a domain-specific level. If you choose Global, then this concurrent program will be considered globally incompatible with your concurrent program, regardless of which domain it is running in.

Related Topics

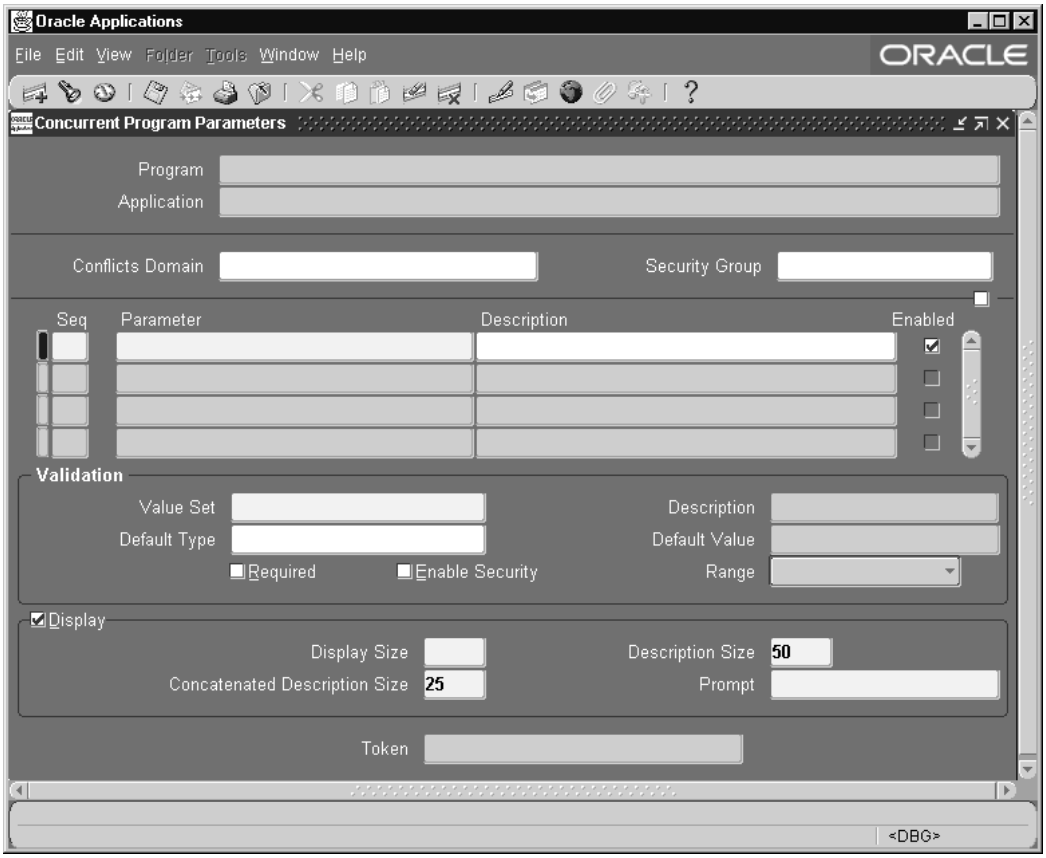
Concurrent Programs Window, page 4-71

Concurrent Program Parameters Window, page 4-82

Defining Program Incompatibility Rules, page 4-34

Modifying an Incompatible Programs List, page 4-53

Concurrent Program Parameters Window



Enter and update the program parameters that you wish to pass to the program executable. Program parameters defined here should match the variables in your execution file.

Conflicts Domain Parameter

Enter the parameter which will hold the value of the conflict domain of the program. For information on conflict domain parameters, see Concurrent Conflict Domains, page 4-35.

Enter the parameter which will hold the value of the conflict domain of the program.

Security Group

This field is for HRMS security only. See: *Customizing, Reporting, and System Administration in Oracle HRMS*.

Sequence

Choose the sequence numbers that specify the order in which your program receives

parameter values from the concurrent manager.

Enabled

Disabled parameters do not display at request submission time and are not passed to your execution file.

Argument Detail

You specify information about your parameter almost exactly as you define a flexfield segment.

(Validation Information) Value Set

Enter the name of the value set you want your parameter to use for validation. You can only select from independent, table, and non-validated value sets.

The maximum size of your value set is 240 characters.

> **Important:** If you are using a value set of dates, this value set should have a format type of either Standard Date or Standard DateTime if you are using the Multilingual Request feature.

(Validation Information) Default Type

If you want to set a default value for this parameter, identify the type of value you need.

Valid types include:

Constant The default value can be any literal value.

Profile The default value is the current value in the user profile option defined in the Default Value field. Use the profile option name, not the end-user name. You do not need to include $PROFILE$.

SQL Statement The default value is determined by the SQL statement you defined in the Default Value field.

Segment The default value is the value entered in a prior segment of the same parameter window.

(Validation Information) Default Value

You can enter a default value for the parameter. This default value for your parameter automatically appears when you enter your parameter window. You determine whether the default value is a constant or a context-dependent value by choosing the default type.

Your default value should be a valid value for your value set. Otherwise you see an error message when you enter your parameter window on the Run Request window and your default value does not appear.

Valid values for each default type include:

Constant Enter any literal value for the default value.

Profile The default value is the current value of the user profile option you specify here. Enter the profile option name, not the end-user name.

Segment The default value is the value entered in a prior segment of the same flexfield window. Enter the name of the segment whose value you want to copy.

SQL Statement The default value is determined by the SQL statement you enter here. Your SQL statement must return exactly one row and one column in all cases.

(Validation Information) Required

If the program executable file requires an argument, you should require it for your concurrent program.

(Validation Information) Enable Security

If the value set for this parameter does not allow security rules, then this field is display only. Otherwise you can elect to apply any security rules defined for this value set to affect your parameter list.

(Validation Information) Range

Choose either Low or High if you want to validate your parameter value against the value of another parameter in this structure. Parameters with a range of Low must appear before parameters with a range of High (the low parameter must have a lower number than the high parameter). For example, if you plan two parameters named "Start Date" and "End Date," you may want to force users to enter an end date later than the start date. You could assign "Start Date" a range of Low and "End Date" a range of High. In this example, the parameter you name "Start Date" must appear before the parameter you name "End Date."

If you choose Low for one parameter, you must also choose High for another parameter in that structure (and vice versa). Otherwise you cannot commit your changes.

(Window Information) Display

Indicate whether to display this parameter in the Parameters window when a user submits a request to run the program from the Submit Requests window.

You should provide a default type and value for any non-displayed parameter.

(Window Information) Display Size

Enter the field length in characters for this parameter. The user sees and fills in the field in the Parameters window of the Submit Requests window.

You should ensure that the total of the value set maximum sizes (not the display sizes) for all of your parameters, plus the number of separators you need (number of parameters minus one), does not add up to more than 240. If your program values' concatenated length exceeds 240, you may experience truncation of your data in some forms.

(Window Information) Description Size

Enter the display length in characters for the parameter value description. Your window may show fewer characters of your description than you specify here if there is not enough room (determined by the sum of your longest prompt plus your display size for this parameter plus seven). However, your window does not display more characters of the description than you specify here.

(Window Information) Prompt

A user sees the prompt instead of the parameter name in the Parameters window of the Submit Requests window.

(Window Information) Concatenated Description Size

Enter the display length in characters for the parameter value description. The user sees the parameter value in the Parameter Description field of the Submit Requests and View Requests forms. The Parameter Description field concatenates all the parameter values for the concurrent program.

> **Tip:** We recommend that you set the Concatenated Description Size for each of your parameters so that the total Concatenated Description Size for your program is 80 or less, since most video screens are 80 characters wide.

(Window Information) Token

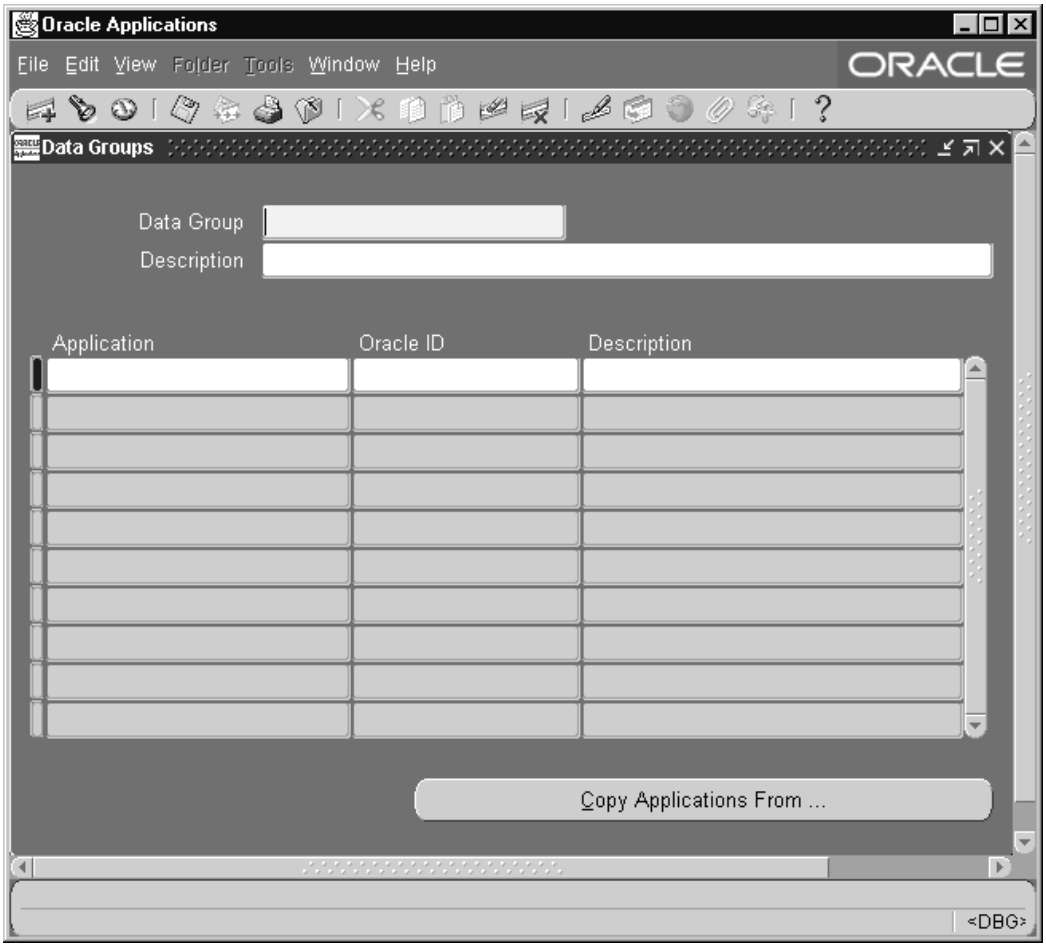
For a parameter in an Oracle Reports program, the keyword or parameter appears here. The value is case insensitive. For other types of programs, you can skip this field.

Related Topics

Concurrent Programs, page 4-71

Incompatible Programs Window, page 4-80

Data Groups Window



Note: Data groups are no longer supported. This section is provided for reference only.

Use this window to define data groups. A data group is a list of Oracle E-Business Suite and the ORACLE usernames assigned to each application.

- If a custom application is developed with Oracle Application Object Library, it may be assigned an ORACLE username, registered with Oracle E-Business Suite, and included in a data group.

An ORACLE username allows access to an application's tables in an ORACLE database. Alldata groups automatically include an entry for Application Object Library.

- A concurrent manager running reports or programs under Oracle E-Business Suite refers to a data group to identify the ORACLE username it uses to access an

application's tables in the database.

- Transaction managers running synchronous programs can only run programs submitted from responsibilities assigned the same data group as the transaction manager. If you create custom data groups, you should create new transaction managers for the applications that use transaction managers. Consult your product documentation to determine if your application uses transaction managers.

Each responsibility within Oracle E-Business Suite is assigned a data group.

During the installation or upgrading of Oracle E-Business Suite, a standard data group is defined, pairing each installed application with an ORACLE username (note: a standard data group is defined for each set of books). You cannot change or delete the predefined values for *Application* or *ORACLE username* in a Standard data group. However, you may:

- Modify the Tool ORACLE username and description associated with an Application-ORACLE username pair.

- Add new Application-ORACLE username pairs to the group.

Data Groups Block

Create a new data group, or modify an existing data group.

You cannot change or delete the predefined values for *Application* or *ORACLE username* in a Standard data group. However, you may modify the Tool ORACLE username and description, or add new Application-ORACLE username pairs to a Standard group.

Data Group

A data group is uniquely identified by its name. You cannot create a data group with a name already in use.

Once saved, data group names cannot be edited.

Application-ORACLE ID Pairs Block

Pair applications with ORACLE usernames.

When you copy a data group, each application, its assigned ORACLE username, and, if present, its Tool ORACLE username and description, appear in this zone automatically. All data groups automatically include an entry for Application Object Library.

Application

Within each data group, an application can be listed only one time.

Oracle ID

Select the ORACLE ID you want to assign to an application. An application uses an ORACLE ID to access tables in the database. Each ORACLE ID allows access to a predefined set of tables in the database.

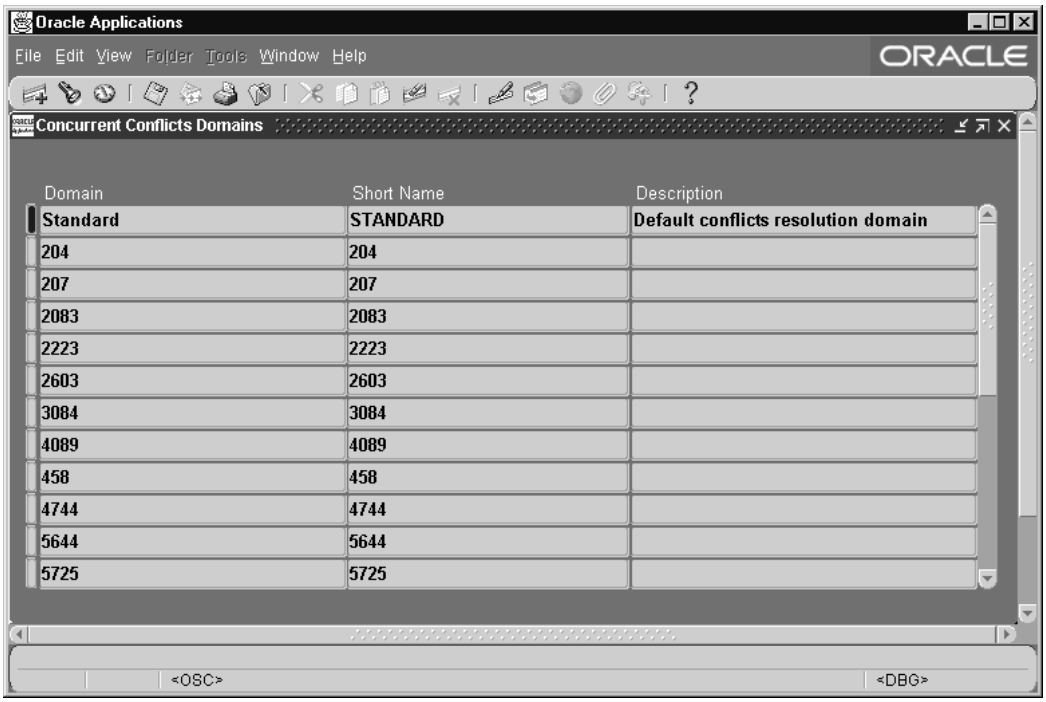
Copy Applications From...

Use this button to copy an existing data group, then add or delete application-ORACLE username pairs to create a new data group.

Related Topics

Overview of Concurrent Programs and Requests, page 4-7

Concurrent Conflicts Domains Window



Concurrent conflicts domains ensure that incompatible concurrent programs are not allowed to run simultaneously using related information.

For example, a conflict domain could be a range of numbers. Two concurrent programs could be incompatible if they used the same range of numbers, but compatible if they used different ranges of numbers.

Concurrent managers use concurrent conflicts domains to determine which concurrent programs cannot run at the same time. For example:

- When concurrent program A is defined as incompatible with concurrent program B, then A and B cannot run at the same time using the same concurrent conflict domain.

- If, for example, the programs A and B are assigned to the concurrent conflicts domains *Standard* when they are submitted, then programs A and B will not run together at the same time.

To define a conflict domain:

1. Enter a unique Domain name. The name you enter here may be used as a value for a parameter in the Submit Requests window.

2. Enter a unique Short Name for your domain. Limit the Short Name to 8 characters.

3. Optionally, you can provide a description for your domain.

Related Topics

Overview of Oracle E-Business Suite DBA Duties, *Oracle E-Business Suite Maintenance Guide*

Concurrent Programs HTML UI

Search for Concurrent Programs

Use this page to search for defined concurrent programs.

From this page you can create a new concurrent program or update an existing one.

Create Concurrent Program

The following are prerequisites to defining a concurrent program:

- Build the execution file for your concurrent program.

- Define a concurrent program executable for your operating system program file.

The combination of application name plus program name uniquely identifies your concurrent program.

Update Annotation

Optionally enter an annotation for the Integration Repository for the concurrent program.

Enabled

Indicate whether users should be able to submit requests to run this program and the concurrent managers should be able to run your program.

Disabled programs do not show up in users' lists, and do not appear in any concurrent manager queues. You cannot delete a concurrent program because its information helps to provide an audit trail.

Program

You see this longer, more descriptive name when you view your requests in the Requests window. If this concurrent program runs through Standard Request Submission, you see this name in the Submit Requests window when you run this program.

Application

The program's application determines what ORACLE username your program runs in and where to place the log and output files.

Short Name

Enter a brief name that Oracle E-Business Suite can use to associate your concurrent program with a concurrent program executable.

Program Type

Available options are:

- Archive - reserved for future use only.
- Autoconfig Type
- Cloning - reserved for internal use only.
- Generic
- Purge - for concurrent programs listed in the Oracle Applications Manager dashboard used for purging data.
- Refresh
- Truncate

Executable

Enter the following:

Name

Select the concurrent program executable that can run your program. You define the executable using the Concurrent Program Executables window. You can define multiple concurrent programs using the same concurrent program executable.

Parameters

Parameters for the concurrent program are listed here. To add a parameter, click on the **Create** button.

Incompatibilities

Identify programs that should not run simultaneously with your concurrent program because they might interfere with its execution. You can specify your program as being incompatible with itself.

Conflict Domain Parameter

Enter the parameter which will hold the value of the conflict domain of the program.

For information on conflict domain parameters, see Concurrent Conflict Domains, page 4-35.

Run Alone

Indicate whether your program should run alone relative to all other programs in the same logical database. If the execution of your program interferes with the execution of all other programs in the same logical database (in other words, if your program is incompatible with all programs in its logical database, including itself), it should run alone.

You can enter any specific incompatible programs in the Incompatible Programs windows.

Application

Although the default for this field is the application of your concurrent program, you can enter any valid application name.

Name

The program name and application you specify must uniquely identify a concurrent program.

Your list displays the user-friendly name of the program, the short name, and the description of the program.

Scope

Enter Set or Program Only to specify whether your concurrent program is incompatible with this program and all its child requests (Set) or only with this program (Program

Only).

Type

Choose the type of incompatibility, either Domain or Global.

For information on incompatibility types, see Incompatible and Run Alone Programs, page 4-34.

Request

Enter the following:

Request Settings

Enter the following:

Type

If you want to associate your program with a predefined request type, enter the name of the request type here. The request type can limit which concurrent managers can run your concurrent program.

Incrementor

For use by Oracle E-Business Suite internal developers only. The incrementor function is shown here.

MLS Function

The MLS (Multi-Lingual Support) function, if any, used by the program.

The Multilingual Concurrent Request feature allows a user to submit a request once to be run multiple times, each time in a different language. If this program utilizes this feature, the MLS function determines which installed languages are needed for the request.

Beginning with Release 12.1, multiple territories and numeric character settings are also supported.

See the *Oracle E-Business Suite Developer's Guide* for more information.

Activity Summarizer

For internal use only.

The Activity Summarizer is a PL/SQL subprogram summarizing about the purgable data in application tables for a concurrent program of type "Purge". It returns a list of table names and rows to be purged. Oracle developers register the PL/SQL procedure as summarizer procedure for the concurrent program.

Refresh Portlet

For internal use only.

Concurrent programs that produce data for a portlet can call a function to refresh the portlet's data. The value for Refresh Portlet indicates when the function should be called.

Allow Multiple Pending Requests

If this box is checked, multiple pending requests are allowed; otherwise, only one pending request is allowed.

Use in SRS

Check the SRS (Standard Request Submission) box to indicate that users can submit a request to run this program from a Standard Request Submission window.

If you check this box, you must register your program parameters, if any, in the Parameters window accessed from the button at the bottom of this window.

Allow Disabled Values

If you check the Use in SRS box, you can also check this box to allow a user to enter disabled or outdated values as parameter values.

Many value sets use special table columns that indicate whether a particular value is enabled (using ENABLED_FLAG, START_DATE_ACTIVE, and END_DATE_ACTIVE columns). These value sets normally allow you to query disabled or outdated values but not enter them in new data. For Standard Request Submission, this means that a user would not normally be allowed to enter disabled values as report parameter values when submitting a report, even if the report is a query-only type report.

Restart on System Failure

Use this option to indicate that this concurrent program should automatically be restarted when the concurrent manager is restored after a system failure.

NLS Compliant

The NLS (National Language Support) box is checked if the program allows for a user to submit a request of this program that will reflect a language and territory that are different from the language and territory that the users are operating in.

For example, users can enter orders in English in the United Kingdom, using the date and number formats appropriate in the United Kingdom, then generate invoices in German using the date and number formats appropriate to their German customers.

If this box is left blank then a user can associate any installed language with the request, but the territory will default to the territory of the concurrent manager environment.

Note that this option should be set only by the developer of the program. The program must be written as NLS Compliant to utilize this feature. See: *Oracle E-Business Suite Developer's Guide.*

Output Preferences

Enter the following:

Save

Indicate whether to automatically save the output from this program to an operating system file when it is run. This value becomes the default for all requests submitted for this program. The output of programs with Save set to No is deleted after printing.

If this is a Standard Request Submission program, users can override this value from the Submit Requests window.

Print

If you enter No, your concurrent program's output is never sent to the printer.

Format

Select the output format from the following:

The format that you select here is used by the concurrent manager to determine how to display your report output. You must ensure that the output format you choose matches the format generated by your report, unless the report is an Oracle Reports report in which case the format you select, determines the output generated.

- HTML
- PCL (HP's Printer Control Language)
- PDF
- PS (Post Script)
- Text

> **Important:** If you choose HTML or PDF as the output type with Oracle Report programs, you must use an appropriate printer driver that handles HTML or PDF files.

Columns / Rows

Enter the minimum column and row length for this program's report output. Oracle E-Business Suite uses this information to determine which print styles can accommodate your report.

Style

The print style you select depends on your system and printer setup. Print styles include:

- 132 columns and 66 lines (Landscape)
- 180 columns and 66 lines (Landwide)
- 80 columns and 66 lines (Portrait)
- 132 columns and 62 lines (A4)

Your list is limited to those styles that meet your program's columns and row length requirements.

Style Required

If your program requires a specific print style (for example, a checkwriting report), use this check box to enforce that print style.

Onsite Setting

The following fields are typically specific to the installation.

General

Enter the following:

Priority

You can assign this program its own priority. The concurrent managers process requests for this program at the priority you assign here.

If you do not assign a priority, the user's profile option Concurrent:Priority sets the request's priority at submission time.

Printer

If you want to restrict your program's output to a single printer, enter the name of the printer to which you want to send your output. If your program has minimum or maximum columns or rows defined, your list of values is limited to those printers that can support your program's requirements.

Users cannot override your choice of printer from the Submit Requests or Requests windows.

Template

The default layout template for the program. Values for this field are available only if the concurrent program has been registered as a data definition with XML Publisher and templates have been loaded to the Template Manager. For more information on XML Publisher and the Template Manager, see the Oracle XML Publisher documentation..

At the time of request submission, the default template is presented to the user. The user can override this value when submitting the request.

Retain for

This field indicates how many days the system should retain data for a request of this concurrent program after the request completes. The system will retain this data for this period even if the "Purge Concurrent Request and/or Manager Data" program is run during this time.

Security Group

This field is for HRMS security only. See: *Customizing, Reporting, and System Administration in Oracle HRMS.*

Log Level for Failure

The log level is used in diagnostics. If a request to run this concurrent program fails, the failure may be recorded in a log file with the specified log level.

Enable Trace

Turns on SQL tracing when program runs.

Enable Time Statistics

Enables the collection of timed statistics, such as CPU and elapsed times, by the SQL

trace facility, as well as the collection of various statistics in the dynamic performance tables.

Delete Log File

By default, a log file is created for each concurrent request. If such log files are not necessary for requests for this concurrent program, you can specify that the log file is automatically deleted for each request of this program.

Target Settings

If you are on a PCP/RAC environment, optionally specify the target node on which requests for this program will run. If no specification is made for the concurrent program, the request will be picked up by any manager able to run it.

If the target node is down, any available manager will pick up the request for processing and log a message to FND_LOG_MESSAGES.

Session Controls

Use this region to specify options for the database session of the concurrent program when it is executed.

Consumer Group

Optionally specify the resource consumer group for the concurrent program. See: Resource Consumer Groups in Oracle E-Business Suite, *Oracle E-Business Suite Maintenance Guide*.

Rollback Segment

Optionally specify a rollback segment to be used with the concurrent program. This rollback segment will be used instead of the default and will be used up until the first commit.

> **Important:** If you specify a rollback segment here, your concurrent program must use the APIs FND_CONCURRENT.AF_COMMIT and FND_CONCURRENT.AF_ROLLBACK to use the specified rollback segment. See: *Oracle E-Business Suite Developer's Guide*.

Optimizer Mode

Optionally specify an optimizer mode. You can choose ALL_ROWS, FIRST_ROWS, Rules, or Choose. You would specify an optimizer mode only for a custom program that may not perform well with the default cost-based optimizer (CBO) and needs tuning. You can use a different optimizer mode until your program is tuned for CBO.

Statistics

This region provides statistics on earlier requests for a defined concurrent program.

Summary information is collected when a request is completed, and stored in the table fnd_conc_prog_onsite_info.

Concurrent Program - Add Parameter

Enter and update the program parameters that you wish to pass to the program executable. Program parameters defined here should match the variables in your execution file.

General

Enter the following:

Enabled

Disabled parameters do not display at request submission time and are not passed to your execution file.

Sequence

Choose the sequence numbers that specify the order in which your program receives parameter values from the concurrent manager.

Parameter

Enter the parameter name. The value is case insensitive.

Validation

Enter the following:

Value Set

Enter the name of the value set you want your parameter to use for validation. You can only select from independent, table, and non-validated value sets.

The maximum size of your value set is 240 characters.

> **Important:** If you are using a value set of dates, this value set should have a format type of either Standard Date or Standard DateTime if you are using the Multilingual Request feature.

Default Type

If you want to set a default value for this parameter, identify the type of value you need.

Valid types include:

Constant The default value can be any literal value.

Profile The default value is the current value in the user profile option defined in the Default Value field. Use the profile option name, not the end-user name. You do not need to

include $PROFILE$.

SQL Statement The default value is determined by the SQL statement you defined in the Default Value field.

Segment The default value is the value entered in a prior segment of the same parameter window.

Default Value

You can enter a default value for the parameter. This default value for your parameter automatically appears when you enter your parameter window. You determine whether the default value is a constant or a context-dependent value by choosing the default type.

Your default value should be a valid value for your value set. Otherwise you see an error message when you enter your parameter window on the Run Request window and your default value does not appear.

Valid values for each default type include:

Constant Enter any literal value for the default value.

Profile The default value is the current value of the user profile option you specify here. Enter the profile option name, not the end-user name.

Segment The default value is the value entered in a prior segment of the same flexfield window. Enter the name of the segment whose value you want to copy.

SQL Statement The default value is determined by the SQL statement you enter here. Your SQL statement must return exactly one row and one column in all cases.

Required

If the program executable file requires an argument, you should require it for your concurrent program.

Enable Security

If the value set for this parameter does not allow security rules, then this field is display only. Otherwise you can elect to apply any security rules defined for this value set to affect your parameter list.

Range

Choose either Low or High if you want to validate your parameter value against the value of another parameter in this structure. Parameters with a range of Low must

appear before parameters with a range of High (the low parameter must have a lower number than the high parameter). For example, if you plan two parameters named "Start Date" and "End Date", you may want to force users to enter an end date later than the start date. You could assign "Start Date" a range of Low and "End Date" a range of High. In this example, the parameter you name "Start Date" must appear before the parameter you name "End Date".

If you choose Low for one parameter, you must also choose High for another parameter in that structure (and vice versa). Otherwise you cannot commit your changes.

If your value set is of the type Pair, this field is display only. The value defaults to Pair.

Display

Enter the following:

Display

Indicate whether to display this parameter in the Parameters window when a user submits a request to run the program from the Submit Requests window.

You should provide a default type and value for any non-displayed parameter.

Token

For a parameter in an Oracle Reports program, the keyword or parameter appears here. The value is case insensitive. For other types of programs, you can skip this field.

Description Size

Enter the display length in characters for the parameter value description. Your window may show fewer characters of your description than you specify here if there is not enough room (determined by the sum of your longest prompt plus your display size for this parameter plus seven). However, your window does not display more characters of the description than you specify here.

Display Size

Enter the field length in characters for this parameter. The user sees and fills in the field in the Parameters window of the Submit Requests window.

You should ensure that the total of the value set maximum sizes (not the display sizes) for all of your parameters, plus the number of separators you need (number of parameters minus one), does not add up to more than 240. If your program values' concatenated length exceeds 240, you may experience truncation of your data in some forms.

The default is the name of the parameter.

Prompt

A user sees the prompt instead of the parameter name in the Parameters window of the Submit Requests window.

Concatenated Description Size

Enter the display length in characters for the parameter value description. The user sees the parameter value in the Parameter Description field of the Submit Requests and View Requests forms. The Parameter Description field concatenates all the parameter values for the concurrent program.

> **Tip:** We recommend that you set the Concatenated Description Size for each of your parameters so that the total Concatenated Description Size for your program is 80 or less, since most video screens are 80 characters wide.

Defining Managers and their Work Shifts in the Oracle Forms UI

This section explains how you can define concurrent managers and specify when a manager is enabled.

A concurrent manager is itself a concurrent program that starts other concurrent programs running. When an application user submits a request to run a program, the request is entered into a database table that lists all of the requests. Concurrent managers read requests from the table and start programs running. See: Concurrent Managers, page 4-160.

In this section, we explain how to specify when a manager is enabled, how to use managers to balance your applications processing workload across different time periods, and how to associate a library of immediate concurrent programs to be called by your manager.

Defining new managers

You can define as many concurrent managers as you want. When you define a manager, you:

- Assign a predefined library of *immediate* concurrent programs to your manager.

  Immediate concurrent programs are subroutines associated with concurrent managers. All other concurrent programs are spawned as independent processes at run time.

- Assign work shifts to your manager, which determines what days and times the manager works.

- For each work shift, you define the maximum number of operating system processes the manager can run concurrently to read requests (start programs) during the work shift.

- Specialize your manager to read only certain kinds of requests.

Program Libraries

For a program that is spawned, a concurrent manager initiates or spawns another operating system process. A program that is immediate runs as part of the concurrent manager's operating system process.

A program library contains immediate concurrent programs that can be called by your manager.

An immediate concurrent program must be registered with a program library. Application developers using Oracle Application Object Librarycan register concurrent programs with a program library.

The Oracle Application Object Library FNDLIBR program library contains Oracle E-Business Suite immediate concurrent programs, and is assigned to the Standard concurrent manager. In most cases, you will include the FNDLIBR library with your manager's definition.

The Internal and the Standard concurrent managers

Oracle E-Business Suite System Administration predefines two managers for you:

- The Internal Concurrent Manager, which functions as the "boss" of all the other managers. The Internal Concurrent Manager starts up, verifies the status of, resets, and shuts down the individual managers.

  You cannot alter the definition of the Internal Concurrent Manager.

- A manager named Standard. The Standard manager accepts any and all requests; it has *no specialization.* The Standard manager is active all the time; it works 365 days a year, 24 hours a day.

  > **Warning:** You should not alter the definition of the Standard concurrent manager. If you do, and you have not defined additional managers to accept your requests, some programs may not run. Use the Standard manager as a safety net, a manager who is always available to run any request. Define additional managers to handle your installation site's specific needs.

Transaction Managers

While conventional concurrent managers let you execute long-running, data-intensive application programs *asynchronously*, transaction managers support *synchronous* processing of particular requests from client machines. A request from a client program to run a server-side program synchronously causes a transaction manager to run it immediately, and then to return a status to the client program.

Transaction managers are implemented as immediate concurrent programs. At runtime, concurrent processing starts a number of these managers. Rather than polling the concurrent requests table to determine what to do, a transaction manager waits to be signalled by a client program. The execution of the requested transaction program takes place on the server, transparent to the client and with minimal time delay. At the end of program execution, the client program is notified of the outcome by a completion message and a set of return values.

Communication with a transaction manager is automatic. The transaction manager mechanism does not establish an ongoing connection between the client and the transaction manager processes. The intent of the mechanism is for a small pool of server processes to service a large number of clients with real-time response.

Each transaction manager can process only the programs contained in its program library. Oracle E-Business Suite developers using *Oracle Application Object Library* can register transaction programs with a program library.

Related Topics

Administer Concurrent Managers, page 4-151

Concurrent Managers, page 4-160

Work Shift Definitions, page 4-102

Using Work Shifts to Balance Processing Workload, page 4-104

Using Time-Based Queues, page 4-106

Work Shift by Manager Report, page 4-115

Work Shifts Report, page 4-115

Work Shifts, page 4-167

Overview of Concurrent Processing, *Oracle E-Business Suite Maintenance Guide*

Work Shift Definitions

When you define a concurrent manager, you assign one or more work shifts to it. Work shifts determine when the manager operates. You define work shifts using the Work Shifts form.

Disabling a work shift

If you define a period of time as a work shift, but do not necessarily want to use the work shift, you can:

- Not assign the work shift to a concurrent manager
- Assign the number of target processes for the work shift as zero (0), on the Define Manager form.

- Delete a work shift assignment using the Define Manager form.

Work Shifts and Hours of the Day

Work shifts can run twenty-four hours a day, from midnight till the next midnight. In military time this is defined as:

- 12:00am - 00:00:00
- 11:59:59pm - 23:59:59

Using work shifts to run through midnight

The military time clock for a twenty-four period starts and stops at midnight.

If you do not want a work shift to run twenty-four hours a day, but you do want to run programs continuously past 12:00 am, you must define two work shifts:

- The first work shift stops at 23:59 (11:59pm).
- The second work shift starts at 00:00 (12:00 am).

For example, you want to run some data-intensive programs during the night, when most of your employees are away from the job site. You define two work shifts which you assign to this manager.

- The first work shift starts at 20:00 (8:00pm) and stops at 23:59 (11:59pm).
- The second work shift starts at 00:00 (12:00am) and stops at 05:00 (5:00am).

Overlapping Work Shifts - Priority Levels

If you assign overlapping work shifts to a concurrent manager, the work shift with the **more specific time period** takes effect for the overlapping time period. For example, a work shift for July 4 overrides a work shift from 9:00 am to 5:00 pm on Monday through Friday.

The following table presents a descending list of priority levels for overlapping work shifts. A work shift with a specific date and range of times has the highest priority. The "Standard" work shift has the lowest priority.

Priority	Work Shift Definition	Example
1	Specific date and range of times	April 15, 2000 8:00am-5:00pm

Priority	Work Shift Definition	Example
2	Specific date and no range of times	April 15, 2000
3	Range of days and range of times	Monday-Friday 8:00am-5:00pm
4	Range of days and no range of times	Monday-Friday
5	Range of times and no date and no range of days	8:00am-5:00pm
6	Standard work shift. No date, days, or time defined.	Standard work shift is 365 days a year, 24 hours a day.

Overlapping Work Shifts with the same priority

When you have overlapping work shifts that have the same level of priority, the work shift with the **largest target processes** takes effect.

For example, you have two work shifts with a range of days and a range of times. You have a "Weekday" work shift from 9:00 am to 5:00 pm on Monday through Friday with 4 target processes.

You also have a "Lunch" work shift from 11:00 am to 1:00 pm on Monday through Friday with 8 target processes.

The "Lunch" work shift takes effect from 11:00 am to 1:00 pm (Mon.-Fri.) because it has the larger number of target processes.

Related Topics

Defining Managers and their Work Shifts, page 4-100

Using Work Shifts to Balance Processing Workload, page 4-104

Using Time-Based Queues, page 4-106

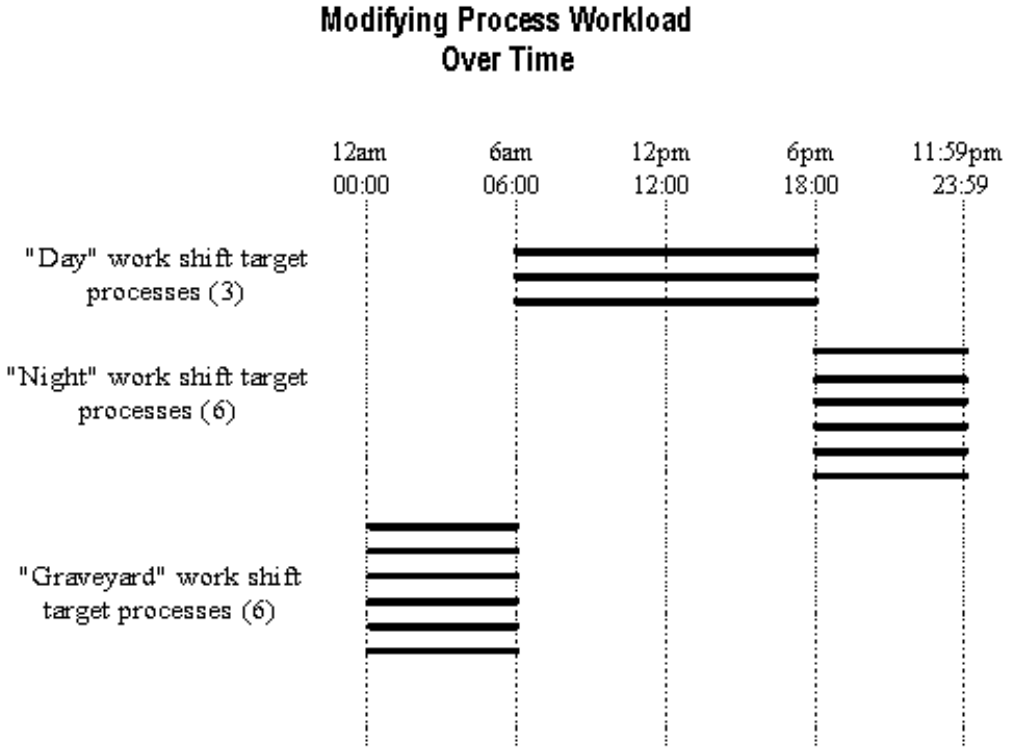
Using Work Shifts to Balance Processing Workload

Part of a manager's definition is how many operating system processes it can devote to reading requests. For each of these processes, referred to as a *target process*, a manager can start one concurrent program.

For each work shift you assign to a manager, you define a number of target processes.

By using work shifts with different numbers of target processes, you can modify your concurrent processing workload according to the day, time of day, and even specific dates.

The figure below illustrates how, by using three work shifts, a manager can be defined to run three programs concurrently from 6:00am-6:00pm, and six programs concurrently from 6:00pm-6:00am.



Failover Sensitivity for Work Shifts

Nodes can become overloaded when a middle-tier node fails and service instances on that node failover to their secondary nodes. The *Load Distribution* feature allows the System Administrator to control the allocation of resources during normal processing. The *Failover Sensitivity* feature allows Work Shifts to failover with fewer target processes than on the original node. This lessens the impact on the existing resources allocated on the secondary node.

The number of failover target processes is entered as part of the standard Work Shift settings in the *Service Instance Definition*. When failover occurs, the ICM uses the **Failover Processes** value in place of the normal running processes value as it iterates through service instances to perform queue sizing.

Related Topics

Defining Managers and their Work Shifts, page 4-100

Work Shift Definitions, page 4-102

Using Time-Based Queues, page 4-106

Work Shift by Manager Report, page 4-115

Work Shifts Report, page 4-115

Work Shifts, page 4-167

Using Time-Based Queues

You can create several time-based queues by defining managers to run programs based on how long those programs have typically run in the past. That is, you can specialize managers to segregate requests according to how long those requests take to run.

To do this, use the Completed Concurrent Requests Report in the System Administrator's report security group. This report lists the actual start date and time and actual completion date and time for concurrent programs that completed running. See: Completed Concurrent Requests Report, page 4-114.

> **Tip:** Run your concurrent programs at different times, perhaps, late at night and then again during the midafternoon, to determine processing time during different workload periods.

For example, based on actual time-to-completion, you can specialize different managers to run the following types of programs:

- inventory pick lists
- payable check runs
- postings
- invoice imports

Augment this approach by defining an "overflow" manager, for example, a manager who can accommodate programs directed to one (or more) of the managers above, but whose work shift is restricted to say, 2:00am-4:00am (02:00-04:00). If some of your long-running programs have not started running before the "overflow" work shift begins, then an additional manager is enabled to accommodate those programs.

Further augment this approach with an "exception" manager defined for *must have* requests. For example, a manager that can run:

- certain programs that must complete by a certain time. The "must-have" manager can be specialized to only read requests for certain programs.
- programs submitted by a particular user, for example, the Company Controller. You can specialize a manager to only read requests from a single application user.

You can even define a second, higher-priority, username for a user to sign on with.

Related Topics

Defining Managers and their Work Shifts, page 4-100

Work Shift Definitions, page 4-102

Using Work Shifts to Balance Processing Workload, page 4-104

Work Shift by Manager Report, page 4-115

Work Shifts Report, page 4-115

Specializing Managers to run only certain programs, page 4-116

Grouping Programs by Request Type, page 4-132

Administer Concurrent Managers, page 4-151

Concurrent Managers, page 4-160

Work Shifts, page 4-167

Creating Services within Oracle Applications Manager

Creating and Editing a Concurrent Manager

Use this page to create a new concurrent manager.

Navigation: Site Map > Request Processing Managers (under Application Services) > (B) Create New or (B) Edit

You can define when a manager runs and how many programs the manager can start simultaneously when you assign work shifts to the manager. Specify which programs a manager can start by defining specialization rules.

General

Enter the following information:

Enabled

Check this box if the manager is enabled.

Manager

The name of the manager.

Short Name

The short name of the manager.

Application

The application name does not prevent a manager from starting programs associated with other applications. To restrict a manager to only running programs associated with certain applications, go to the Rules section.

The combination of an application and the name you define for your manager uniquely identifies the manager.

Cache Size

Enter the number of requests your manager remembers each time it reads which requests to run. For example, if a manager's work shift has 1 target process and a cache value of 3, it will read three requests, and try to run those three requests before reading any new requests.

> **Tip:** Enter a value of 1 when defining a manager that runs long, time-consuming jobs, and a value of 3 or 4 for managers that run small, quick jobs.

Program Library

Select a library of immediate concurrent programs to make available to your manager. Your manager can only run immediate concurrent programs that are registered in the selected program library. Concurrent managers can run only those immediate concurrent programs listed in their program library. They can also run concurrent programs that use any other type of concurrent program executable.

Resources Group

Optionally enter the resource consumer group for this manager.

Rules

Use the Rules section to specialize your manager to run only certain kinds of requests. Without specialization rules, a manager accepts requests to start any concurrent program.

A listing of available rules is displayed. Check the **Include** check box for a rule to include it.

The following information is also given for each rule:

- Type
- Application
- Name
- Description

To edit any of this information, use the **Edit** button. Use the **Remove** button to remove a rule from the list. To create a new rule, use the **Create New** dropdown list and click **Go**.

Work Shifts

Use the Work Shifts section to assign work shifts to your manager. A work shift defines the dates and times the manager is enabled, as well as the number of processes the manager can start running during the work shift.

To add a work shift, use the **Add from Available Shifts** button.

For each work shift listed, the following is displayed:

Sleep Seconds

The sleep time for your manager during this work shift. Sleep time is the number of seconds your manager waits between checking the list of pending concurrent requests (concurrent requests waiting to be started).

> **Tip:** Set the sleep time to be very brief during periods when the number of requests submitted is expected to be high.

Processes

The number of operating system processes you want your work shift to run simultaneously. Each process can run a concurrent request.

For example, if a work shift is defined with three (3) target processes, the manager can run up to three requests simultaneously.

Failover Processes

In the case of node failover, the maximum number of processes that the work shift can run simultaneously.

Nodes can become overloaded when a middle-tier node fails and service instances on that node failover to their secondary nodes. The Failover Processes value should be smaller than the normal Processes value, to lessen the impact on the existing resources allocated on a secondary node. When failover occurs, the ICM uses the Failover Processes value in place of the normal running processes value as it iterates through service instances to perform queue sizing.

Nodes

If you are operating in a parallel concurrent processing environment and you want your manager to operate on a specific node, select the name of the node.

The primary node, if available, is the node your concurrent manager operates on. If the primary node or the database instance on it goes down, your concurrent manager migrates to its secondary node. Your concurrent manager migrates back to its primary node when that node becomes available.

Nodes must be previously registered with Oracle E-Business Suite, using the Nodes window in Oracle E-Business Suite.

Creating and Editing a Transaction Manager

Use this page to create a new Transaction Manager. Transaction Managers handle synchronous requests from client machines.

Navigation: Site Map > Transaction Managers (under Application Services) > (B) Create New or (B) Edit

General

Enter the following information:

Enabled

Check this box if this transaction manager is enabled.

Manager

The name of the transaction manager.

Short Name

The short name for your transaction manager.

Application

The application associated with the transaction manager.

The combination of an application and the short name you specify here uniquely defines the transaction manager.

Program Library

Select a library of immediate transaction programs to make available to your manager. Your manager can only run immediate transaction programs that are registered in the selected program library. Transaction managers can run only those immediate transaction programs listed in their program library. They can also run transaction programs that use any other type of transaction program executable.

Work Shifts

Use the Work Shifts section to assign work shifts to your manager. A work shift defines the dates and times the manager is enabled, as well as the number of processes the manager can start running during the work shift.

To add a work shift, use the **Add from Available Shifts** button.

For each work shift listed, the following is displayed:

Sleep Seconds

The sleep time for a transaction manager determines how often a manager will check to see if it should shut down.

> **Tip:** Set the sleep time to be very brief during periods when the number of requests submitted is expected to be high.

Processes

The number of operating system processes you want your work shift to run simultaneously. Each process can run a concurrent request.

For example, if a work shift is defined with three (3) target processes, the manager can run up to three requests simultaneously.

Nodes

If you are operating in a parallel concurrent processing environment and you want your manager to operate on a specific node, select the name of the node.

The primary node, if available, is the node your concurrent manager operates on. If the primary node or the database instance on it goes down, your concurrent manager migrates to its secondary node. Your concurrent manager migrates back to its primary node when that node becomes available.

Nodes must be previously registered with Oracle E-Business Suite, using the Nodes window in Oracle E-Business Suite.

Creating and Editing an Internal Monitor

Use this page to create a new Internal Monitor.

Internal Monitors monitor the Internal Concurrent Manager in a parallel concurrent processing environment. If the Internal Concurrent Manager exits abnormally (for example, because its node or its database instance goes down), an Internal Monitor restarts it on another node.

General

Enter the following information:

Enabled

Check this box if this internal monitor is enabled.

Manager

The name of the internal monitor.

Short Name

The short name for your internal monitor.

Application

The application associated with the internal monitor.

The combination of an application and the short name you define for your internal monitor uniquely identifies the monitor.

Program Library

For an Internal Monitor, the program library is FNDIMON.

Work Shifts

Use the Work Shifts section to assign work shifts. A work shift defines the dates and times the manager is enabled, as well as the number of processes the manager can start running during the work shift.

To add a work shift, use the **Add from Available Shifts** button.

For each work shift listed, the following is displayed:

Sleep Seconds

The sleep time for your manager during this work shift. Sleep time is the number of seconds your manager waits between checking the list of pending concurrent requests (concurrent requests waiting to be started).

> **Tip:** Set the sleep time to be very brief during periods when the number of requests submitted is expected to be high.

Processes

The number of operating system processes you want your work shift to run simultaneously. Each process can run a concurrent request.

For example, if a work shift is defined with three (3) target processes, the manager can run up to three requests simultaneously.

Nodes

If you are operating in a parallel concurrent processing environment and you want your manager to operate on a specific node, select the name of the node.

The primary node, if available, is the node your concurrent manager operates on. If the primary node or the database instance on it goes down, your concurrent manager migrates to its secondary node. Your concurrent manager migrates back to its primary node when that node becomes available.

Nodes must be previously registered with Oracle E-Business Suite using the Nodes window in Oracle E-Business Suite.

Create a New Work Shift

Use this page to define work shifts for your services. Define work shifts to specify when your services can work.

Navigation: Site Map - Administration > [Service Instance type] (under Application Services) > Create [Service] > Workshifts region, (B) Add from Available Shifts > (B) Create New Site Map > Work Shift Library (under Application Services) > Create New

- Name - The name of your work shift should be intuitive, for instance "Week Days", "Weeknights" or "Weekends".
- Description - Add a description for your work shift.
- Schedule - For each work shift, specify a time period covering a range of days or a particular date. Specify if you are scheduling by day or by date.
- Day - Enter the first and last days of this shift. For instance, if your shift name is "Week Days", you could enter "Monday" in the "Days of Week From" field and "Friday" in the "Days of Week To" field. If you enter a value in the "Days of Week From" field, you must enter a value in the "Days of Week To field".
- Date - Enter a date here to create a date-specific work shift. Date-specific work shifts override work shifts that do not specify a specific date. If you want to enter a value in this field (specify a date), you may not enter values for the Days of Week fields for this row.
- Time - Enter the times of day at which your concurrent shift begins/ends. The time format is HH24:MM. For example, if your work shift name is "Week Days", you could enter "09:00" (9:00 am) as the start time and "17:00" (5:00 pm) as the end time. Note that Oracle E-Business Suite uses a 24-hour clock.

List of Work Shifts

This page displays the available work shifts.

Navigation: Site Map - Administration > Request Processing Managers (under Applications Systems) > Create New or Edit [Service] > Workshifts region, (B) Add from Available Shifts

The following columns are shown: Name, Start Day, End Day, Start Time, End Time, Date, and Description.

You can use the buttons to edit or delete an existing work shift, or create a new one.

Completed Concurrent Requests Report

This report displays how long concurrent programs actually run. Use this report to segregate requests, based on their typical time-to-complete, by specializing concurrent managers to only read requests for certain programs.

Use this report to record parameters and error messages associated with concurrent programs that have been run.

Report Parameters

If you do not enter any parameters, the report returns values for all completed concurrent requests.

Program Application Name

Choose the application name associated with the program whose completed concurrent requests you wish to report on.

Choose only an application name, without a program name, if you wish to run a report on all completed concurrent requests associated with an application.

Program Name

Choose the name of a program whose completed concurrent requests you wish to report on. You must enter a value for Program Application Name before entering a value for Program Name.

User Name

Choose the name of an application user whose completed concurrent requests you wish to report on.

Start Date/End Date

Enter the start date and end date for your report.

Report Headings

The report headings list the specified parameters and provide you with general information about the contents of the report.

Related Topics

Reviewing Requests, Request Log Files, and Report Output Files, *Oracle E-Business Suite Maintenance Guide*

Managing Concurrent Processing Files and Tables, *Oracle E-Business Suite Maintenance*

Guide

Using Time-Based Queues, page 4-106

Specializing Managers to run only certain programs, page 4-116

Grouping Programs by Request Type, page 4-132

Work Shift by Manager Report

This report documents the work shifts assigned to each concurrent manager. Use the report when defining or editing concurrent managers.

Report Parameters

None.

Report Headings

The report headings provide you with general information about the contents of the report.

Related Topics

Defining Managers and their Work Shifts, page 4-100

Work Shifts Report, page 4-115

Work Shifts Report

This report documents all of your work shift definitions. Use this report when defining or editing concurrent manager work shifts.

Report Parameters

None.

Report Headings

The report headings provide you with general information about the contents of the report.

Related Topics

Defining Managers and their Work Shifts, page 4-100

Work Shift Definitions, page 4-102

Work Shifts Report, page 4-115

Work Shift by Manager Report, page 4-115

Work Shifts field help, page 4-167

Specializing Managers to Run Only Certain Programs

This essay explains how you can specialize managers to run only certain programs.

Introduction to Specialization Rules

Every time your users request a concurrent program to be run, their request is inserted into a database table. Concurrent managers read requests from this table, and start running programs if the manager is defined to read the particular request.

Without specialization rules, a manager reads requests to start *any* concurrent program.

Using specialization rules, you can specialize a manager to read only certain kinds of requests to start concurrent programs, for example, only requests to start Oracle General Ledger programs, or only requests to start programs requested by the user "Fred". See: Concurrent Managers, page 4-160.

A special type of specialization rule is the *combined* specialization rule, that can combine more than one action to define a single rule. See: Combined Specialization Rules, page 4-169.

Related Topics

Defining Specialization Rules, page 4-116

Examples - Using Specialization Rules, page 4-121

Defining Combined Specialization Rules, page 4-126

Controlling Concurrent Managers, page 4-134

Concurrent Managers field help, page 4-160

Defining Specialization Rules

A specialization rule associates an action with a type of request. There are two kinds of actions: Include and Exclude.

- Include defines a manager to only read requests of the type specified.
- Exclude defines a manager to read all requests except the type specified.

Requests to run concurrent programs may be allowed or disallowed on the basis of:

- the ORACLE ID of the request's Set of Books (for multiple installs) or Organization if you are using multiple organizations.

- the program itself or the program's application
- the request type of the program
- the user who submitted the request
- a combined rule, which combines more than one action to generate a single rule. The combined rule applies its actions to one or more types of request.

  For example, a combined rule can exclude an action from an Oracle ID and exclude another action from a specific program.

Using more than one rule

Each rule performs one action. When using more than one rule, the rules are evaluated as follows:

- Include rules are evaluated together using 'OR' statements as the binding logic.

  For example, If you use the rules:

  - Include X
  - Include Y

    The result of the rules allows the manager to run either X 'OR' Y but does not require that both programs be run.

- Exclude rules are evaluated together using 'AND' statements as the binding logic.

  For example, If you use the rules:

  - Exclude 1
  - Exclude 2

    The result of the rules prohibits the manager from running programs 1 'AND' 2 together or separately.

- Include rules are evaluated first, then Exclude rules are evaluated. Include rule(s) and Exclude rule(s) are evaluated together as an AND statement. For example, (Include X OR Y) AND (Exclude 1 AND 2).
- An Exclude rule overrides an Include rule.

Specialization rule actions, their binding logic, and examples are presented in the following two tables. See: Specialization Rule Logic - Examples, page 4-117.

Specialization Rule Logic - Examples

Include Rules	Result
Include *X*	Run only program X
Include *X*	Run program X
OR	...or
Include *User Sam*	Run requests by User Sam
	Net result: Run everyone's requests for program X, and run all of Sam's requests.

Exclude Rules	Result
Exclude *37*	Do not run program 37
Exclude *37*	Do not run program 37
AND	...and
Exclude *User Sam*	Do not run requests by User Sam
	Net result: Do not run anyone's requests for program 37, and do not run any of Sam's requests.

Include and Exclude Rules	Result
Include *User Sam*	Run only requests by User Sam
AND	...and
Exclude *37*	Do not run program 37
	Net result: Run all of Sam's requests except requests to run program 37.
Include *X*	(Run program X

Include and Exclude Rules	Result
OR	...or
Include *User Sam*	Run requests by User Sam)
---------- AND	...and
Exclude *37*	(Do not run program 37
AND	...and
Exclude *User Mary*	Do not run requests by User Mary)
	Net result: Run program X except when requested by Mary, and run all of Sam's requests except requests to run program 37.

The following table gives examples of the action types associated with specialization rules.

Rule Action	Type	Example	Explanation
INCLUDE	Combined Rule	Oracle Project Accounting - Tim's Budgets	Manager *only* reads requests to start programs defined by the Combined Rule "Tim's Budgets".
INCLUDE	ORACLE ID	APPS2	Manager *only* reads requests to start programs that connect to the APPS2 (a single install in a multiple install schema) Oracle ID.
INCLUDE	Program	Oracle Project Accounting - Sales Forecast	Manager *only* reads requests to start the concurrent program named "Sales Forecast".

Rule Action	Type	Example	Explanation
INCLUDE	Request Type	Oracle Inventory - Overnight Reports	Manager *only* reads requests to start programs belonging to the request type "Overnight Reports".
INCLUDE	User	Tim	Manager *only* reads requests to start programs submitted by the application user "Tim".
EXCLUDE	Combined Rule	Oracle General Ledger - Month End Reports	Manager reads *all* requests to start programs *except* those defined by the Combined Rule "Month End Reports".
EXCLUDE	ORACLE ID	APPS2	Manager reads *all* requests to start programs *except* those that connect to the APPS2 Oracle ID.
EXCLUDE	Program	Application Object Library - Purge Audit Tables	Manager reads *all* requests to start programs *except* requests for the program named "Purge Audit Tables".
EXCLUDE	Request Type	Oracle Purchasing - Weekend Programs	Manager reads *all* requests to start programs *except* those belonging to the request type "Weekend Programs".

Rule Action	Type	Example	Explanation
EXCLUDE	User	Margaret	Manager reads *all* requests to start programs *except* those submitted by the application user "Margaret".

Related Topics

Specializing Managers to run only certain programs, page 4-116

Examples - Using Specialization Rules, page 4-121

Defining Combined Specialization Rules, page 4-126

Using Combined Rules, page 4-128

Differences Between Specialization and Combined Rules, page 4-131

Concurrent Managers field help, page 4-160

Combined Specialization Rules field help, page 4-169

Examples - Using Specialization Rules

Following are examples of using specialization rules to define what requests a concurrent manager can read. When multiple rules are used to specialize a manager, the words OR and AND appear between each rule to clarify the relationship among multiple specialization rules.

Using Include and Exclude actions

Include Program - Oracle Assets, *No entry for Name field*.

Result The manager only reads requests to run concurrent programs for the application "Oracle Assets".

Include Program - Oracle Assets, *No entry for Name field*.

OR

Include Program - Oracle Payables, *No entry for Name field*.

Net Result The manager only reads requests to run concurrent programs for the application "Oracle Assets", or for the application "Oracle Payables".

The use of multiple Include actions expands the manager's ability to read requests beyond that of a single Program (single Include action).

Exclude Oracle ID - APPS2

Result The manager reads requests to run concurrent programs that connect to any Oracle ID, except those programs that connect to Oracle ID "APPS2".

Exclude Oracle ID - APPS2

AND

Exclude Program - Oracle Payables, *No entry for Name field*.

Net Result The manager reads requests to run concurrent programs that connect to any Oracle ID, except programs that connect to Oracle ID "APPS2", and programs for the application "Oracle Payables".

Simplifying your work

Multiple rules may not always be necessary, or the number or complexity of rules can be simplified. Consider the example below.

Include Program - Oracle Sales and Marketing, *No entry for Name field*.

OR

Include Request Type - Sales Forecasts

Net Result The manager only reads requests to run concurrent programs for the application "Oracle Sales and Marketing", or programs whose request type is "Sales Forecasts".

In this example, both rules are not necessary when programs belonging to the request type "Sales Forecasts" all connect to the Oracle ID "OSM". There is no need for the second Type Include rule.

Exclude rules override Include rules

Include Program - Oracle Payables, *No entry for Name field*.

AND

Exclude Program - Oracle Payables Invoice Aging Report

Net Result The manager reads all requests for concurrent programs for the application "Oracle Payables", but does not read requests to run the Oracle Payables program "Invoice Aging Report".

Include Program - Signon Audit Forms

AND

Exclude Request Type - Signon Audit Reports

Net Result If the System Administrator program Signon Audit Forms belongs to the Request Type "Signon Audit Reports", the manager will not read requests to run the program, even though it has been specifically identified by an Include rule. The Exclude rule overrides the Include rule.

Specializing to only run a Program against specific Oracle IDs

In the following example, a manager can be specialized to only run a program against a specific Oracle ID. This is useful when there are multiple installations of an Oracle Application.

Include Program - Oracle Payables Invoice Aging Report

AND

Exclude Oracle ID - APPS2

Net Result The manager only reads requests to run the Oracle Payables program "Invoice Aging Report" when the program does not connect to the Oracle ID "APPS2". The Exclude action overrides the Include action.

However, when the Invoice Aging Report runs against another Oracle ID, for example, "APPS", then this manager will read requests to run the program.

Distinguishing a Program from a Request Type

You can specialize a manager to read requests to run all the programs belonging to a Request Type, except for individual programs you wish to identify.

Include Request Type - Oracle General Ledger Reports

AND

Exclude Program - Oracle General Ledger Account Analysis

Net Result If the Account Analysis program belongs to the request

type Oracle General Ledger "Reports", then this manager will run every program in the request type Oracle General Ledger Reports, except the program Account Analysis.

Preventing specific programs from running

You can use an Exclude action more than once. For example, suppose your manager reads all requests to run concurrent programs for a particular application, but you want to prevent your manager from running two specific programs. You can:

Include Program - Oracle General Ledger, N*o entry for Name field*.

AND

Exclude Program - Oracle General Ledger Consolidation Audit

AND

Exclude Program - Oracle General Ledger Consolidation Rules

Net Result The manager reads requests for any concurrent programs for the application "Oracle General Ledger", except for the programs "Consolidation Audit" and "Consolidation Rules".

Specializing to run only specific programs at certain times

Using multiple Include rules, you can specialize a manager to run only specific programs. Then, when you define the manager's work shift, you can control when the manager reads requests to run the specific programs.

Include Program - Oracle Payables Invoice Aging Report

OR

Include Program - Oracle Purchasing Receipt Accruals

Net Result The manager only reads requests to run the Oracle Payables Invoice Aging Report, or the Oracle Purchasing Receipt Accruals program.

> **Tip:** If you only wanted these two reports run during the night you can define the manager's work shift to run from 2:00am-6:00am (02:00-06:00).

> **Tip:** When you first submit the requests to

run the programs, you can define a resubmission interval, for example, 1 month, to resubmit the programs to run every month.

Specializing according to Application User

You can specialize managers to only read requests from specific users.

Include User - Markus Kalkin

Net Result The manager only reads requests submitted by the application user "Markus Kalkin".

Include User - Markus Kalkin

OR

Include Program - Oracle Inventory Process Demand Interface

OR

Include Program - Oracle Inventory Summarize Demand Histories

Net Result The manager reads both requests submitted by user Markus Kalkin and requests to run the Oracle Inventory programs "Process Demand Interface" and "Summarize Demand Histories".

Tip: If you want specific programs submitted by a specific user to "jump ahead" of other requests waiting to be run, you can define and specialize a manager as in the example above, and set the user profile option Concurrent:Priority for the user to a high priority (Concurrent:Priority sets the priority of requests submitted by the user).

- Define a manager and give it a descriptive name.
- Specialize the manager as in the example above.
- Set the user profile option

Concurrent:Priority for user Markus to 10.

Related Topics

Specializing Managers to run only certain programs, page 4-116

Defining Specialization Rules, page 4-116

Defining Combined Specialization Rules, page 4-126

Concurrent Managers field help, page 4-160

Defining Combined Specialization Rules

A combined specialization rule combines more than one action to generate a single rule. The actions are combined as AND statements so that the rule is defined as:

Action 1 AND . . .

Action 2 AND . . .

Action 3 AND . . . so on.

You can create combined rules and use them with several managers, instead of duplicating a complex rule each time.

There are two kinds of Actions you may use to build a combined rule; Exclude and Include. Each action is defined by one line within the rule. Combining the specialization lines or individual actions defines the overall combined rule.

An Exclude action overrides a Include action.

For example, you can define an Exclude *application program x* action and a Include *user Yvonne Jones* action. Combining these two actions generates the combined rule "read all requests from user Yvonne Jones except requests to run program x". See: Combined Specialization Rules, page 4-169.

Combined specialization rule actions, their binding logic, and examples are presented in the following table.

Combination Rule Include Lines	Result
Include *Program X*	Run only program X
Include *Program X*	Run program X
AND	...and

Combination Rule Include Lines	Result
Include *User Sam*	Run requests by User Sam Net result: Run only Sam's requests for program X.

Combination Rule Exclude Lines	Result
Exclude *Program 37*	Do not run program 37
Exclude *Program 37*	Do not run program 37
AND	...and
Exclude *User Sam*	Do not run requests by User Sam Net result: Do not run anyone's requests for program 37, and do not run Sam's requests.

Combination Rule Include and Exclude Lines	Result
Include*User Sam*	Run requests by User Sam
AND	...and
Exclude *Program 37*	Do not run program 37 Net result: Run all of Sam's requests except requests to run program 37.
Include *Program Application General Ledger*	(Run General Ledger Programs
AND	...and
Include*User Sam*	Run requests by User Sam)
---------- AND	...and

Combination Rule Include and Exclude Lines	Result
Exclude *Program 37*	(Do not run program 37
AND	...and
Exclude *Program 38*	Do not run program 38)
	Net result: Run Sam's requests for programs from the application General Ledger, except programs 37 and 38.

Related Topics

Specializing Managers to run only certain programs, page 4-116

Defining Specialization Rules, page 4-116

Using Combined Rules, page 4-128

Differences Between Specialization and Combined Rules, page 4-131

Concurrent Managers field help, page 4-160

Combined Specialization Rules field help, page 4-169

Using Combined Rules

Using combined rules you can precisely specialize a manager.

A combined rule combines more than one action to generate a single rule. Each action is defined by one line within the rule. Combining the lines or individual actions defines the overall combined rule.

> **Tip:** You can use a combined specialization rule as one of many rules to specialize a manager.

Using single Exclude and Include actions

A single Exclude action within a combined rule acts the same way as a single Exclude action that defines a specialization rule. Both instruct a manager to read *all* requests to run concurrent programs *except* those identified by the action.

Exclude	Oracle ID - APPS
Result	The manager reads requests to run concurrent programs

that connect to any Oracle ID, except those programs that connect to Oracle ID "APPS".

A single Include action within a combined rule acts the same way as a single Include action that defines a specialization rule. Both actions instruct a manager to read *only* the requests that satisfy the action.

Include Oracle ID - APPS2

Result The manager only reads requests to run concurrent programs that connect to Oracle ID "APPS2".

Using multiple Exclude actions

Using multiple Exclude actions as multiple lines within a combined rule is equivalent to using multiple Exclude actions as multiple specialization rules.

You can exclude more kinds of requests by adding more Exclude lines to your combined rule.

Exclude Program - Oracle Sales & Marketing, *No entry for Name field*.

AND

Exclude Program - Oracle Inventory, *No entry for Name field*.

Net Result The manager reads all requests to run concurrent programs *except requests for* programs for the application "Oracle Sales & Marketing", *and requests for* programs for the application "Oracle Inventory".

Using multiple Include actions

Using multiple Include actions adds more requirements to a combined rule, and excludes more kinds of requests.

You cannot use two Include actions for the same action type. Each Include action is an exclusive statement for a particular type of action. For example, you cannot require a request to be for a program that connects to two different Oracle IDs.

Include Program - Oracle Payables, *No entry for Name field*.

AND

Include Program - Oracle Payables Confirm Receipt Batch

Net Result The manager only reads requests to run a single program, *Confirm Receipt Batch*, and only if that program is from the application "Oracle Payables".

Using Exclude and Include actions

You cannot use Exclude and Include actions for the same type of action. Each Include action is an exclusive statement for a particular type of action.

For example, it does not make sense to *require* a request to be for a program that connects to the Oracle ID "APPS" and disallow a request to connect to another Oracle ID.

Exclude overrides Include

When using multiple lines within a Combined Rule, the Exclude action always overrides a Include action.

Include	Program - Oracle Payables Invoice Import
AND	
Exclude	Oracle ID - APPS2
Net Result	The manager reads requests to run the Oracle Payables Invoice Import program, but will not run the program when it connects to the Oracle ID "APPS2". The Exclude action overrides the Include action.

Specializing a manager to run one program submitted by one user

You can define a combined rule that instructs a manager to only read requests to run a single program when submitted by a specific user.

Include	User - Sheryl
AND	
Include	Program - Oracle Project Accounting Distribute Usage Costs
Net Result	The manager only reads requests submitted by Sheryl to run the Distribute Usage Costs program.

Restricting the programs a manager will run for a specific user

You can define a combined rule that instructs a manager to ignore requests to run a certain programs when submitted by a specific user.

Include	User - Sheryl
AND	
Exclude	Program - Oracle Project Accounting Expenditure Status

Net Result The manager only reads requests submitted by Sheryl, excluding requests to run the Oracle Project Accounting program *Accounting Expenditure Status*.

Specifying Oracle ID and excluding a program from a request type

Include Request Type - Oracle Project Accounting Expenditure Reports

AND

Include Oracle ID - APPS2

AND

Exclude Program - Oracle Project Accounting Expenditure Status

Net Result The manager only reads requests to run programs belonging to the Oracle Project Accounting request type " Reports", run against the Oracle ID "APPS2", excluding the program *Expenditure Reports*.

Related Topics

Specializing Managers to run only certain programs, page 4-116

Defining Combined Specialization Rules, page 4-126

Differences Between Specialization and Combined Rules, page 4-131

Concurrent Managers field help, page 4-160

Combined Specialization Rules field help, page 4-169

Differences Between Specialization and Combined Rules

The primary difference between a specialization rule and a combined specialization rule is in how the use of multiple actions affects the outcome of the rule, as described in the following table:

Rule	Action	Effect of Multiple Actions	Relationship to Other Rules
Specialization Rule	INCLUDE	With each additional Include rule, the manager can read MORE REQUESTS.	Each rule establishes an OR condition. *OR...INCLUDE...*

Rule	Action	Effect of Multiple Actions	Relationship to Other Rules
Specialization Rule	EXCLUDE	With each additional Exclude rule, the manager is excluded from, and reads, FEWER REQUESTS.	Each rule establishes an AND condition. *AND...EXCLUDE...*
Combined Rule Specialization Line	EXCLUDE	With each additional Exclude line, the manager is excluded from, and reads, FEWER REQUESTS.	Each line within a rule establishes an AND condition. *AND...EXCLUDE...*
Combined Rule Specialization Line	INCLUDE	With each additional Include line or additional requirement, the manager reads FEWER REQUESTS.	Each line within a rule establishes an AND condition. *AND...INCLUDE...*

Related Topics

Specializing Managers to run only certain programs, page 4-116

Examples - Using Specialization Rules, page 4-121

Defining Combined Specialization Rules, page 4-126

Using Combined Rules, page 4-128

Concurrent Managers field help, page 4-160

Combined Specialization Rules field help, page 4-169

Grouping Programs by Request Type

As System Administrator, you may want to group similar programs together. You do this by defining request types and assigning them to the programs that users request in Oracle E-Business Suite. You can define concurrent managers that only run programs that belong to a particular request type.

Using request types to specialize concurrent managers can help optimize the processing of Oracle E-Business Suite by letting certain types of programs run without having to wait for other types of programs to finish processing. Using request types saves you time when you create a concurrent manager's specialization rules.

Using Request Types

Specializing a concurrent manage by request type involves three steps:

1. Define a Request Type using the Concurrent Request Types form.
2. Assign the Request Type to each concurrent program you want to identify as a member of this request type using the Concurrent Programs form.
3. Select the Request Type when you specialize a concurrent manager using the Concurrent Managers form.

Examples of using Request Types

Some example request types you may want to define are:

Quick

For concurrent programs that take a relatively short time to run.

Overnight

For concurrent programs that take a long time to run, which you typically schedule to run during the late night or early morning hours.

Month-End Reports

For concurrent programs that generate reports you run at the end of each month.

For example, if you run ten report programs at the end of each month, you could define a request type called " Month-End Reports" and assign it to your ten report programs.

Then you can use specialization rules to define a concurrent manager that only runs requests of type " Month-End Reports". This way, you do not have to specify your ten different report programs when you define your concurrent manager. You can also easily assign the ten programs to more than one manager.

Related Topics

Overview of Concurrent Processing, *Oracle E-Business Suite Maintenance Guide*

Using Time-Based Queues, page 4-106

Completed Concurrent Requests Report, page 4-114

Specializing Managers to run only certain programs, page 4-116

Administer Concurrent Managers, page 4-151

Concurrent Managers, page 4-160

Controlling Concurrent Managers

This essay explains how to control your concurrent managers.

Manager States

Individual managers read requests to start concurrent programs and actually start programs running when certain conditions are satisfied, such as the manager's work shift definition, number of target processes, and specialization rules.

You can start, shut down, or reset the concurrent managers at any time. Oracle E-Business Suite provides an Internal Concurrent Manager that processes these commands. You can issue commands either to individual managers, or, by altering the state of the Internal Concurrent Manager, you can control every manager at once.

> **Note:** Start your concurrent managers on machines with hostnames of 30 or fewer characters. Managers may fail to start on machines with longer hostnames.

Starting Individual Managers

You can restart or activate managers on an individual basis. Restarting a concurrent manager forces the Internal Concurrent Manager to reread the definition for that concurrent manager. Activating a manager cancels a previous command to deactivate it, and allows the Internal Concurrent Manager to submit a request to start that manager when its work shift starts.

You should restart an individual manager when you:

- modify its work shift assignments
- modify a work shift's target number of processes
- modify its specialization rules
- change a concurrent program's incompatibility rules

Deactivating Individual Managers

When you shut down an individual manager, you can choose whether to abort all requests and deactivate the manager immediately, or to allow it to finish processing its current requests before deactivating.

If you choose to Deactivate the manager, requests that are currently running are allowed to complete.

When you terminate requests and deactivate an individual manager, requests that are currently running are immediately stopped and marked for resubmission (when the manager is activated).

Oracle E-Business Suite concurrent programs are designed so that no data is lost or duplicated when a terminated request is resumed after a shut down. This applies for shutdowns that are normal (e.g., using the "Deactivate concurrent manager" request) or abnormal (e.g., after a hardware failure).

> **Important:** When a manager is selected and explicitly deactivated, it remains that way until you select and explicitly activate that manager. As a prerequisite, the Internal manager must be activated beforehand.

Controlling the Internal Concurrent Manager

When you activate the Internal Concurrent Manager, you activate all other managers as well, except those managers that were deactivated on an individual basis.

When you deactivate the Internal Concurrent Manager, it issues commands to deactivate all active managers. Managers that were deactivated on an individual basis are not affected.

If you terminate requests and deactivate the Internal Concurrent Manager, it issues commands to all other managers to terminate their requests and deactivate. Requests that are currently running are immediately stopped and marked for resubmission when the managers are activated.

Verify Concurrent Manager Status

The Internal Concurrent Manager continuously monitors each concurrent manager's operating system process. This process monitoring is referred to as the Internal Concurrent Manager's PMON cycle. The length of the PMON cycle is one of the arguments passed by the STARTMGR command, which starts up the Internal Concurrent Manager.

You can instruct the Internal Concurrent Manager to immediately verify the operating status of your individual concurrent managers, or to perform a PMON check.

Startup Threshold for Concurrent Managers

Concurrent Managers are started from a Service Manager, which in turn is started by the Internal Concurrent Manager. You can set a threshold for the number of requests the Internal Concurrent Manager will make to start a concurrent manager after it fails to start.

During each ICM PMON cycle, the managers are verified and the system attempts to place a lock on each specific manager. If a manager is not up as expected, then the ICM submits a request to start it. However, a manager may have an underlying issue, such as a configuration issue or corrupted executable, that prevents it from starting. By

setting a maximum number of attempts the ICM will make to start a manager over a set time, you can prevent the ICM from continuously making futile attempts to start these managers.

After the underlying problem is fixed, you can restart the manager from the Administer Managers window.

The startup threshold is defined by two profile options:

- CONC: Manager Startup Threshold Limit - This value determines the number of attempts to restart a manager before the system will stop and alert the system administrator. The default value of this profile is 10 (attempts).

- CONC: Manager Startup Threshold Time (minutes) - This value determines the length of time the attempts will be made to restart a manager. If the Threshold Limit as defined above is reached within this time limit, the attempts will stop until the underlying issue has been addressed. The default value of this profile is 60 minutes (1 hour).

If a manager has failed to start after the specified number of attempts (cycles), the manager will not be checked. You can fix the underlying problem, and after it is addressed, you can go to the Administer Managers window, select the manager, and click the **Fixed** button.

The concurrent manager startup threshold can be disabled by setting the profile option CONC: Manager Startup Threshold Limit to 0. This setting will cause the Threshold functionality to be ignored when managers are being checked for restarting.

Controlling Managers from the Administer Managers form

Use the Administer Concurrent Managers form to issue commands to your concurrent managers.

You can also have the Internal Concurrent Manager "manually" verify the status of your individual managers, and restart individual managers. See: Administer Concurrent Managers, page 4-151.

The following table describes control functions for the Internal Manager.

Control Function	Description
Activate concurrent manager	Activates the Internal manager and all other managers, except managers that were deactivated individually using "Deactivate concurrent manager".
Verify concurrent manager status	Manually executes the process monitoring (PMON) cycle.

Control Function	Description
Deactivate concurrent manager	Deactivates the Internal manager and all other managers.
Terminate requests and deactivate manager	All running requests (running concurrent programs) are terminated, and all managers are deactivated.

The following table describes control functions for any other manager.

Control Function	Description
Activate concurrent manager	If the manager is defined to work in the current work shift, it starts immediately. Cancels "Deactivate concurrent manager" and "Terminate requests and deactivate manager".
Restart concurrent manager	Internal manager rereads the manager's definition, and the rules for concurrent program incompatibilities. You should restart a manager when you: - Change work shift assignments - Modify the number of target processes - Modify specialization rules - Change concurrent program incompatibilities
Deactivate concurrent manager	Deactivates the manager. All requests (concurrent programs) currently running are allowed to complete before the manager shuts down. A manager will not restart until you select the manager and choose "Activate concurrent manager".
Terminate requests and deactivate manager	All running requests (running concurrent programs) handled by the manager are terminated. Once deactivated, a manager will not restart until you select the manager and choose "Activate concurrent manager".

Controlling Managers using CONCSUB

You can use CONCSUB to start, stop, and verify managers and other services from the command line. These queue control requests are submitted the same way as other requests submitted through CONCSUB. Before using this utility for queue control, you

should become familiar with the CONCSUB utility and its parameters. See: Submitting Concurrent Requests (CONCSUB), page 4-45 for more information.

The following parameters are used in queue control:

queue control request application name Always 'FND'.

queue control request name Required. The name of your request.

manager application short name Required. The application short name of your concurrent manager.

manager short name Required. The short name of your concurrent manager.

Example:

```
CONCSUB SCOTT/TIGER SYSADMIN 'System Administrator' SYSADMIN CONCURRENT
\
FND ACTIVATE FND STANDARD
```

This command will submit an ACTIVATE request for the Standard manager.

Valid queue control requests are:

- ABORT
- ACTIVATE
- DEACTIVATE
- RESTART
- SHUTDOWN
- STARTUP
- VERIFY
- SUSPEND
- RESUME

Controlling the Internal Concurrent Manager from the Operating System

To start the Internal Concurrent Manager, use the shell script adcmctl.sh.

Alternatively, use one of two other commands you may use from the operating system to control the Internal Concurrent Manager: STARTMGR, which starts the Internal Concurrent Manager; and CONCSUB, which can be used to deactivate or abort the Internal Concurrent Manager, or to instruct the Internal Concurrent Manager to verify the operating system process for each individual manager.

The following table compares the Internal manager control states displayed by the Administer Concurrent Managers form with their corresponding operating system command. Not all arguments are shown.

From the Administer Concurrent Managers Form	From the Operating System (not all arguments shown)
Activate concurrent manager	STARTMGR (syntax may vary with platform)
Verify concurrent manager status	CONCSUB FND VERIFY
Deactivate concurrent manager	CONCSUB FND DEACTIVATE
Terminate requests and deactivate manager	CONCSUB FND ABORT

Starting the Internal Concurrent Manager from the Operating System

To start the Internal Concurrent Manager, use the shell script adcmctl.sh.

This command starts the Internal Concurrent Manager, which in turn starts any concurrent managers you have defined.

Alternatively, to start the concurrent managers, you can invoke the STARTMGR command from your operating system prompt.

You must have write privileges to the "out" and "log" directories of every application so that the concurrent managers can write to these directories. You can start the concurrent managers with many different options. An option on some operating systems is to send an electronic mail note to a given user when the concurrent managers shut down. See your installation guide for a discussion of this command.

Use the STARTMGR command:

- during installation of Oracle E-Business Suite
- after you shut down the concurrent managers
- after MIS restarts the operating system
- after the database administrator restarts the database

The STARTMGR command takes up to ten optional parameters.

- Each parameter except PRINTER has a default.
- You can modify the STARTMGR command and your environment to set your own defaults.

Enter the following command at your system prompt to start the Internal Concurrent Manager:

```
$ startmgr  <optional parameters>
```

You can pass the parameters in any order. For example:

```
$ startmgr sysmgr="<username>/<password>"  mgrname="std"
 printer="hqseq1"  mailto="jsmith"  restart="N"
 logfile="mgrlog"  sleep="90"  pmon="5"  quesiz="10"
```

The startmgr script accepts an Oracle username/password as the sysmgr parameter. Alternatively, you could pass an Oracle E-Business Suite username/password as an appmgr parameter. If no sysmgr or appmgr parameter is provided on the command line, startmgr will prompt you for the Oracle password.

See: Setting Up Concurrent Managers, page 4-176

Viewing the Internal Concurrent Manager startup parameters

The ICM for concurrent requests can be accessed using several navigation paths, including:

- Concurrent Processes window > Internal Manager Log
- Oracle Applications Manager > Site Map > Administration > Request Processing Managers > [Name of Service] > View Processes > ICM Log (B)

The Internal Concurrent Manager's log file displays startup parameter values executed by the STARTMGR command. An example is shown below.

```
logfile=/dbfiles/apps_inst/apps/adsdemo/logs/appl/conc/log/LA0375_0903.m
gr
          PRINTER=noprint
           mailto=applmgr
          restart=N
             diag=N
            sleep=30
             pmon=4
           quesiz=1
```

The Internal Concurrent Manager log file has the default location $APPLCSF/$APPLLOG. However, if the variable $APPLLDM is set to "product", then the location for the log file is $APPLCSF/system/$APPLLOG.

Shutting down the Internal Concurrent Manager from the Operating System

From the operating system prompt, you can use the CONCSUB utility to submit a concurrent request, under the SYSADMIN username and the System Administrator responsibility.

The CONCSUB utility submits a concurrent request and returns you to the operating system prompt. You must wait until the concurrent request completes.

To check on the status of your concurrent request, use the Concurrent Requests form.

```
CONCSUB username/password 'Responsibility application shortname'
 'Responsibility name' 'Username' [WAIT={Y|N|n}] CONCURRENT
 'Program application shortname' PROGRAM
```

Parameters

username/password

The ORACLE username and password that connects to Oracle Application Object Library data. Alternatively, an Oracle E-Business Suite username and password for a user with the System Administrator responsibility.

Responsibility application shortname

The application shortname of the responsibility. For the System Administrator responsibility, the application shortname is SYSADMIN.

Responsibility name

The name of the responsibility. For the System Administrator responsibility, the responsibility name is *System Administrator*.

Username

The application username of the person who submits the request. For example, SYSADMIN is the username of the System Administrator.

WAIT={Y|N|*n*}

Set WAIT to Y if you want CONCSUB to wait until the request you submitted completes before CONCSUB returns you to the operating system prompt.

Set WAIT to N (the default value) if you do not want CONCSUB to wait.

You can also enter an integer value of *n* seconds for CONCSUB to wait before it exits.

When used, WAIT must be entered before CONCURRENT.

Program application shortname

The application shortname of the program. For the DEACTIVATE, ABORT, and VERIFY programs, the application shortname is FND.

PROGRAM

To submit the Shutdown All Managers concurrent request, use the program DEACTIVATE.

To submit the Shutdown Abort Managers concurrent request, use the program ABORT.

To submit the Verify All Managers Status concurrent request, use the program VERIFY.

Example Syntax using CONCSUB

```
CONCSUB <Username/Password> SYSADMIN 'System Administrator'
 SYSADMIN  CONCURRENT FND DEACTIVATE

CONCSUB <Username/Password> SYSADMIN 'System Administrator'
 SYSADMIN  CONCURRENT FND ABORT

CONCSUB <Username/Password> SYSADMIN 'System Administrator'
 SYSADMIN  CONCURRENT FND VERIFY
```

Using CONCSUB to shut down your managers

Use CONCSUB to shut down the concurrent managers:

- before MIS shuts down the operating system
- before the database administrator shuts down the database
- when you want concurrent manager and concurrent program definitions to take effect

Then, use the STARTMGR command to restart the Internal Concurrent Manager, which starts the concurrent managers.

Example - nightly shutdown using CONCSUB

You can use the token WAIT with value Y (WAIT=Y) if you want to use CONCSUB to issue a concurrent request from within a shell script containing a sequence of steps. Using the token WAIT insures the managers deactivate, abort, or verify status before the shell script proceeds to the next step.

See: Controlling the Internal Concurrent Manager from the Operating System, page 4-138

1. The shell script, customized for specific operating system, starts.
2. When the shell script passes control to CONCSUB, CONCSUB waits until the program DEACTIVATE is complete before it returns control to the shell script.

   ```
   CONCSUB username/password SYSADMIN 'System Administrator' \
   SYSADMIN WAIT=Y CONCURRENT FND DEACTIVATE
   ```

3. Script issues the command to shut down the database.
4. Script issues the command to backup the database.
5. Script issues the command to startup the database.

6. `$ startmgr sysmgr="apps/fnd" mgrname="std" printer="hqseq1" mailto="jsmith" restart="N" logfile="mgrlog" sleep="90" pmon="5" quesiz="10"`

   The shell script passes control to STARTMGR, which starts up the Internal manager (and all the other managers).

7. Shell script completes.

Hiding the password using CONCSUB

If the username/password are still supplied, the CONCSUB utility will work as usual.

If username only is supplied (no '/password' in the first argument), it will prompt you for an Oracle E-Business Suite username and password.

In the following example, CONCSUB would connect using the .dbc file, and then only run if the Oracle E-Business Suite user "sysadmin" with password *password* is successfully authenticated.

```
CONCSUB Apps:User SYSADMIN 'System Administrator' SYSADMIN/password
 CONCURRENT FND VERIFY
```

The user can put the password in a file, and then redirect it to standard input (stdin). In UNIX the command would be executed as follows:

```
CONCSUB Apps:User SYSADMIN 'System Administrator' SYSADMIN
 CONCURRENT FND
FNDMNRMT Y 0 20221 < password.file
```

where password.file is an ASCII file that contains the password. This method is recommended for use in shell scripts or batch processes.

Overview of Parallel Concurrent Processing

This essay explains what parallel concurrent processing is, describes the environments it runs in, and explains how it works.

What is Parallel Concurrent Processing?

Parallel concurrent processing allows you to distribute concurrent managers across multiple nodes in a cluster, massively parallel, or networked environment. Instead of operating concurrent processing on a single node while other nodes are idle, you can spread concurrent processing across all available nodes, fully utilizing hardware resources.

Benefits of Parallel Concurrent Processing

Parallel concurrent processing provides Oracle E-Business Suite users with the following benefits:

- High performance - the ability to run concurrent processes on multiple nodes to improve concurrent processing throughput.
- Fault Tolerance - the ability to continue running concurrent processes on available nodes even when one or more nodes fails.
- Adaptability - the ability to integrate with platform-specific batch queue and load-balancing systems to maximize concurrent processing performance on a particular platform.
- Single Point of Control - the ability to administer concurrent managers running on multiple nodes from any node in a cluster, massively parallel, or networked environment.

Parallel Concurrent Processing Environments

Parallel concurrent processing runs in multi-node environments, such as cluster, massively parallel, and networked environments. In these environments, each node consists of one or more processors (CPUs) and their associated memory. Each node has its own memory that is not shared with other nodes And each node operates independently of other nodes, except when sharing a resource such as a disk.

With parallel concurrent processing, one or more concurrent managers run on one or more nodes in a multi-node environment. You decide where concurrent managers run when configuring your system.

You can define any set of concurrent manager specialization rules, and apply them across nodes in any way desired. For example, three "Oracle General Ledger" concurrent managers could be spread across three nodes. Or an "Oracle Payables" concurrent manager and an "Oracle General Ledger" concurrent manager could run simultaneously on the same node.

The following are examples of environments in which parallel concurrent processing can run:

Cluster Environments

In a cluster environment, multiple computers, each representing a single node, share a common pool of disks.

With parallel concurrent processing in a cluster environment, a single ORACLE database resides in the common disk pool, while multiple instances of Oracle Real Application Clusters (Oracle RAC) run simultaneously on multiple nodes in the cluster. Multiple concurrent managers are also distributed across the nodes in the cluster.

Massively Parallel Environments

In a massively parallel environment, multiple nodes are housed in a single computer. All nodes share access to a common pool of disks.

With parallel concurrent processing in a massively parallel environment, separate RAC instances run simultaneously on multiple nodes, with multiple concurrent managers also distributed across nodes.

Networked Environments

In networked environments, multiple computers of the same type are connected via a local area network (LAN) to a single database server, or alternatively, to a cluster of database servers.

With parallel concurrent processing in a networked environment, concurrent managers run on multiple workstations. A single database server can run a single instance of Oracle; or a cluster of database servers can run multiple Oracle instances using Oracle RAC.

How Parallel Concurrent Processing Works

Concurrent Managers

With parallel concurrent processing, each node with concurrent managers may or may not be running an Oracle instance. On a node that is not running Oracle, the concurrent manager(s) connect via Net8 to a node that is running Oracle.

Parallel concurrent processing is activated along with Generic Service Management (GSM); it can not be activated independently of GSM. Beginning with Release 12.2, GSM is always activated. With parallel concurrent processing implemented with GSM, the Internal Concurrent Manager (ICM) tries to assign valid nodes for concurrent managers and other service instances. Primary and secondary nodes need not be explicitly assigned. However, you can assign primary and secondary nodes for directed load and failover capabilities.

> **Note:** In previous releases, you must have assigned a primary and secondary node to each concurrent manager.

Initially, a concurrent manager is started on its defined primary node, or if none exists, the node that the ICM assigns to it. In case of node or Oracle instance failure, all concurrent managers on that node migrate to their respective secondary nodes if defined.

A concurrent manager on its secondary node migrates back to its primary node once that node becomes available. During migration, the processes of a single concurrent manager may be spread across its primary and secondary nodes.

Internal Concurrent Manager

The Internal Concurrent Manager can run on any node, and can activate and deactivate concurrent managers on all nodes. Since the Internal Concurrent Manager must be

active at all times, it needs high fault tolerance. To provide this fault tolerance, parallel concurrent processing uses Internal Monitor Processes.

Internal Monitor Processes

The sole job of an Internal Monitor Process is to monitor the Internal Concurrent Manager and to restart that manager should it fail. The first Internal Monitor Process to detect that the Internal Concurrent Manager has failed restarts that manager on its own node.

Only one Internal Monitor Process can be active on a single node. You decide which nodes have an Internal Monitor Process when you configure your system.

Internal Monitor Processes, like concurrent managers, have assigned work shifts, and are activated and deactivated by the Internal Concurrent Manager.

Concurrent Programs and Requests

You can optionally define a target node for a concurrent program. When a request for the program is submitted, only available managers running on the specified node will pick it up.

If no specification is made for the target node of a concurrent program, a request for it will be picked up by any manager available to run it. If a node specification is made for a concurrent program and the node is up, only available managers running on the specified node will pick up the request. However, if the target node is down, any available manager will pick up the request for processing and log an appropriate message in FND_LOG_MESSAGES.

If no specification is made for the target instance of a concurrent program, a request for it will be picked up by the first manager available to run it and will be run in the instance where the manager is already connected. If an instance specification is made for a concurrent program and the instance is up, it will be picked up by the first manager available to run it and the manager will run the request in the specified instance. However, if the target instance is down, the manager will run the request in the instance where it is already connected and log an appropriate message.

Restarting a Request

The Internal Concurrent Manager (ICM) will only restart a request that a failed manager was running if the following conditions are met:

1. The ICM got the database lock for the manager that was running the request.
2. The phase of the request is Running (phase_code = 'R').
3. The program for this request has "Restart on System Failure" set.
4. In addition to the all of the above three requirements, at least one of the following requirements must be met:

- The ICM is just starting up, (that is, it has just been spawned on a given node and is going through initial code before the main loop).
- The node of the concurrent manager for which the ICM got the lock is down according to a network ping.
- The RAC database instance (TWO_TASK) defined for the node of that concurrent manager is down (this is not applicable if you are using a "balance" TWO_TASK on that node).
- The node of the concurrent manager for which the ICM got the lock is down according to a TNS ping of the APPS listener.

Log and Output File Access

The concurrent log and output files from requests that run on any node are accessible online from any other node. Users need not log onto a node to view the log and output files from requests run on that node.

Integration with Platform-Specific Queuing

Some cluster or massively parallel systems have their own mechanisms for queuing batch processes. Because users may wish to manage all processing with these mechanisms (and not just Oracle E-Business Suite processing), parallel concurrent processing is designed to integrate with them. Thus, you can match your concurrent process management to the specific capabilities of your operating platform.

For more information on integrating with platform-specific queuing, refer to the installation documentation for your platform.

Managing Parallel Concurrent Processing

This section describes how to manage parallel concurrent processing.

Parallel concurrent processing is always active when Generic Service Management (GSM) is active. Parallel concurrent processing can no longer be activated independently of Generic Service Management. Beginning with Release 12.2, GSM is always used in concurrent processing, so parallel concurrent processing is also activated.

However, automatic activation of PCP does not additionally require that primary nodes be assigned for all concurrent managers and other GSM-managed services. If no primary node is assigned for a service instance, the Internal Concurrent Manager (ICM) assigns a valid concurrent processing server node as the target node. In general, this node will be the same node where the Internal Concurrent Manager is running. In the case where the ICM is not on a concurrent processing server node, the ICM chooses an active concurrent processing server node in the system. If no concurrent processing

server node is available, no target node will be assigned.

Note that if a concurrent manager does have an assigned primary node, it will only try to start up on that node; if the primary node is down, it will look for its assigned secondary node, if one exists. If both the primary and secondary nodes are unavailable, the concurrent manager will not start (the ICM will not look for another node on which to start the concurrent manager). This strategy prevents overloading any node in the case of failover.

The concurrent managers are aware of many aspects of the system state when they start up. When an ICM successfully starts up it checks the TNS listeners and database instances on all remote nodes and if an instance is down, the affected managers and services switch to their secondary nodes. Processes managed under GSM will only start on nodes that are in Online mode. If a node is changed from Online to Offline, the processes on that node will be shut down and switch to a secondary node if possible.

Concurrent processing provides database instance-sensitive failover capabilities. When an instance is down, all managers connecting to it switch to a secondary middle-tier node.

However, if you prefer to handle instance failover separately from such middle-tier failover (for example, using TNS connection-time failover mechanism instead), use the profile option Concurrent:PCP Instance Check. When this profile option is set to OFF, Parallel Concurrent Processing will not provide database instance failover support; however, it will continue to provide middle-tier node failover support when a node goes down.

Defining Concurrent Managers

You define concurrent managers either in the Create New Request Processing Manager page in Oracle Applications Manager or in the Concurrent Managers form. When you define a manager, you specify the manager type, which may be either Concurrent Manager, Internal Monitor, or Transaction Manager.

There are three other types of managers that Oracle E-Business Suite predefines for you: the Internal Concurrent Manager, which describes the Internal Concurrent Manager process, the Conflict Resolution Manager, and the Scheduler. For the Conflict Resolution Manager and Scheduler you can assign the primary and secondary nodes. For the Internal Concurrent Manager you assign the primary node only.

You may assign a primary and a secondary node to each concurrent manager. You may also assign primary and secondary system queue names, if a platform-specific queue management system is available on your platform. See: Concurrent Managers, page 4-160.

Administering Concurrent Managers

Target Nodes

Using the Services Instances page in Oracle Applications Manager (OAM) or the Administer Concurrent Managers form, you can view the target node for each concurrent manager in a parallel concurrent processing environment. The target node is the node on which the processes associated with a concurrent manager should run. It can be the node that is explicitly defined as the concurrent manager's primary node in the Concurrent Managers window or the node assigned by the Internal Concurrent Manager.

If you have defined primary and secondary nodes for a manager, then when its primary node and ORACLE instance are available, the target node is set to the primary node. Otherwise, the target node is set to the manager's secondary node (if that node and its ORACLE instance are available). During process migration, processes migrate from their current node to the target node.

Control Across Nodes

Using the Services Instances page in Oracle Applications Manager or the Administer Concurrent Managers form, you can start, stop, abort, restart, and monitor concurrent managers and Internal Monitor Processes running on multiple nodes from any node in your parallel concurrent processing environment. You do not need to log onto a node to control concurrent processing on it. You can also terminate the Internal Concurrent Manager or any other concurrent manager from any node in your parallel concurrent processing environment.

In an environment enabled with parallel concurrent processing, primary node assignment is optional for the Internal Concurrent Manager. The Internal Concurrent Manager can be started from any of the nodes (host machines) identified as concurrent processing server enabled. In the absence of a primary node assignment for the Internal Concurrent Manager, the Internal Concurrent Manager will stay on the node (host machine) where it was started. If a primary node is assigned, the Internal Concurrent Manager will migrate to that node if it was started on a different node.

If the node on which the Internal Concurrent Manager is currently running becomes unavailable or the database instance to which it is connected to becomes unavailable, the Internal Concurrent Manager will be restarted on a alternate concurrent processing node. If no primary node is assigned, the Internal Concurrent Manager will continue to operate on the node on which it was restarted. If a primary node has been assigned to the Internal Concurrent Manager, then it will be migrated back to that node whenever both the node and the instance to which the Internal Concurrent Manager connects to from that node becomes available

Starting Up Managers

You start up parallel concurrent processing as you would ordinary concurrent

processing, by running the adcmctl.sh script from the operating system prompt.

The Internal Concurrent Manager starts up on the node on which you run the adcmctl.sh script. If it has a different assigned node, it will migrate to that node if available.

After the Internal Concurrent Manager starts up, it starts all the Internal Monitor Processes and all the concurrent managers. It attempts to start Internal Monitor Processes and concurrent managers on their primary nodes, and resorts to a secondary node only if a primary node is unavailable.

Shutting Down Managers

You shut down parallel concurrent processing by issuing a "Stop" command in the OAM Service Instances page or a "Deactivate" command in the Administer Concurrent Managers form. All concurrent managers and Internal Monitor processes are shut down before the Internal Concurrent Manager shuts down.

Terminating Concurrent Processes

You can terminate running concurrent processes for a concurrent manager on the local node or on remote nodes by issuing an "Abort" command from the OAM Service Instances page or a "Terminate" command from the Administer Concurrent Managers form.

Migrating Managers

Most process migration occurs automatically in response to the failure or subsequent availability of a primary node. However, you may migrate processes manually by changing the node assignments for a concurrent manager or Internal Monitor Process using the Concurrent Managers form. To put your changes into effect, issue a "Verify" command against the Internal Concurrent Manager from the Administer Concurrent Managers form.

Related Topics

Concurrent Managers, page 4-160

Administer Concurrent Managers Window

View the status of your concurrent managers (including any transaction managers) and, if you wish, change the status of any manager by issuing a control command. For example, you can deactivate a manager that is currently active, then view its new status after the change takes effect.

Use the Refresh button to refresh the data shown.

Administer Concurrent Managers Block

Node

In a parallel concurrent processing environment, a manager's processes are targeted to run on this node.

If a concurrent manager is defined to use a platform-specific system queue, this field displays the name of the queue to which the manager submits its processes.

Processes Actual

Each manager process can run one concurrent request (start one concurrent program). Typically, the number of actual processes equals the number of target processes (the maximum number of requests a manager can run).

However, the number of actual processes may be less than the number of target processes due to manager deactivation or manager migration.

Processes Target

This field displays the maximum number of manager processes that can be active for this manager.

Requests Running/Requests Pending

Typically, when there are requests pending, this number should be the same as the number of actual processes. However, if there are no pending requests, or requests were just submitted, the number of requests running may be less thanthe number of actual processes.

Moreover, if a concurrent program is incompatible with another program currently running, it does not start until the incompatible program has completed. In this case, the number of requests running may be less than number of actual processes even when there are requests pending.

Controlling a Specific Manager - Status

This field displays the status of a manager after you have chosen a specific action for it using the top row of buttons near the bottom of the window.

You can control concurrent managers individually or collectively by controlling the Internal Concurrent Manager. This field is blank when managers have been activated by the Internal Concurrent Manager.

In a parallel processing environment, this field displays *Target node/queue unavailable* when the primary and secondary nodes (or system queues) are not available.

The actions you can choose for controlling a manager are:

Terminate

When you terminate requests and deactivate the Internal Concurrent Manager, all running requests (running concurrent programs) are terminated, and all managers are deactivated.

Managers previously deactivated on an individual basis are not affected.

You can terminate requests and deactivate individual managers. All running requests (running concurrent programs) handled by the manager are terminated.

Once deactivated, a manager does not restart until you select the manager and choose the Activate button.

Deactivate

When you deactivate the Internal Concurrent Manager, all

other managers are deactivated as well. Managers previously deactivated on an individual basis are not affected.

You can deactivate individual managers. Once deactivated, a manager does not restart until you select the manager and choose the Activate button.

When you deactivate a manager, including the Internal Concurrent Manager, all requests (concurrent programs) currently running are allowed to complete before the manager(s) shut down.

Verify

This choice appears only when you select the Internal Concurrent Manager.

The Internal Concurrent Manager periodically monitors the processes of each concurrent manager. You can force this process monitoring or PMON activity to occur by choosing the Verify button.

Another result of selecting this choice is that the Internal Concurrent Manager rereads concurrent program incompatibility rules.

Restart

This choice appears only when you select an individual manager.

When you restart a concurrent manager, the manager rereads its definition.

You should restart a manager when you have made the following changes using the Define Concurrent Manager form, and you wish those changes to take effect:

- Change work shift assignments
- Modify the number of Target Processes
- In a parallel concurrent processing environment, change node or system queue information

Fixed

This choice appears only when a manager has been down due to an underlying issue beyond the concurrent manager startup threshold and the Internal Concurrent Manager has stopped attempting to restart it. After the underlying problem has been fixed, this choice is available so that you can indicate that it has been fixed and the ICM should try to restart the concurrent manager again.

Activate

When you activate the Internal Concurrent Manager, you activate all other managers as well, except those managers that were deactivated on an individual basis.

You cannot activate the Internal Concurrent Manager from the PC client. The Internal Concurrent Manager is only activated from the server.

You can also activate an individual concurrent manager that is currently deactivated, so long as the Internal manager is active. If the manager is defined to work in the current work shift, then the Internal manager starts it immediately.

Reviewing a Specific Manager

View details of a concurrent manager's operation

Processes

You can view the details of the processes of a given concurrent manager. Processes that are currently active, migrating, or terminating, as well as processes that have been terminated or deactivated, are displayed.

Requests

For a selected manager you can view all running and pending requests handled by the manager.

The following actions are available only for certain services managed Generic Service Management. These services must be defined to accept commands to suspend their operations.

Suspend

Suspend the operations of the service.

Resume

Resume the operations of the service.

Related Topics

Concurrent Processing Window, page 4-155

Defining Managers and their Work Shifts, page 4-100

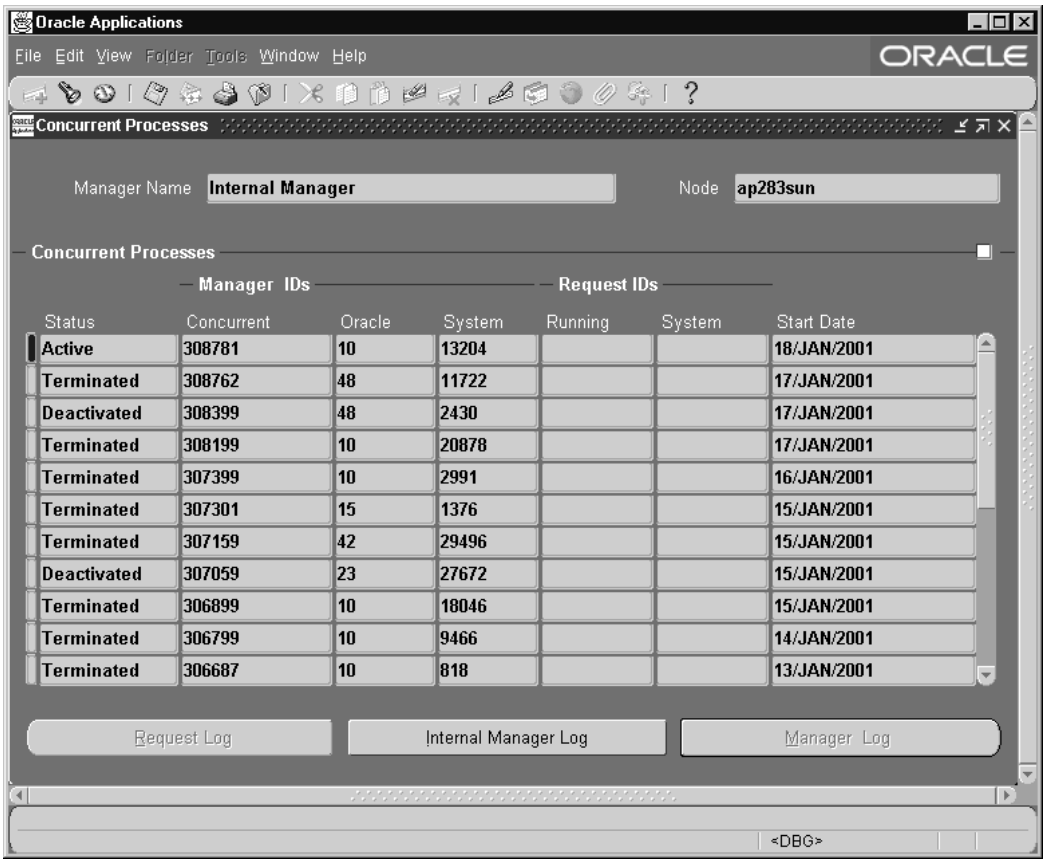
e

Controlling Concurrent Managers, page 4-134

Overview of Parallel Concurrent Processing, page 4-143

Life cycle of a concurrent request, *Oracle E-Business Suite Maintenance Guide*

Concurrent Processes Window



View status information about the processes of a specific concurrent manager, whose name and node are identified near the top of the window.

Displaying this window automatically queries all processes that are currently active, migrating, or terminating, as well as processes that have been terminated or deactivated.

Display order is by status value (Active, Migrating, Terminating, Terminated, Deactivated) and within status, by the order in which processes were started.

If you wish to reduce the number of displayed processes, you can delete records by submitting the "Purge Concurrent Request and Managers" report from the Run Requests form. You can delete records according to the number of days since the processes were started. However, you cannot delete the records of currently active managers.

Status

This field cannot be updated. The following are valid status values:

Active Currently running manager processes display as "Active".

Deactivated Manager processes that are no longer running display as "Deactivated".

These processes were deactivated by you choosing the Deactivate button in the Administer Concurrent Managers block, or by the Internal Concurrent Manager deactivating a concurrent manager at the end of that manager's work shift.

Migrating Managers that are migrating between primary and secondary nodes display as "Migrating".

In a parallel concurrent processing environment, concurrent managers run on either the primary or secondary node assigned to them. Managers migrate to the secondary node if the primary node or the database instance on the primary node is unavailable. Managers migrate back to the primary node once it becomes available.

Terminating Manager processes that are being terminated display as "Terminating".

These processes were terminated by you choosing the Terminate button in the Administer Concurrent Managers block, or by a user selecting "Terminate" in the Concurrent Requests form.

Terminated Manager processes that have been terminated display as "Terminated".

These processes were terminated by you choosing the Terminate button in the Administer Concurrent Managers block, or by a user selecting "Terminate" in the Concurrent Requests form.

Manager Identifiers Concurrent

This field displays a number generated by the individual concurrent manager that identifies the process. This field cannot be updated.

This number may be referenced if an operating system process ID is not available.

You can use this number to view the log file associated with the process. (This is the same log file you view when you select Manager Log from the View field of the Concurrent Requests form).

- At the operating system level, go to the pertinent directory. The log file is located in

$APPLCSF/$APPLLOG by default. However, if APPLLDM is set to "product" then the the log file is under $APPLCSF/system/$APPLLOG.

- For concurrent managers, use w<*number*>.mgr.
- For Internal Monitor processes, use i<*number*>.mgr.

Manager Identifiers Oracle

This field displays the ORACLE process ID associated with the manager process. This field cannot be updated.

Manager Identifiers System

This field displays the operating system process ID associated with the manager process. This field cannot be updated.

Request Identifiers Running

Please note the following about this field:

- Normally this field is blank, as the runtime of a request is typically very short.
- For a terminated manager, the ID of the request being processed at the time of termination is displayed.

Request Identifiers System

This field displays the operating system process ID for a spawned concurrent process.

Viewing Log Files

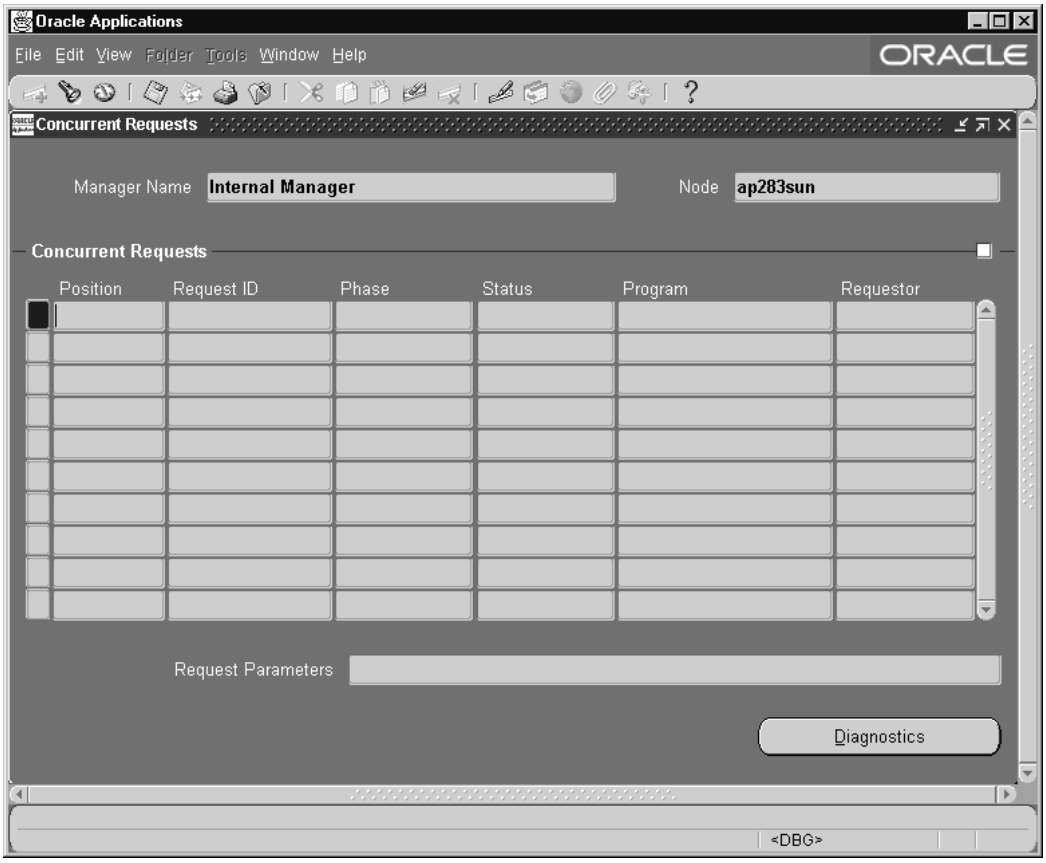
Use the three buttons near the bottom of the window to view log files. Log files record information that may be helpful when diagnosing problems.

Request Log Choose this button to view the log file of the process associated with the running request.

Internal Manager Log Choose this button to view the Internal Concurrent Manager's log file.

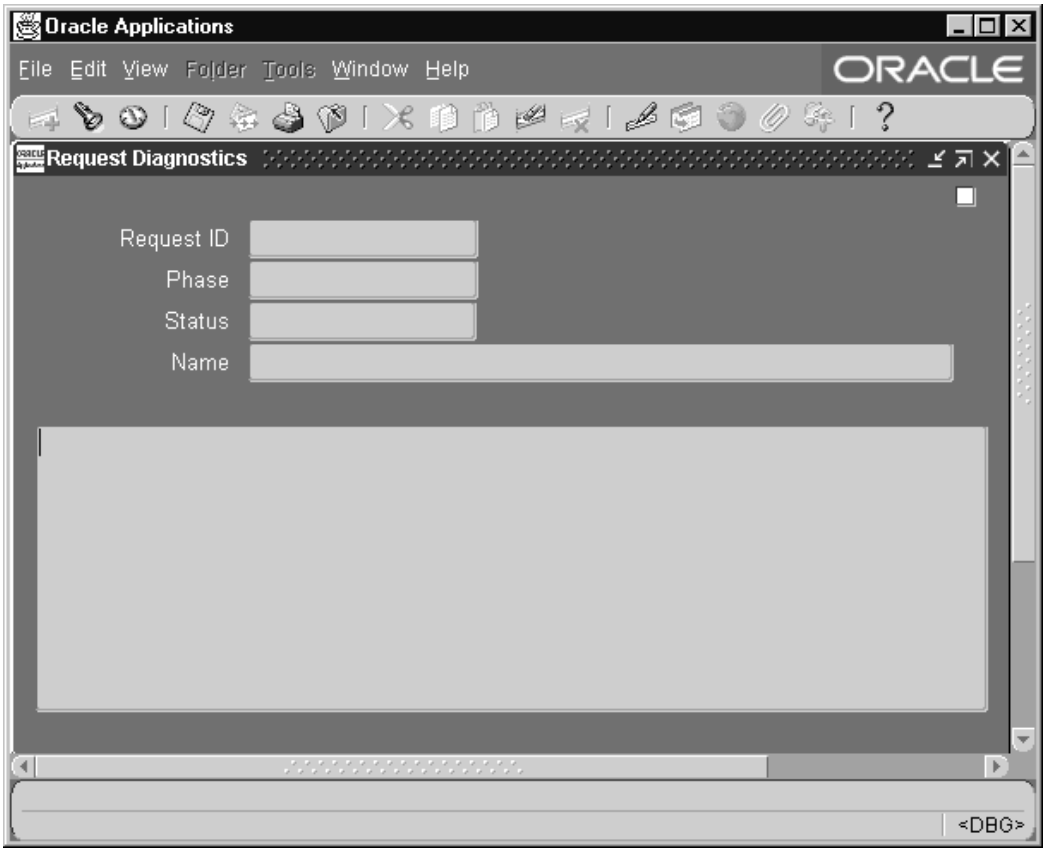
Manager Log Choose this button to view the log file of the concurrent manager who started running the request.

Concurrent Requests Window



View all running and pending requests for a selected manager, whose name and node are identified near the top of the window.

Request Diagnostics Window



This window informs you when the request completed or if it did not complete, shows you a diagnostic message indicating why.

Related Topics

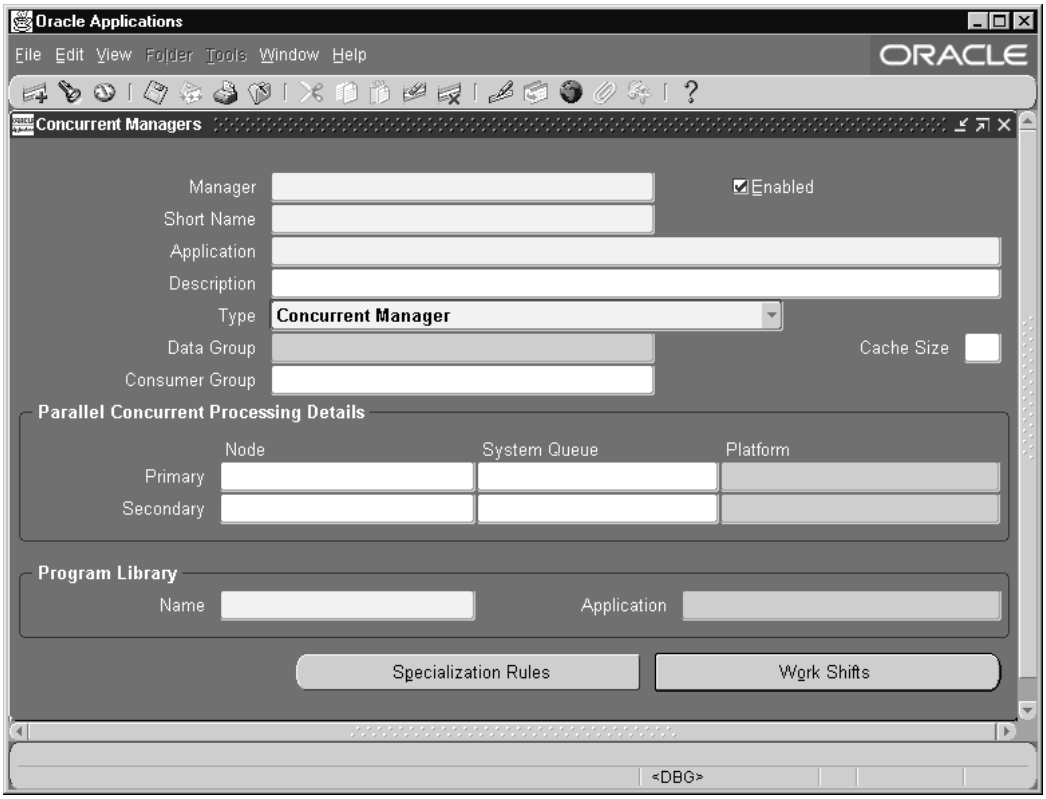
Concurrent Manager field help, page 4-160

Defining Managers and their Work Shifts, page 4-100

Controlling Concurrent Managers, page 4-134

Overview of Parallel Concurrent Processing, page 4-143

Concurrent Managers Window



Use this window to define your concurrent managers. You can determine when a manager runs and how many programs a manager can start simultaneously when you assign workshifts to the manager. Determine which programs a manager can start by defining specialization rules.

Concurrent Managers Block

The combination of an application and the name you define for your manager uniquely identifies the manager.

Application

The application name does not prevent a manager from starting programs associated with other applications. To restrict a manager to only running programs associated with certain applications, go to the Specialization Rules window.

Type

Once you define a concurrent manager, you cannot update this field. There are several types of managers:

Concurrent Manager	Concurrent Managers start concurrent programs running.
Internal Monitor	Internal Monitors monitor the Internal concurrent manager in a parallel concurrent processing environment. If the Internal Concurrent Manager exits abnormally (for example, because its node or its database instance goes down), an Internal Monitor restarts it on another node.
Transaction Manager	Transaction managers handle synchronous requests from client machines.

Cache Size (Concurrent Manager only)

Enter the number of requests your manager remembers each time it reads which requests to run. For example, if a manager's workshift has 1 target process and a cache value of 3, it will read three requests and wait until these three requests have been run before reading new requests.

In reading requests, the manager will only put requests it is allowed to run into its cache. For example, if you have defined your manager to run only Order Entry reports then the manager will put only Order Entry requests into its cache.

If you enter 1, the concurrent manager must look at its requests list each time it is ready to process another request.

By setting the cache size at a higher number, the concurrent manager does not have to read its requests list each time it runs a request. However, the manager does not recognize any priority changes you make for a particular request if it has already read that request into its cache. Further, even if you give a higher priority to a new request, that new request must wait until the buffer is empty and the manager returns to look at the requests list. That request may have to wait a long time if you set the buffer size to a high number.

You should use cache size to tune your concurrent managers to work most efficiently for you site's needs. If your organization tends to reprioritize jobs going to a certain manager, that manager should have its buffer size set fairly low.

> **Tip:** Enter a value of 1 when defining a manager that runs long, time-consuming jobs, and a value of 3 or 4 for managers that run small, quick jobs.

Data Group (Transaction Manager only)

The data group the transaction manager uses to connect to the database. Transaction managers only run programs submitted from responsibilities that use the same data group as the transaction manager.

Note: Data groups are no longer supported by Oracle E-Business Suite. This section is provided for reference only.

Resource Consumer Group

The resource consumer group for the manager. For more information on resource consumer groups, see: Resource Consumer Groups in Oracle E-Business Suite, *Oracle E-Business Suite Maintenance Guide*.

(Parallel Concurrent Processing Details) Node

If you are operating in a parallel concurrent processing environment and you want your manager to operate on a specific node, select the name of the node.

The primary node, if available, is the node your concurrent manager operates on. If the primary node or the database instance on it goes down, your concurrent manager migrates to its secondary node. Your concurrent manager migrates back to its primary node when that node becomes available.

Nodes must be previously registered with Oracle E-Business Suite, using the Nodes window. See: Nodes, page 4-175.

(Parallel Concurrent Processing Details) System Queue

If you are operating in a parallel concurrent processing environment and you want your manager to use a platform-specific queue management system instead of generic concurrent processing queue management, specify the queue or class name of that system. For example, you may choose a system queue name from a platform-specific queue management system like NQS or IBM Load Leveler.

The primary system queue is the queue you associate with the primary node. The secondary system queue is the queue you associate with the secondary node.

Important: To ensure that your manager uses your platform-specific queue management system, you should start the concurrent managers in the proper mode. Refer to platform-specific documentation to determine if your platform supports interfacing with system queues. For UNIX platforms, refer to the appropriate Oracle E-Business Suite installation Update. For all other platforms, refer to the appropriate Oracle E-Business Suite installation guide.

Program Library

Concurrent managers can run only those immediate concurrent programs listed in their program library. They can also run concurrent programs that use any other type of concurrent program executable as long as the specialization rules include them.

Immediate concurrent programs must be registered in a program library by an applications developer using Oracle Application Object Library.

Transaction Managers can only run programs listed in their program library.

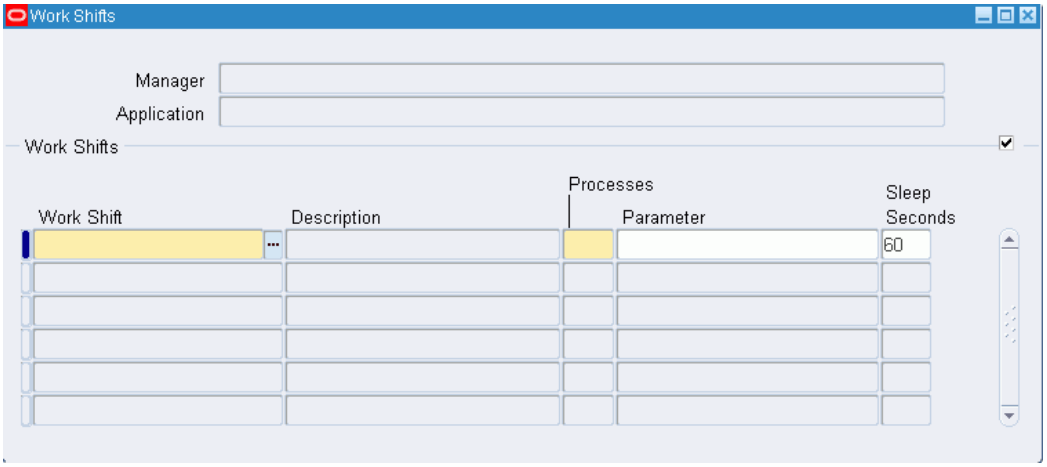
Defining Manager Operations

The two buttons near the bottom of the window display additional windows for defining when your manager operates, and, if you wish, specializing your manager to run only certain kinds of programs.

Specialization Rules You define what kinds of requests the manager reads by defining specialization rules for your manager.

Work Shifts You define when the manager operates by assigning one or more work shifts to your manager. With each work shift, you can vary the number of programs the manager can run simultaneously.

Work Shifts Window



Assign work shifts to a concurrent manager. A work shift defines the dates and times the manager is enabled. For each work shift you define the number of processes the manager starts running.

Work shifts are defined using the Work Shifts form. See: Work Shifts, page 4-167.

Processes

Enter the number of operating system processes you want your work shift to run simultaneously. Each process can run a concurrent request.

For example, if a work shift is defined with three (3) target processes, the manager can run up to three requests simultaneously.

Parameter

Enter the parameter string for a service under Generic Service Management. The value of this field is dependent on the service type definition.

These parameters are used only for managers that are services, such as the OAM Generic Collection Service, the OPP, the Workflow services, and so on. The parameters are specific to each service. For example, the OPP service uses these parameters:

```
oracle.apps.fnd.cp.opp.OPPServiceThread:2:0:max_threads=5
```

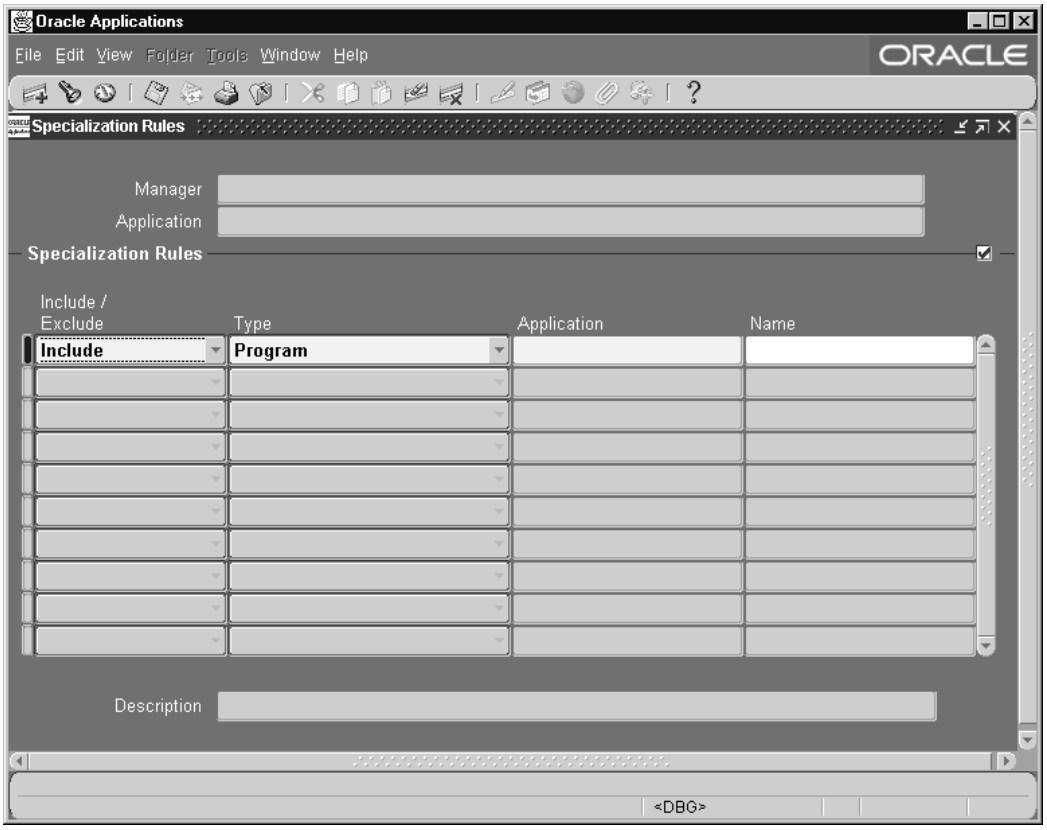
This tells the service to use the class OPPServiceThread, use 2 service threads, and use a maximum of 5 request threads.

Sleep Seconds

Enter the sleep time for your manager during this work shift. Sleep time is the number of seconds your manager waits between checking the list of pending concurrent requests (concurrent requests waiting to be started).

> **Tip:** Set the sleep time to be very brief during periods when the number of requests submitted is expected to be high.

Specialization Rules Window



Specialize your manager to run only certain kinds of requests. Without specialization rules, a manager accepts requests to start *any* concurrent program.

Include/Exclude

Select from the poplist whether or not to include or exclude those requests that are based on the rule to run.

Type

Select the type of specialization rule you want to assign to your manager. Based on the rule's action you selected, allow or disallow, requests can be run by your manager according to a:

- Combined Rule

  For example, only requests that satisfy the combined rule you select are allowed to be run by your manager. Or conversely, requests that satisfy a certain combined rule are excluded from running.

  Combined specialization rules, which combine more than one logical statement, are

defined using the Combined Specialization Rules form. See: Combined Specialization Rules, page 4-169.

- ORACLE ID

  For example, programs with a certain ORACLE ID are excluded from running. Or conversely, a concurrent manager only includes programs with a specific ORACLE ID.

- Program

  For example, only the program you select is excluded from running. Or conversely, a concurrent manager only includes the programs you select. You can also include or exclude all programs belonging to a specific application using the Program type by entering the application in the Application field and leaving the Name field empty.

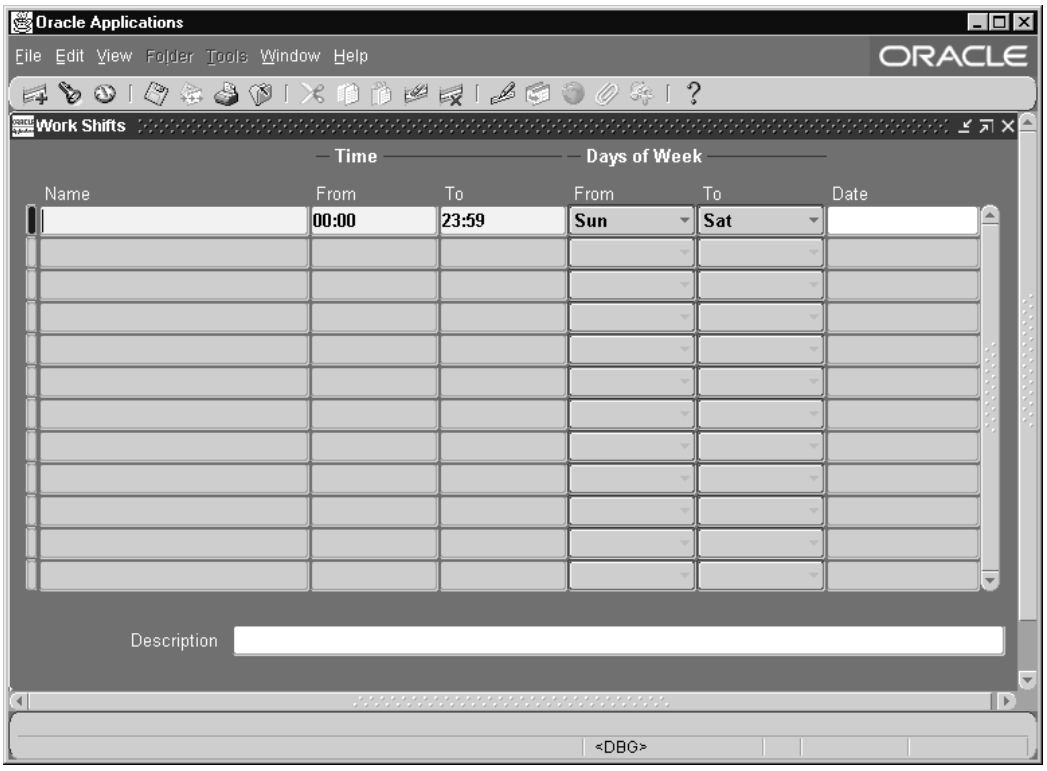
- Request Type (of the program)

  For example, programs of a certain request type are excluded from running. Or conversely, a concurrent manager only includes programs with the request type you select.

- User (application username at sign on)

  For example, all programs submitted by a certain user are excluded from running. Or conversely, a concurrent manager includes only programs submitted by the user you select.

Work Shifts Window



Use this window to name and define your concurrent manager work shifts. Define work shifts to specify when your concurrent managers can work.

For each work shift, specify a time period covering a range of days or a particular date. See: Work Shifts Definitions, page 4-102.

Name

The name of your concurrent work shift should be intuitive, for instance "Week Days", "Weeknights" or "Weekends".

From/To

Enter the times of day at which your concurrent shift begins/ends. The time format is HH24:MM. For example, if your work shift name is "Week Days", you could enter "09:00" (9:00 am) as the start time and "17:00" (5:00 pm) as the end time. Note that Oracle E-Business Suite uses a 24-hour clock.

Days of Week From/Days of Week To

Enter the first and last days of this shift. For instance, if your shift name is "Week Days",

you could enter "Monday" in the "Days of Week From" field and "Friday" in the "Days of Week To" field. If you enter a value in the "Days of Week From" field, you must enter a value in the "Days of Week To field". You may not use the Date field for this row.

Date

Enter a date here to create a date-specific workshift. For instance, you can name a workshift "Memorial Day", and enter the date in this field to enable this workshift only on the Memorial Day holiday.

Date-specific workshifts override workshifts that do not specify a specific date. If you want to enter a value in this field (specify a date), you may not enter values for the Days of Week fields for this row. See: Overlapping Work Shifts - Priority Levels, page 4-103.

Related Topics

Defining Managers and their Work Shifts, page 4-100

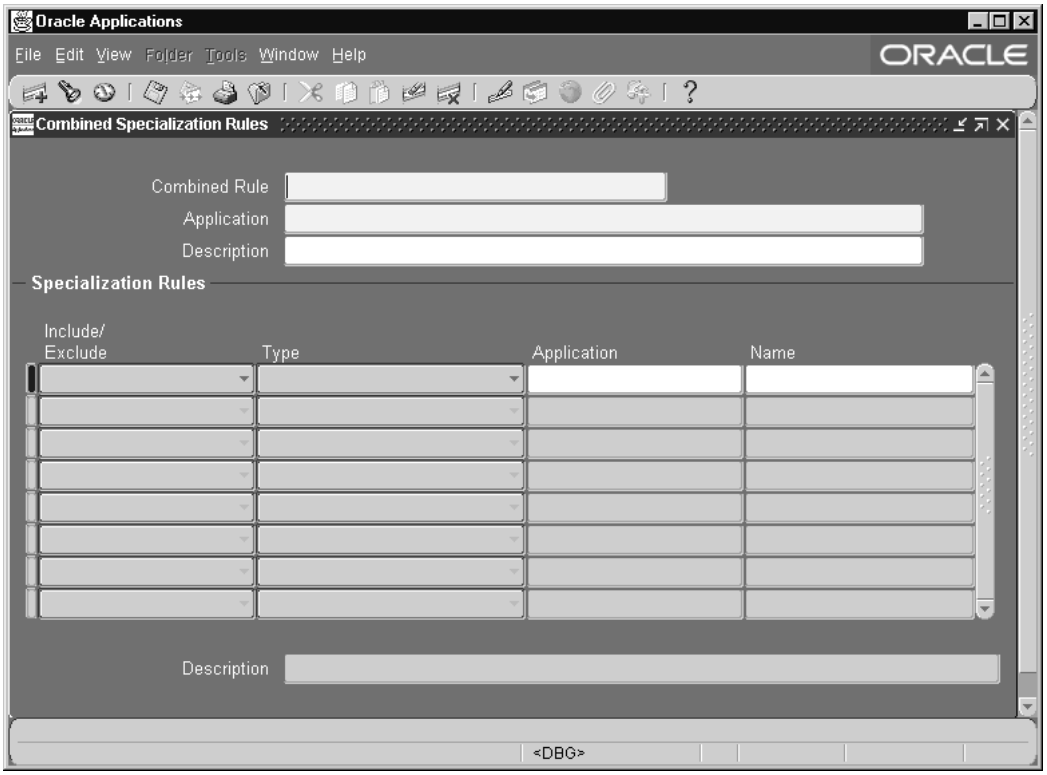
Work Shift by Manager Report, page 4-115

Work Shifts Report, page 4-115

Administer Concurrent Managers field help, page 4-151

Concurrent Managers field help, page 4-160

Combined Specialization Rules Window



Define rules identifying which requests a concurrent manager can read. With the rules you define here, you may specialize the function of a concurrent manager.

Using this window, you can define several Include and Exclude statements, each referred to as a specialization line, and combine the lines into a single specialization rule referred to as a *Combined Rule*.

Unlike the individual rules you define using the Specialization Rules window from within the Concurrent Managers window, the combined rules you define here differ in two ways:

- You can combine Include and Exclude statements. This enables you to identify very specific requests for running concurrent programs.
- Within a combined rule, using multiple Include statements restricts a concurrent manager more.

  With individual rules you define using the Specialization Rules window (within the Concurrent Managers window), the more "Include" rules you define, the less restricted a manager becomes.

  See: Concurrent Managers, page 4-160

Combined Specialization Rules Block

Together, the application name and the name you define for your combined specialization rule uniquely identifies the rule.

Application

The application name does not prevent a concurrent manager from starting programs associated with other applications.

Specialization Rules Block

Define the individual rules (statements) that make up your combined specialization rule.

- Each rule in this block defines one statement.
- The sum of all the specialization rules defines your combined specialization rule.

Include/Exclude

Select from the poplist whether to include or exclude those requests that are based on the rule to run.

Type

Select the type of specialization rule you want to enforce on a concurrent manager.

You cannot combine two Include rules of the same type. For example, you cannot include programs to be associated with an ORACLE ID, then, on another line, include programs to be associated with a second, different ORACLE ID.

Based on a rule's action, exclude or include, programs can be run by your manager according to a:

- ORACLE ID

  For example, programs with a certain ORACLE ID are excluded from running. Or conversely, a concurrent manager only includes programs with a specific ORACLE ID.

- Program

  For example, only the program you select is excluded from running. Or conversely, a concurrent manager only includes the programs you select. You can also include or exclude all programs belonging to a specific application using the Program type by entering the application in the Application field and leaving the Name field empty.

- Request Type (of the program)

  For example, programs of a certain request type are excluded from running. Or conversely, a concurrent manager only includes programs with the request type you select.

- User (application username at sign on)

  For example, all programs submitted by a certain user are excluded from running. Or conversely, a concurrent manager includes only programs submitted by the user you select.

Related Topics

Specializing Managers to run only certain programs, page 4-116

Defining Specialization Rules, page 4-116

Defining Combined Specialization Rules, page 4-126

Using Combined Rules, page 4-128

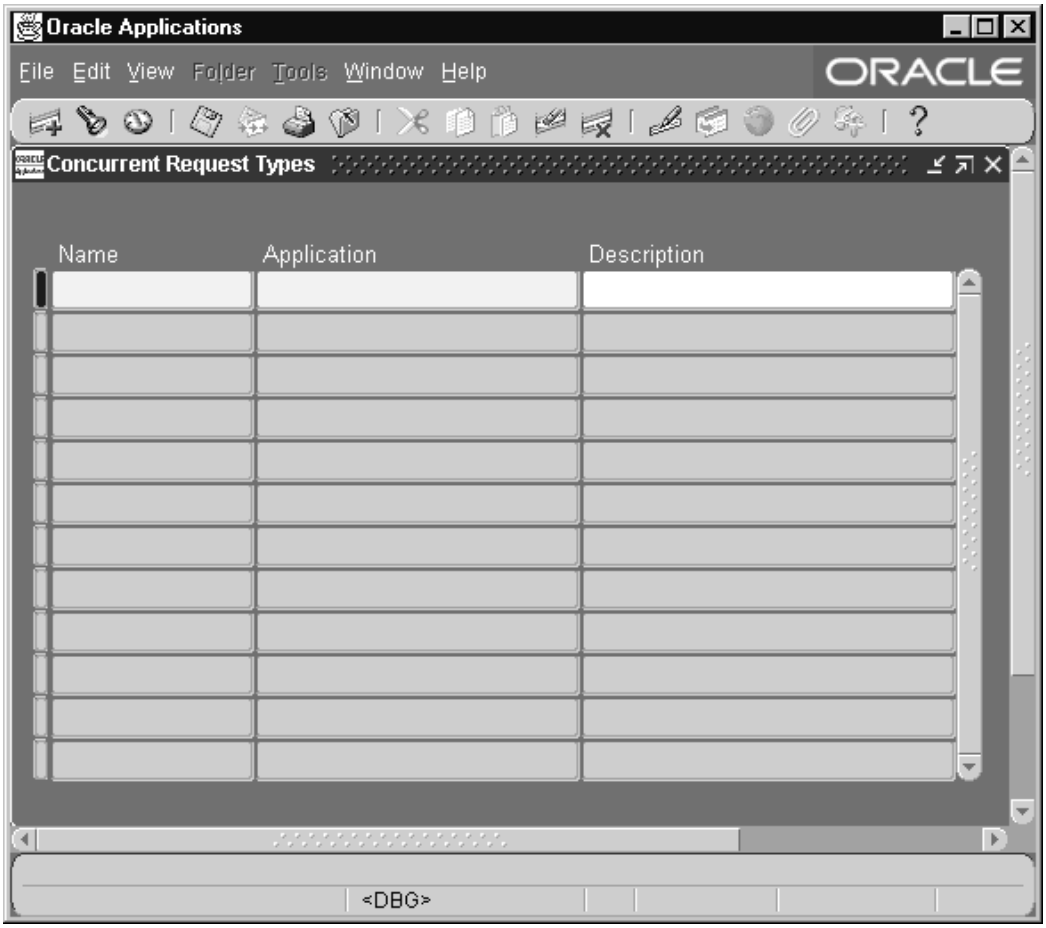
Differences Between Specialization and Combined Rules, page 4-131

Grouping Programs by Request Type, page 4-132

Administer Concurrent Managers field help, page 4-151

Concurrent Managers field help, page 4-160

Concurrent Request Types Window



Use this window to identify several concurrent programs as a group by assigning each program a common *request type*.

You assign a request type defined here to a concurrent program using the Concurrent Programs window. Then, when you define a concurrent manager using the Define Concurrent Manager window, you can define the manager to run (Allow) or not run concurrent programs based on their request type.

For example, you could define a request type as "end-of-month reports", assign that request type to several concurrent programs, then define a concurrent manager to only run "end-of-month" requests.

Concurrent Request Types Block

Name and describe each type of concurrent request you want to define. The combination of application name plus request type uniquely identifies your concurrent request type.

This application name does not prevent you from assigning this request type to concurrent programs associated with other application names.

Related Topics

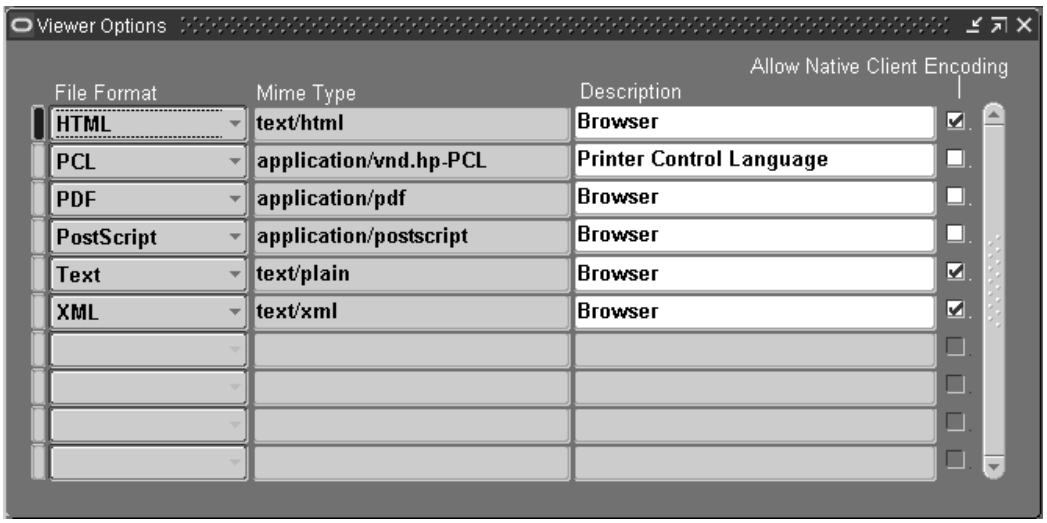
Specializing Managers to run only certain programs, page 4-116

Grouping Programs by Request Type, page 4-132

Concurrent Managers field help, page 4-160

Combined Specialization Rules field help, page 4-169

Viewer Options Window



Use this form to define the MIME types for the output formats of your concurrent requests. These MIME types are used in viewing reports.

For each file format, you can associate one or more MIME types.

A user can use one MIME type to view reports of a certain format. For example, a user can view all text format reports in Microsoft Word. The MIME types for supported formats for a particular user are set by several profile options. They are:

- Viewer: Application for HTML
- Viewer: Application for PCL
- Viewer: Application for PDF
- Viewer: Application for PostScript
- Viewer: Application for Text

This MIME type is sent to a browser window when the user views a report of that file format.

Viewer Options Block

Associate one or more MIME types with each supported file format. By defining viewer options, you can specify the application or applications that are available for displaying files of each format.

File Format

The file format.

MIME Type

The MIME type to use for the file output.

Allow Native Client Encoding

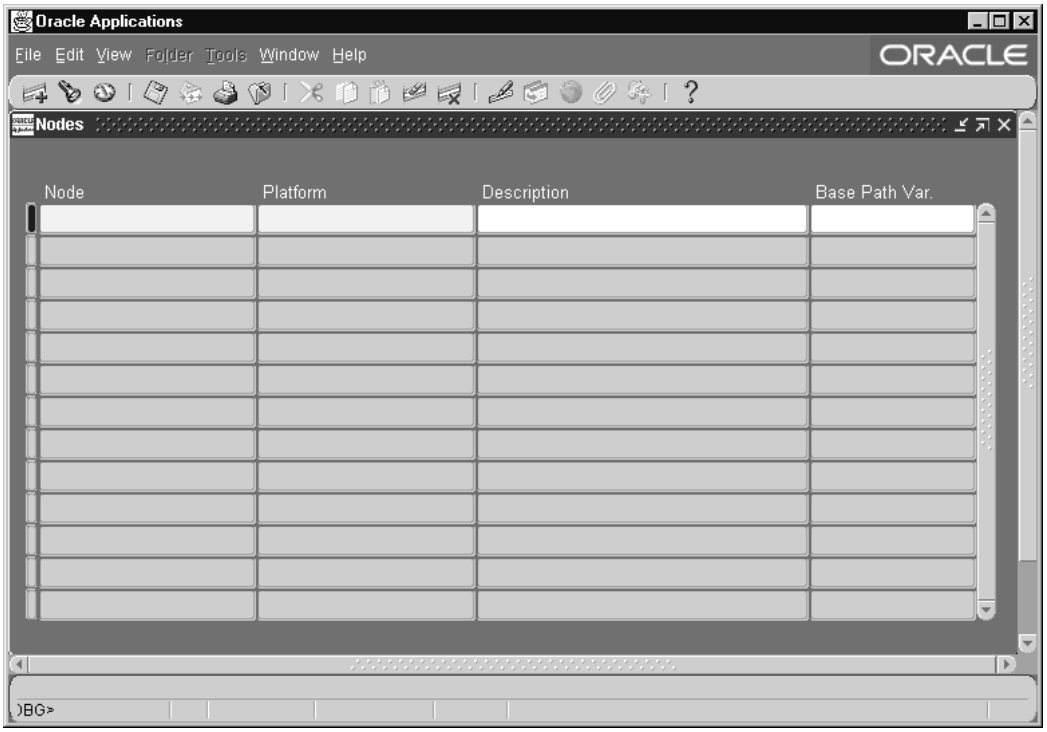
If this box is checked, the Report Viewer will convert the output file into the character set specified by the FND: Native Client Encoding profile option.

Related Topics

Defining the Reports Viewer, *Oracle E-Business Suite Maintenance Guide*

Reviewing Requests, Request Log Files, and Report Output Files, *Oracle E-Business Suite Maintenance Guide*

Nodes Window



A node consists of one or more processors and their associated memory. In parallel concurrent processing environments (such as cluster, massively parallel, and homogeneous networked environments) each node operates independently of other nodes except when sharing resources, such as a disk.

You can assign concurrent managers to different nodes to spread your concurrent processing workload and increase throughput. A concurrent manager runs its processes on the nodes to which it is assigned.

Nodes Block

Node

Enter the operating system name of a node.

Platform

Select the operating system platform that your node resides on.

Base Path Variable

Consult your installation manual to determine the correct base path variable for your

platform to determine the location of the concurrent managers' log and out files for this node.

Related Topics

Overview of Parallel Concurrent Processing, page 4-143

Concurrent Managers field help, page 4-160

Setting Up, Starting, and Shutting Down Concurrent Managers

Oracle E-Business Suite concurrent managers run processes in the background on a server machine. You must set up and start the concurrent managers for each product group before you can use your Oracle E-Business Suite products. Refer to Managing Application Tier Services, *Oracle E-Business Suite Maintenance Guide* for more information on the adcmctl script for starting and stopping concurrent managers. The instructions in this section apply whether you are installing or upgrading.

> **Note:** Run your concurrent managers on machines with hostnames of 30 or fewer characters. Managers may fail to run on machines with longer hostnames.

Setting Up Concurrent Managers

For UNIX

Keep the following in mind when you start the concurrent managers:

- Concurrent managers inherit directory privileges from the user who starts them. If you plan to start the managers from a login other than the main applications login, applmgr, ensure that the login has the appropriate directory privileges.

  For more information on directory privileges, see *Oracle E-Business Suite Concepts*.

Parameter values set in startmgr override any other values. Command line values override environment and default values, and so on. Another section in this manual contains more information on editing the startmgr script.

You can change directory privileges while the managers are running, and the changes will be effective immediately. You can change environment variables and startup parameters while the managers are running, but your changes will not take effect until the concurrent managers are restarted. To put changes into effect, shut down the managers, make the necessary modifications, and restart the managers.

For Windows

On Windows platforms, the OracleConcMgr<SID> service (where <SID> is the database

<SID> and the value of the LOCAL variable) spawns the concurrent manager programs. The OracleConcMgr<SID> service is initially created by the Rapid Install program. If you need to recreate the service, use one of the following two methods:

Method 1

From the command prompt:

```
C:\> cd %COMMON_TOP%\admin\install
C:\> adsvcm.cmd <NT User> <NT Password>
```

where NT User is the user that runs the concurrent manager service, and NT Password is the password of the user that runs the service.

Method 2

Invoke the GUI program ccmsetup.exe (located in %FND_TOP%\bin), and choose the option to create the service.

Parameters

The following entries describe the concurrent manager startup parameters, which are read by the concurrent manager service and passed to the concurrent manager programs. The GUI program ccmsetup.exe (located in %FND_TOP%\bin) should be used to create and set these parameters in the Windows registry. Note that the registry settings override defaults. These parameters can also be used by the adcmctl.cmd script (see Managing Application Tier Services, *Oracle E-Business Suite Maintenance Guide* for more information).

Database Connection

Schema Name — The APPS schema name should be set to the APPS user ID.

Password — This is the password to the Oracle E-Business Suite account in the database.

TNS Aliasname — This is the database <SID>.

Server Startup

Use this Account — Check the "Use this Account box" to specify the OS account (<NT User> and <NT Password>) that runs the CCM service. If you do not check this box, the system account launching the service will not have permission to access network resources.

Startup Options

Sleep — Number of seconds (integer) the internal concurrent

manager waits between times it looks for Queue Controlled (in statuses Deactivate/Abort/Verify/Activate) requests. The default value is 60.

Restart

This parameter is used in the event of abnormal termination. Its value is the number of minutes (integer) the internal concurrent manager (ICM) has to be up/active/running before it exits abnormally in order for it to be automatically restarted. The ICM must be active for at least as long as the time specified by this parameter; that is, the duration between startup and termination must be greater than the duration specified by this parameter. The default value is N. The default value prevents the manager from restarting after abnormal termination.

Queue Size

Number of pmon cycles (integer) the internal concurrent manager waits between times it checks for normal changes in concurrent manager operation. Normal changes include the start or end of a work shift and changes to concurrent manager definitions entered in the Concurrent Managers form. The default value is 1.

PMON cycle

Number of sleep cycles (integer) the internal concurrent manager waits between times it checks for concurrent managers that have failed while running a concurrent request. The default value is 20.

Printer Name

Name of the printer to which the concurrent managers send request output, where the requests are submitted from within a concurrent program if the submitting program (parent request) does not have a printer associated with it.

Enable Distributed Concurrent Processing

Check this box to enable distributed concurrent processing.

Shutdown

Deactivate/ Normal

Shuts down the concurrent managers using normal shutdown methods.

Terminate/Abort Processes

When shutting down the concurrent managers, aborts the concurrent managers.

> **Note:** This option is currently disabled. Checking this box will have no effect.

File Logging

Activation Log The name of the log file generated upon startup of the Internal Concurrent Manager. Defaults to CM_<SID>.log.

Deactivation Log The name of the log file generated upon shutdown of the Internal Concurrent Manager. Defaults to CS_<SID>.log.

Record Diagnostic Messages Checking this box will cause concurrent managers to produce diagnostic output regularly. Note that leaving this box unchecked prevents large log files.

Save Log History Checking this box prevents log files from being overwritten when the concurrent manager is started.

Starting the Concurrent Managers

Before starting the concurrent managers, you must start the Oracle E-Business Suite TNS listener on all nodes. The TNS listener must be started by the applmgr user.

For UNIX

You can start the concurrent managers from by running the script startmgr from the operating system command line. To start the concurrent manager from the operating system prompt, use the following syntax:

```
$ startmgr \
sysmgr="<APPS username>/<APPS password>" \
mgrname="<name>" \
PRINTER="<printer>" \
mailto="<userid1 userid2...>" \
restart="N|<minutes>" \
logfile="<filename>" \
sleep="<seconds>" \
pmon="<cycles>" \
quesiz="<cycles>" \
diag="Y|N"
```

All parameters are optional. You can pass parameters to the script in any order. These parameters can also be used by the adcmctl.sh script (see Managing Application Tier Services, *Oracle E-Business Suite Maintenance Guide* for more information).

Parameters

The following entries describe the concurrent manager startup parameters. The default values apply if you do not specify different values in the startmgr script, on the command line when you run startmgr, or in your environment.

sysmgr APPS schema name should be set to the APPS schema user ID and password. You will be prompted for the password

if you omit the parameter and use the default value. The default value is $FNDNAM.

mgrname Name of the internal concurrent manager (alphanumeric characters only). The default value is std.

PRINTER Name of the printer to which the concurrent managers send request output, where the requests are submitted from within a concurrent program if the submitting program (parent request) does not have a printer associated with it.

mailto List of users who receive mail when the internal concurrent manager stops running. The default value is the user who starts managers.

restart This parameter is used in the event of abnormal termination. Its value is the number of minutes (integer) the internal concurrent manager (ICM) has to be up/active/running before it exits abnormally in order for it to be automatically restarted. The ICM must be active for at least as long as the time specified by this parameter; that is, the duration between startup and termination must be greater than the duration specified by this parameter. The default value is N. The default value prevents the manager from restarting after abnormal termination.

logfile The name of the internal concurrent manager's log file. The default value is <mgrname.mgr>.

sleep Number of seconds (integer) the internal concurrent manager waits between times it looks for Queue Controlled (in statuses Deactivate/Abort/Verify/Activate) concurrent requests. The default value is 60.

pmon Number of sleep cycles (integer) the internal concurrent manager waits between times it checks for concurrent managers that have failed while running a concurrent request. The default value is 20.

quesiz Number of pmon cycles (integer) the internal concurrent manager waits between times it checks for normal changes in concurrent manager operation. Normal changes include the start or end of a work shift and changes to concurrent manager definitions entered in the Concurrent Managers form. The default value is 1.

diag diag=Y tells all concurrent managers to produce diagnostic output regularly. The default value diag=N prevents large log files.

Example

```
$ startmgr sysmgr="apps/password" \
mgrname="std" \
PRINTER="hqseq1" \
mailto="jsmith" \
restart="N" \
logfile="mgrlog" \
sleep="30" \
pmon="5" \
quesiz="2"
```

For Windows

The OracleConcMgr<SID> service is launched from the Control Panel Services applet. It can also be launched from the command line using the following command:

```
C:\> net start OracleConcMgr<SID>
```

- Concurrent managers inherit directory privileges from the user who installs and starts them. If you plan to install and start the managers from a login other than the main applications login (applmgr) ensure that the login has the appropriate directory privileges. The Windows account that starts the concurrent manager service must be the same one that installed it.

- Startup parameter values apply in this order:
  - Values set in the registry (through the use of ccmsetup)
  - Default values

    This means that values set in the registry override the default values.

  You cannot change startup parameters while the managers are running. To put changes into effect, shut down the managers, make the necessary modifications, and restart the managers.

Restarting the Concurrent Managers

You must restart the concurrent managers whenever you start the Oracle8*i* Server database or change the concurrent manager startup parameters.

On UNIX platforms, concurrent managers append to their own log file if the log files exist when they restart. Therefore, the user who restarts the concurrent managers must either own the existing files, have write privilege for them, or delete them before restarting.

On Windows, the concurrent manager logs are overwritten when the concurrent

managers are restarted. Checking the "Save Log History" option in the ccmsetup GUI tool saves the previous log files.

The concurrent managers delete temporary files when each concurrent process finishes. If the concurrent managers stop abnormally, however, they may not delete these files.

> **Tip:** Delete temporary files only if they have not been accessed more recently than a few days ago. This helps to prevent the loss of files required by the operating system or the concurrent managers.
>
> Ideally, delete temporary files during maintenance windows when the database is down and no applications programs are active.

Network Failure Recovery

As part of its shutdown process, the ICM determines if it's being forced to shutdown due to losing its database connection. This is done by looking for specific error messages ORA-3113, ORA-3114, or ORA-1041. If one of these error messages is detected, the ICM spawns the reviver process, which attempts to make a database connection. If unsuccessful, it sleeps for a period before trying again. This continues until either a successful connection is made or it receives a signal to shut itself down.

When a successful connection is made, the process kills the old ICM database session, and then starts a new ICM using the normal start manager script. Once the ICM is restarted, it starts up any other managers that had also shut down, and normal processing resumes.

Shutting Down the Concurrent Manager Service (Windows)

The OracleConcMgr<SID> service may be shut down from the Control Panel Services applet. It can also be stopped from the command line using the following command:

```
C:\> net stop OracleConcMgr<SID>
```

Although you can shut down concurrent managers from Oracle E-Business Suite System Administrator's responsibility, this does not stop the concurrent manager service. You must still stop the concurrent manager service from the Windows Control Panel Services applet before you can restart the concurrent managers.

> **Note:** The OracleConcMgr<SID> service may take several minutes to shut down because it needs to finish processing currently running requests.

> **Warning:** Do not use the Task Manager to stop the concurrent manager service or other Applications processes unless you are advised to do so by Oracle Worldwide Support.

Removing the Concurrent Manager Service (Windows)

If you need to remove the concurrent manager service, ensure that it is not running.

To remove or delete the OracleConcMgr<SID> service, use one of the following 2 methods:

Method 1

At the command prompt, type:

```
C:\> cd %COMMON_TOP%\admin\install
C:\> adsvcm.cmd -deinstall
```

Method 2

Invoke the GUI program ccmsetup.exe, and choose the option to remove the service.

Once you have done this, you will need to reinstall the concurrent manager service in order to process concurrent requests.

Starting and Stopping the Concurrent Manager with the Concurrent Manager Operator User

Traditionally the operator starting and stopping the application services needed to know the APPS username and password in order to start the application services on an application tier that was running the Concurrent Manager. Starting with Release 12.1.3, it is possible to create an Oracle E-Business Suite user (FND User) with the responsibility "Concurrent Manager Operator" and use this user's username and password start and stop the application services. The procedure is the following:

1. Createa new user, for example, "CONCOPER", and assigning the "Concurrent Manager Operator" responsibility to this user.

2. On the application tier, update the following four variables in the AutoConfig context file, as listed in the following table:

AutoConfig Variable	New Value
s_cp_user	CONCOPER (or the one you created)
s_cp_password_type	AppsUser
s_cp_resp_shortname	FND
s_cp_resp_name	Concurrent Manager Operator

3. Run AutoConfig on the application tier(s).

Following this change, the application tier services can be started and stopped by calling `adstrtal.sh` and `adstpall.sh` with the `-secureapps` option and the script will prompt for the Oracle E-Business Suite user's username and password rather than the APPS username and password.

For example

```
[applmgr@app01]$ adstrtal.sh -secureapps
Enter the Applications username: CONCOPER
Enter the Applications password:
```

File Conventions

The following tables list the locations and file naming conventions for log, output, and temporary files. The location of product log and output files depends on whether you have set up a common directory.

For UNIX

File Type	Location	Filename
Internal Concurrent Manager Log	$FND_TOP/$APPLLOG with Common Directory: $APPLCSF/$APPLLOG	<mgrname>.mgr
Concurrent Manager Log	$FND_TOP/$APPLLOG with Common Directory: $APPLCSF/$APPLLOG	w<nnn>.mgr
Request Log	Default: $<PROD>_TOP/$APPLLOG with Common Directory: $APPLCSF/$APPLLOG	l<request ID>.req
Request Output	Default: $<PROD>_TOP/$APPLOUT with Common Directory: $APPLCSF/$APPLOUT	o<request ID>.out
Temporary	$APPLTMP or $REPORTS60_TMP	OF<abcd12345>.t where <abcd12345> is a random OS-generated string

Parameters

The variable parameters have the following values:

mgrname For UNIX, the name specified with the mgrname parameter in the startmgr command. If no name is specified, the filename is std.mgr.

For Windows, the name specified with the mgrname parameter in the ccmsetup.exe program. If no name is specified, the filename is std.mgr.

nnn A sequence number between 1 and 999 is generated by the concurrent processing facility.

<PROD>_TOP The product's top environment variable, such as GL_TOP.

request ID The number that identifies the concurrent request.

USERNAME Up to eight characters (uppercase) of the application username of the user that requested the concurrent process.

<abcd12345> Naming convention in which <abcd> are random letters and <12345> designate the operating system process ID of the concurrent process that generated the file.

Directory Privileges

Oracle recommends that you start the managers from the applmgr login to ensure that they inherit the correct directory privileges. The user who starts the concurrent managers then owns the log and output files that the concurrent managers create.

Oracle recommends that you start the managers from the applmgr login to ensure that they inherit the correct directory privileges.

For UNIX, any user who runs an environment file and has access to the startmgr script can start the concurrent managers.

> **Warning:** Always start the concurrent managers from the applmgr login if you are using parallel concurrent processing on multiple nodes.

For UNIX

This section describes directory privileges for UNIX.

Setting the startmgr User ID with setuid in UNIX

To ensure that startmgr inherits the applmgr directory privileges, you can use the UNIX setuid facility to set startmgr to the applmgr login's UNIX user ID. The concurrent managers then inherit the applmgr privileges no matter which login runs startmgr. This allows you to start or restart concurrent managers using the Administer Concurrent Managers form, regardless of the originating UNIX login. Note that you must reset the user ID with setuid if you modify or copy startmgr. Refer to your online UNIX documentation for information on setuid.

> **Note:** The use of the setuid command may cause unexpected behavior on certain platforms that employ dynamic linking of libraries. Please refer to the Oracle E-Business Suite Installation Update for your platform for any information regarding this problem.

Directory Privileges for Logins other than applmgr

If you do not set the startmgr script to the applmgr user ID and you start the managers from a login other than applmgr, that login needs to have these privileges:

- Read and execute privileges on all Oracle E-Business Suite directories
- Write privilege for all directories defined by the following variables:
  - APPLLOG (typically log directory or directories)
  - APPLOUT (typically out directory or directories)
  - APPLCSF (common directory for log and output files)
  - APPLTMP (temporary directory)
  - REPORTS60_TMP (temporary directory for Oracle Reports files)
- Write privilege for these directories: /tmp and /usr/tmp

You can verify that a login has the necessary privileges on a certain directory with this command:

```
$ ls -ld < directory>
```

Here is an example:

```
$ ls -ld /usr/tmp
drwxrwxrwx 3 root 22880 Mar 10 11:05 /usr/tmp
      ^^^
```

The three letters marked in the sample response indicate that all users have read, write, and execute privilege for the directory.

Printing

This section contains printer reference material including information on how to create and register executable printing programs.

To register printers in the Printers window of Oracle E-Business Suite, your Oracle E-Business Suite System Administrator needs to know each printer's operating system name. Your installation update tells you where to find the printer names for your platform. Installation updates may also contain other information on setting up your printers.

Printing (for UNIX)

This section contains printer reference material specific to the UNIX operating system, including information on how to create and register executable printing programs.

Standard Print Subroutine

The standard printing subroutine that you can select in the Printer Drivers form uses Oracle Application Object Library routines to print reports. This requires fewer machine resources than printing through a customized executable program or a shell command such as lp or lpr.

When you use the subroutine, there may be options available through the descriptive flexfield at the bottom of the form. These options vary by platform and may include the following: mail Notify user by electronic mail when report finishes printing.

- Mail: Notify user by electronic mail when report finishes printing.
- Priority: Set the priority for reports in the print queue.

Check your installation update for any additional options available on your platform.

Executable Printing Programs

Oracle E-Business Suite supports the use of executable programs for printing. However, we recommend that you use executable programs only to provide features unavailable through Oracle E-Business Suite printer drivers, such as:

- Interpreting special characters in the text passed to the printer. For example, you need a program to interpret 8-bit characters sent to a 7-bit compatible printer
- Interpreting arguments passed by Oracle E-Business Suite. For example, you need a program to perform different actions based on different output filenames.

If you do not need to support special features such as these, print through the standard printing subroutine and printer drivers defined in the Oracle E-Business Suite database. This makes the most efficient use of machine resources.

Upgrading to Existing Executable Programs

Because printing through the standard printer subroutine uses machine resources more effectively than printing through executable programs, we recommend the following if you used executable printing programs in the previous release of Oracle E-Business Suite.

- If predefined printer drivers can replace the executable program, simply register the drivers along with the printer types in the Printer Types window.

  For example, Oracle E-Business Suite provides a print style Landscape and the printer driver LANDSCAPESUB. They perform the same function as the program land, which enables DEC LN03 printers to print 132 characters per line.

- If no predefined drivers will work, you may be able to create a customized driver that can replace the executable. You create drivers with the Printer Drivers form.

- If you cannot replace the executable with a simple printer driver definition, you can continue to use the executable by registering it or the shell script that calls it with Oracle E-Business Suite.

Writing an Executable Program

Executable printing programs can format report output through escape sequences or a printer programming language. Creating them requires a thorough knowledge of both printer operation and a computer programming language. Follow the guidelines in this section if you need to create an executable printing program.

Printer Styles

An executable program should be able to format report output for various print styles, including these:

- Portrait: 80 columns wide, 66 lines per page

- Landscape 132 characters wide, 66 lines per page (62 lines per page on A4 style paper)

- Landwide 180 characters wide, 66 lines per page (62 lines per page on A4 style paper)

Formatting Arguments

If the program handles formatting for various print styles internally, you can pass arguments from the printer drivers to the program to determine which print style to use.

If the program does not contain print style formatting commands, you can define the commands in a shell script that calls the program. You then define the shell script as the printing program in a printer driver and pass arguments that determine the print style from the driver to the script.

The printer driver that calls the executable program or shell script must be able to pass the following arguments:

- Name of the destination printer
- Number of copies to print
- Banner on title page
- Filename

Initialization and Reset

You do not have to add printer initialization and reset strings to your program if you can define these strings in the Printer Drivers form.

Character Mode Oracle Reports Commands

We recommend that you design your executable programs to work with the standard Oracle Reports print drivers. The following standard drivers are located in the $FND_TOP/$APPLREP directory:

- P.prt - Portrait style
- L.prt - Landscape style
- W.prt - Landwide style
- A.prt - A4 style

The program should not misinterpret the commands for bold on, bold off, and page size that the standard drivers imbed in Oracle E-Business Suite reports. If necessary, you can create customized Oracle Reports drivers as described below.

Location of Program

When you have compiled and linked the source code or written a shell script, move the program to the $APPLBIN subdirectory under the top directory of your custom development area. Keep copies of the source file in your custom development area as a backup.

Creating Customized Character Mode Oracle Reports Print Drivers

The Oracle Reports print drivers set the font styles for italics, underlining, and bolding. If your executable printing program cannot use the standard Oracle Reports drivers, create a customized driver for each print style you will use with the program.

To create a customized driver, copy L.prt, P.prt, A.prt, or W.prt from $FND_TOP/$APPLREP to your custom development area. Modify a standard driver as needed for your executable printing program. Give the customized driver a new filename but keep the .prt extension. Then copy the customized driver to $FND_TOP/$APPLREP.

Tip: Use the executable program name and print style letter as the

driver name. For example, use HPLJ3P.prt for portrait style printing with the executable program HPLJ3. When you print in portrait style with this program, the concurrent managers pass the HPLJ3P.prt driver to Oracle Reports as DESFORMAT=HPLJ3P.

Registering Executable Programs

When you have created your executable programs and, optionally, your shell scripts and Oracle Reports drivers, register them in the Printer Drivers form.

1. Navigate to the Printer Drivers window and create a new printer driver name. Also add the user name, description, and platform.
2. In the SRW Driver field, enter the name (without the .prt extension) of a standard or customized Oracle Reports driver. All drivers must be in the directory $FND_TOP/$APPLREP.
3. Enter Program as the driver method.
4. Enter No in the Spool File field.
5. Enter No in the Standard Input field.
6. In the program name field, enter the name of the executable program or the shell script that calls it. Include the full path name if this file is not in the $FND_TOP/bin directory.
7. Add the arguments that Oracle E-Business Suite passes to the program or shell script. The driver must pass the following to the executable program:
   - Name of the destination printer
   - Number of copies to print
   - Banner on title page
   - Filename
   - Add the initialization and reset strings to the appropriate fields if the program does not send these strings to the printer.

Printing (For Windows)

This section contains printer reference material specific to the Windows operating system. It also explains how to create and register executable printing programs.

Operating System Names for Printers

To register printers in the Printers window of Oracle E-Business Suite, your Oracle E-Business Suite system administrator needs to know each printer's operating system name. For Windows, you can obtain the printer's name from the Printers folder in My Computer. If you register an invalid printer, the operating system's default printer will be used instead.

Standard Print Subroutine

The standard printing subroutine that you can select in the Printer Drivers form uses Oracle Application Object Library routines to print reports. This requires fewer machine resources than printing through a customized executable program DOS command such as PRINT.

Executable Printing Programs

Oracle E-Business Suite supports the use of executable programs for printing. However, we recommend that you use executable programs only to provide features unavailable through Oracle E-Business Suite printer drivers, such as:

- Interpreting special characters in the text passed to the printer. For example, you need a program to interpret 8-bit characters sent to a 7-bit compatible printer.

- Interpreting arguments passed by Oracle E-Business Suite. For example, you need a program to perform different actions based on different output filenames.

If you do not need to support special features such as these, print through the standard printing subroutine and printer drivers defined in the Oracle E-Business Suite database. This makes the most efficient use of machine resources.

Upgrading Existing Executable Programs

Because printing through the standard printer subroutine uses machine resources more effectively than printing through executable programs, we recommend the following if you used executable printing programs in the previous release of Oracle E-Business Suite:

- If predefined printer drivers can replace the executable program, simply register the drivers along with the printer types in the Printer Types form.

  For example, Oracle E-Business Suite provides a print style Landscape and the printer driver LANDSCAPESUB. They perform the same function as the program land, which enables DEC LN03 printers to print 132 characters per line.

- If no predefined drivers will work, you may be able to create a customized driver that can replace the executable. You create drivers with the Printer Drivers form.

- If you cannot replace the executable with a simple printer driver definition, you can

continue to use the executable by registering it or the shell script that calls it with Oracle E-Business Suite.

Writing an Executable Program

Executable printing programs can format report output through escape sequences or a printer programming language. Creating them requires a thorough knowledge of both printer operation and a computer programming language. Follow the guidelines in this section if you need to create an executable printing program.

Printer Styles

An executable program should be able to format report output for various print styles, including these:

- Portrait: 80 columns wide, 66 lines per page
- Landscape 132 characters wide, 66 lines per page (62 lines per page on A4 style paper)
- Landwide 180 characters wide, 66 lines per page (62 lines per page on A4 style paper)

Formatting Arguments

If the program handles formatting for various print styles internally, you can pass arguments from the printer drivers to the program to determine which print style to use.

If the program does not contain print style formatting commands, you can define the commands in a command file that calls the program. You then define the .cmd file as the printing program in a printer driver and pass arguments that determine the print style from the driver to the script.

The printer driver that calls the executable program or .cmd file must be able to pass the following arguments:

- Name of the destination printer
- Number of copies to print
- Banner on title page
- Filename

Initialization and Reset

You do not have to add printer initialization and reset strings to your program if you can define these strings in the Printer Drivers form.

Character Mode Oracle Reports Commands

We recommend that you design your executable programs to work with the standard Oracle Reports print drivers. The following standard drivers are located in the %FND_TOP%\%APPLREP% directory:

- P.prt - Portrait style
- L.prt - Landscape style
- W.prt - Landwide style
- A.prt - A4 style

The program should not misinterpret the commands for bold on, bold off, and page size that the standard drivers imbed in Oracle E-Business Suite reports. If necessary, you can create customized Oracle Reports drivers as described in the next section.

Location of Program

When you have compiled and linked the source code or written a command file, move the program to the %APPLBIN% subdirectory under the top directory of your custom development area. Keep copies of the source file in your custom development area as a backup.

Creating Customized Character Mode Oracle Reports Print Drivers

The Oracle Reports print drivers set the font styles for italics, underlining, and bolding. If your executable printing program cannot use the standard Oracle Reports drivers, create a customized driver for each print style you will use with the program.

To create a customized driver, copy L.prt, P.prt, A.prt, or W.prt from %FND_TOP%\%APPLREP% to your custom development area. Modify a standard driver as needed for your executable printing program. Give the customized driver a new filename but keep the .prt extension. Then copy the customized driver to %FND_TOP%\%APPLREP%.

> **Tip:** Use the executable program name and print style letter as the driver name. For example, use HPLJ3P.prt for portrait style printing with the executable program HPLJ3. When you print in portrait style with this program, the concurrent managers pass the HPLJ3P.prt driver to Oracle Reports as DESFORMAT=HPLJ3P.

For more information, see the Oracle Reports Developer documentation.

Registering Executable Programs

When you have created your executable programs and, optionally, your Oracle Reports drivers, register them in the Printer Drivers form.

Define a printer driver and corresponding print style for each print style that your executable program supports. Complete the following steps to register an executable program for a printer driver:

1. Navigate to the Printer Drivers form and create a new printer driver name. Also add the user name, description, and platform.

2. In the SRW Driver field, enter the name (without the .prt extension) of a standard or customized Oracle Reports driver. All drivers must be in the directory %FND_TOP%\%APPLREP%.

3. Enter Program as the driver method.

4. Enter No in the Spool File field.

5. Enter No in the Standard Input field.

6. In the Program Name field, enter the name of the executable program or the command file that calls it. Include the full path name if this file is not in the %FND_TOP%\bin directory.

7. Add the arguments that Oracle E-Business Suite passes to the program or command file. The driver must pass the following to the executable program:

   - Name of the destination printer
   - Number of copies to print
   - Banner on title page
   - Filename
   - Add the initialization and reset strings to the appropriate fields if the program does not send these strings to the printer.

Related Topics

Overview of Printers and Printing, page 6-1

Printer Drivers Window, page 6-53

Managing Concurrent Processing with Oracle Applications Manager

The Oracle Applications Manager allows administrators to manage E-Business Suite systems from an HTML console. Oracle Applications Manager can be used for a wide variety of tasks such as administering services including concurrent managers, examining system configuration, managing Oracle Workflow, examining applied patches, and measuring system usage.

Oracle Applications Manager provides diagnostic features for Applications systems. The console displays errors recently reported by system components such as transaction managers or concurrent requests. For running processes such as forms or concurrent requests, system administrators can examine the database session details, including any currently executing SQL.

Oracle Applications Manager allows administrators to configure, monitor, and control concurrent processing. Combined with the Service Management feature, Oracle Applications Manager can be used to monitor and control concurrent managers, as well as other application tier services.

Using the Oracle Applications Manager, you can:

- view a summary of concurrent managers
- view details of a concurrent manager
- create or edit a concurrent manager
- view a summary of concurrent requests
- view details of a concurrent request
- submit a concurrent request
- search for a concurrent request based on its attributes, date submitted or completed, or its duration or wait time.

Service Instances

The Service Instances pages contain detailed information on the service instances for a particular service type, and display functions you can perform on the services.

Service types include, but are not limited to, the following:

- Internal Concurrent Manager
- Conflict Resolution Manager
- Scheduler/Prerelease Manager
- Request Processing Manager
- Internal Monitor
- Transaction Manager

The information and functionality available depends on the service type. Information may include the following:

- Status - Click on the Status icon for a service to see more information.
- State - The current state of a service. If you perform an action on that service, the state column value is updated.
- Node - In a parallel concurrent processing environment, a service's processes are

targeted to run on the node displayed here. If a service is defined to use a platform-specific system queue, this column displays the name of the queue to which the service submits its processes.

- Number of Running Requests
- Number of Pending Requests
- Actual Processes - The number of operating system processes. Typically, the number of actual processes equals the number of target processes (the maximum number of requests a service can run). However, the number of actual processes may be less than the number of target processes service deactivation or service migration.
- Target Processes - This column displays the maximum number of service processes that can be active for this service.

Controlling Service Instances

You can select a service instance and use the drop down menu above the table to perform the actions listed below. Or you can use the drop down menu at the top right to perform a single action on all service instances.

Service Instances of a Request Processing Manager

This page shows you information on service instances for a request processing manager. This type of manager runs concurrent requests.

Navigation: Applications Systems > System Activity > (Services region) Request Processing Manager

The following information is displayed:

- Status
- State
- Node
- Number of Running Requests
- Number of Pending Requests
- Actual Processes
- Target Processes
- Details (Show/Hide) - If you choose Show, the sleep interval will be displayed.

You can use the buttons at the top to perform the following on a selected service instance:

- Delete
- Edit
- View Status
- View Processes
- View Concurrent Requests

To create a new service instance, use the **Create New** button.

Start

You can start (activate) a service instance.

Stop

You can deactivate individual services. Once deactivated, a service does not restart until you select the service and choose the Start button.

When you deactivate a manager, all requests (concurrent programs) currently running are allowed to complete before the manager shuts down.

Restart

When you restart a manager, the processes are shut down and then brought back up.

Abort

You can abort or terminate individual services.

Concurrent Manager Service Status

For concurrent managers, the following information is shown:

General

- Node - the node on which the concurrent manager is running
- Debug - this setting indicates whether debugging information is recorded in the concurrent manager log file. Set this option to "On" using the **Set Debug On** button to record debugging information.
- Sleep Interval - the number of seconds your manager waits between checking the list of pending concurrent requests (concurrent requests waiting to be started).

Processes

- Target

- Active

Concurrent Requests

- Pending
- Stand by
- Running

Processes

The Processes page shows information on the concurrent processes of a service instance. You navigate to this page from the Service Instances page for a service.

Navigation: Site Map - Administration > Service Status (under Application Services) > (Services region) [Service] > (B) View Processes

You navigate to this page from the Service Instances page for a service.

The following information is given for each process:

- Status - The status of the process. The following are valid statuses:
  - Active - Currently running service processes display as "Active".
  - Deactivated - Manager processes that were explicitly deactivated by a system administrator, either by deactivating the service or by shutting down the Internal Concurrent Manager.
  - Migrating - Services that are migrating between primary and secondary nodes display as "Migrating". In a parallel concurrent processing environment, services run on either the primary or secondary node assigned to them. Services migrate to the secondary node if the primary node or the database instance on the primary node is unavailable. Services migrate back to the primary node once it becomes available.
  - Terminating - service processes that are being terminated display as "Terminating". These processes were terminated by you choosing the Terminate button in the Administer Concurrent Managers form, by you choosing Abort in the Service Instances page, or by a user selecting "Terminate" in the Concurrent Requests form.
  - Terminated - service processes that have been terminated display as "Terminated". These processes were terminated by you choosing the Terminate button in the Administer Concurrent Managers form, by you choosing Abort in the Service Instances page, or by a user selecting "Terminate" in the Concurrent Requests form.
- SPID - The operating system process ID associated with the service process.

- AUDSID - The database session ID for the service process. If the AUDSID value appears as a link, you can click on the value to bring up the Database Session Information page.
- Oracle SPID - The ORACLE system process ID associated with the service process.
- Start Date - The start date for the process.

You can use the buttons to view the following:

- Environment - The environment variable values for this service instance.
- Manager Log - The manager log.
- ICM Log - The Internal Concurrent Manager log.

This page can be added to the Support Cart.

Service Instances for a Service Manager

This page shows you information on service instances for a service manager. Service managers perform actions on behalf of the Internal Concurrent Manager (ICM). They are controlled automatically by the ICM as needed and cannot be manually controlled.

Navigation: Applications Systems > System Activity > (Services region) Service Manager

The following information is displayed:

- Status
- State
- Node

You can use the buttons at the top to perform the following on a selected service instance:

- View Status
- View Processes

Service Instances for the Internal Concurrent Manager

This page shows you information on the service instance for the Internal Concurrent Manager (ICM).

Navigation: Applications Systems > System Activity > (Services region) Internal Concurrent Manager

The following information is displayed:

- Status
- State
- Node
- Number of Pending Requests - for the ICM, these are either service control requests (activate, deactivate, etc.) or requests marked for termination.
- Details (Show/Hide) - If you choose **Show,** the sleep interval will be displayed.

You can use the buttons at the top to perform the following on the service instance:

- View Status
- View Processes
- View Actions
- Edit

Controlling Service Instances

You can select the service instance and use the drop down menu above the table to perform the actions below.

Stop

You can stop (deactivate) an individual service.

When you stop the Internal Concurrent Manager, all other managers are deactivated as well. Managers previously deactivated on an individual basis are not affected.

Any service that was active when the ICM was stopped will be restarted when the ICM is brought back up. Managers that were deactivated on an individual basis will not be brought back up with the ICM.

Stop All

Use this function to stop all services.

Stop Selective

Use this function to select which services you want to stop, and then stop only those services.

Abort

You can abort or terminate individual services.

When you abort (terminate) requests and terminate the Internal Concurrent Manager, all running requests (running concurrent programs) are terminated, and all managers are terminated. Managers previously deactivated on an individual basis are not affected.

Any service that was active when the ICM was aborted will be restarted when the ICM is brought back up. Managers that were deactivated on an individual basis will not be

brought back up with the ICM.

Verify

The Internal Concurrent Manager periodically monitors the processes of each concurrent manager. You can force this process monitoring, or PMON activity, to occur by choosing the Verify action.

Status Overview

System Activity - Status Overview

This page displays a list of the system's application tier services and their statuses. It also lists the number of actual processes and target processes.

Navigation: Applications Dashboard > System Activity (drop-down menu)

You can select a service and use the **View Details** button to view more information on that service, as well as perform certain actions on them.

- Service Instances
- Internal Concurrent Manager
- Conflict Resolution Manager
- Scheduler/Prerelease Manager
- Request Processing Manager
- Internal Monitor
- Transaction Manager

Click the **View All** button to see all services listed. Click the **View Set** button to view the listing in sets of ten.

Click on the **Activity Monitors** tab to see information on Database Sessions and Concurrent Requests.

Service Instances for the Conflict Resolution Manager

This page shows you information on service instances for the Conflict Resolution Manager (CRM).

Navigation: Applications Systems > System Activity > (Services region) Conflict Resolution Manager

The following information is displayed:

- Status

- State
- Node
- Number of Pending Requests - the number of Pending/Standby requests. For each Pending/Standby request, the CRM will evaluate the constraints (such as incompatibilities, single thread, user limit, etc.) and change the request to Pending/Normal when appropriate.

You can use the buttons at the top to perform the following on a selected service instance:

- View Status
- View Processes
- View Concurrent Requests
- Edit

Controlling Service Instances

You can select a service instance and use the drop down menu above the table to perform the actions below. Or you can use the drop down menu at the top right to perform a single action on all service instances.

Verify

You can use the Verify option for the Conflict Resolution Manager to force it to "re-cache" its information on incompatibilities among concurrent programs. Concurrent programs may be defined to be incompatible with other programs; that is, they should not run simultaneously with each other because they might interfere with each other's execution.

The Conflict Resolution Manager will also re-cache its information on users. A user may be assigned a maximum number of requests that may be run simultaneously using the "Concurrent: Active Requests Limit" profile option. The Conflict Resolution Manager rebuilds its list of users when you choose Verify.

Service Instances for a Scheduler/Prerelease Manager

This page shows you information on service instances for a Scheduler/Prerelease Manager. The Scheduler checks for and manages requests with advanced schedules.

Navigation: Applications Systems > System Activity > (Services region) Scheduler/Prerelease Manager

The following information is displayed:

- Status

- State
- Node
- Actual Processes
- Target Processes

You can use the buttons at the top to perform the following on a selected service instance:

- View Status
- View Processes
- Edit

Controlling Service Instances

You can use the dropdown list to **Verify** a Scheduler/Prereleaser Manager.

Service Instances of an Internal Monitor

This page shows you information on service instances for an Internal Monitor. The purpose of an Internal Monitor is to monitor the Internal Concurrent Manager and restart it when it exits unexpectedly.

Navigation: Applications Systems > System Activity > (Services region) Internal Monitor

The following information is displayed:

- Status
- State
- Node
- Actual Processes
- Target Processes
- Details (Show/Hide) - If you choose **Show,** the sleep interval will be displayed.

You can use the buttons at the top to perform the following on a selected service instance:

- Delete
- Edit
- View Status

- View Processes

To create a new service instance, use the **Create New** button.

Controlling Service Instances

You can select a service instance and use the drop down menu above the table to perform the actions below. Or you can use the drop down menu at the top right to perform a single action on all service instances.

Start

You can start (activate) a service instance.

Stop

You can deactivate individual services. Once deactivated, a service does not restart until you select the service and choose the Start button.

Abort

You can abort or terminate individual services.

Service Instances of a Transaction Manager

This page shows you information on the transaction manager. service instances.

Navigation: Site Map > Transaction Managers (under Application Services)

The following information is displayed:

- Details (Show/Hide) - Click **Show** to display the Sleep Interval setup for the selected Transaction Manager and the percent Estimated Availability. The sleep interval can be edited by clicking the **Edit** button.

- Name - Drills down to the **Service Instances Processes** page.

- Status - Drills down to the **Status** page for the selected transaction manager.

- State - The current state of a service. If you perform an action on that service, the state column value is updated.

- Node- In a parallel concurrent processing environment, a service's processes are targeted to run on the node displayed here. If a service is defined to use a platform-specific system queue, this column displays the name of the queue to which the service submits its processes.

- Actual Processes - The number of operating system processes. Typically, the number of actual processes equals the number of target processes (the maximum number of requests a service can run). However, the number of actual processes may be less than the number of target processes due to service deactivation or service migration.

- Target Processes - This column displays the maximum number of service processes

that can be active for this service.

- Timeouts - the number of timeouts that have occurred for this manager since its last activation.

You can use the buttons at the top to perform the following on a selected service instance:

- Delete
- Edit - Launches the **Edit Manager** page.
- View Status - Launches the **Status** page.
- View Processes - Launches the Service Instances Processes page.

To create a new service instance, use the **Create New** button.

Transaction Manager Diagnostics

The following features can help you diagnose transaction manager issues:

Set Debug Level

Use the drop-down list to set the debug level for the transaction manager. Choose one of the following options and click the **Set Debug Level** button. This will set the debug level for all Transaction Managers and will be enabled for future sessions.

- Client side debugging
- Both Client and Server side debugging
- Server side debugging
- Off

> **Note:** Because debugging can adversely affect performance, it is important to turn it off when you are finished.

Time Transaction Manager

If a transaction manager is performing poorly, use the Time Transaction Manager feature to help diagnose the source of the problem. The Time Transaction Manager test reports the time consumed by each activity involved in a single transaction.

To run the test, select a transaction manager and click the Time Transaction button. This will invoke the Time Transaction Manager launch page. Click the **Run Test** button. The test results page will display the following information:

- Elapsed Time - the total time required to complete the test.
- Program - the test program name.

- User - the user ID of the initiator of the test. Drills down to the User Details page.
- Session ID
- Transaction ID
- Time - the time the activity began.
- Source Type - the type of activity and whether it was initiated by the client or the server. If you activated client-side only or server-side only the test will show only those activities of the selected source. To see both, select Both Client and Server side debugging.
- Action - description of the activity
- Message - any message returned by the activity
- Function - the PL/SQL function
- Elapsed Time (in hundredths of seconds)

From this screen, click **Finish Test** to return to the **Service Instances** page, or click **Purge** to purge the debug information for the session.

Controlling Service Instances

You can select a service instance and use the drop down menu above the table to perform the actions listed below. Or you can use the drop down menu at the top right to perform a single action on all service instances.

Start

Starts (activates) a service instance.

Stop

Deactivates individual services. Once deactivated, a service does not restart until you select the service and choose the **Start** button.

When you deactivate a manager, all transaction requests currently running are allowed to complete before the manager shuts down.

Restart

When you restart a transaction manager, its processes are shut down and then brought back up.

Abort

You can abort or terminate individual services.

OAM Generic Collection Service

The OAM Generic Collection Service is a generic service managed by Generic Service Management. It provides file uploading, signaling, purging, and other management for

other service runtime processes such as the Forms Listener runtime process.

A running instance of the OAM Generic Collection service includes a main process which uses the java service cartridge API to consume the messages in the Generic Service Management Advanced Queue (AQ). After the service instance is started, it spawns four subprocesses:

- Forms runtime instance upload process, which uploads the Forms runtime instance files from the node to the Oracle E-Business Suite database periodically based on the load interval.
- On-demand runtime instance upload process, which uploads the Forms runtime instance files based on the custom message received from the AQ.
- On-demand Forms Runtime Diagnostics (FRD) and termination signaling process, which signals the Forms runtime process to generate an FRD log for FRD messages, or terminates the runtime process, producing a termination message. The message is the custom message received from the AQ.
- Forms runtime instance purge process, which purges the runtime instance tables and FRD log files. The numbers of days to keep these data are set as service parameters.

There is only one OAM Generic Collection Instance running per application system per node.

The OAM Generic Collection Service takes these parameters:

- NODE: the name of the node on which the service runs.
- LOADINTERVAL: the load interval for periodic runtime instance information uploading.
- ORACLE_HOME: the ORACLE_HOME in which the Forms Listener runs.
- RTI_KEEP_DAYS: the number of days to keep the runtime instance data in the database.
- FRD_KEEP_DAYS: the number of days to keep Forms Runtime Log files.

Concurrent Processing Charts and Reports

Concurrent Processing Charts

Main Navigation Path: Site Map > Monitoring (subtab) > Performance (heading) > Concurrent Processing Charts (link)

Overview

Oracle Applications Manager offers a number of configurable charts for monitoring the performance of concurrent processing.

There are the following groups of charts:

- Concurrent Requests
- Concurrent Managers
- Utilization

In the Concurrent Requests group, there are several charts, such as "Current Requests by Status," "Running Requests per Application," and "Pending Requests per Responsibility". In the Concurrent Managers group, there are charts such as "Pending Requests per Manager". In the Utilization group, there is a chart that depicts how many running requests and available processes exist per manager.

To view a chart, click its name in the table. In some cases, the charts are interactive and you can drill down on a particular bar or segment to see more details.

To set up a chart, click the **Chart Setting** icon. On the Change Chart Settings page, you can modify the chart type, refresh interval, and data items of a chart.

Concurrent Processing Activity Reports

Navigation: Site Map - Monitoring > Concurrent Processing Reports (under Usage)

Launch the Concurrent Processing Activity Reports from this page. The concurrent processing statistics reports enable you to analyze historical trends relating to request runtimes, success rates, and individual user requests.

- Concurrent Request Statistics by Program
- Concurrent Request Statistics by Username
- Concurrent Program Statistics by Name

Concurrent Request Statistics by Program

Navigation: Site Map - Monitoring > Concurrent Processing Reports (under Usage) > Concurrent Request Statistics by Program

This report summarizes concurrent request statistics by program. These statistics can be useful when scheduling requests or balancing load across nodes (using specialization rules). This report is based on data in the fnd_concurrent_requests table, and is limited to the data in that table since the last time the table was purged using the "Purge Concurrent Request and/or Manager Data" concurrent program.

By default, the report displays data for the past week. Use the Search Criteria region to filter the results based on Application, Minimum Duration, and the reporting time

period. The default sort order is by Total duration in descending order. All duration values are in minutes.

- Application
- Program
- Total - the total of all individual runtimes for the program.
- Average - the average runtime for this program.
- Minimum - the shortest individual runtime for this program.
- Maximum - the longest individual runtime for this program.
- Times Run - the number of times this program has been run. This field drills down to the Search Results page showing the list of requests.

You can select a row for a concurrent program and click the Requests button to drill down to the Search Results page showing the list of requests.

Concurrent Request Statistics by Username

Navigation: Site Map > Concurrent Processing (under Activity) > Concurrent Request Statistics by Username

This report summarizes the concurrent request statistics by username. These statistics can be useful to determine the usage pattern of different users. The columns displayed in the report are:

- Username - click on the username to drill down to the User Details page.
- Requests Completed (number) - drills down to the Search Results page showing the list of requests.
- Total Runtime - the total runtime for all the requests submitted by the user (in hours).

By default, the report displays data for the past week grouped by username. Use the Search Criteria region to filter the results based on Username, Minimum Total Runtime, and the reporting time period.

You can select a row for a username and click the **Requests** button to drill down to the Search Results page showing the user's list of requests.

User Details

This page is accessed by drilling down on the Username field from those pages which display it.

The following contact information is displayed for the username (if available). Data is retrieved from the FND_USER table

- User Name
- Full Name
- Phone
- Phone
- E-mail
- Fax

Programs Usage Statistics Report

Navigation: Site Map > Monitoring > Concurrent Processing Reports (under Usage) > Programs Usage Statistics Report

This report provides a summary of statistics on concurrent programs. Summary information is collected when a request is completed, and stored in the table fnd_conc_prog_onsite_info.

Filter the display on this page by Application or Program name.

> **Note:** Statistics recorded here are as of the Reset Date. The reset date can be viewed on the Program Runtime Statistics page.

The report includes the following fields:

- Application - the application to which the concurrent request belongs
- Program - the program name drills down to the Program Runtime Statistics page.
- Average - the average runtime for this program in seconds.
- Minimum - the shortest individual runtime for this program in seconds.
- Maximum - the longest individual runtime for this program in seconds.
- Times Run - the total number of times the report has been run.
- Success Rate - the percent of the total requests that completed with a Normal status.
- Total Time - the total runtime in seconds for all completed submissions of this program.

By default, the report is ordered by Times Run in descending order. Click the View Details button to display the Program Runtime Statistics page for the selected program.

Program Runtime Statistics

The following fields are shown for the concurrent program selected from the

Concurrent Program Statistics by Name page:

- Last Run Date - the date and time this program was last run.
- Last Run Request ID
- Reset Date - the date and time from which these statistics have been gathered.
- Times Successful - the number of times this program has completed with a status of Normal.
- Times Warning - the number of times this program has completed with a status of Warning.
- Times Error - the number of times this program has completed with a status of Error.

Viewing Concurrent Requests in Oracle Applications Manager

Oracle Applications Manager enables you to view details of concurrent requests. You can view concurrent requests by category or search for requests by specified criteria.

The Concurrent Requests pages can be accessed at:

Site Map > Monitoring > Concurrent Requests (under Current Activity)

Completed (Last Hour) Concurrent Requests

Choose either Table View or Chart View. The Chart View displays a graph of the completed requests by Status.

The Table View displays the following fields:

- Request ID
- Short Name
- Program Name
- Completion Status - the status in which the request completed. Valid statuses are Normal, Error, Warning, Cancelled, and Terminated.
- Requestor - drills down to the User Details page.
- Duration - the amount of time required for the request to run in hours, minutes, and seconds (HH:MM:SS).
- Started At - the time the request actually started running.

Also, you can click on "Show" under the Details column to see additional details for a request, such as

- Printing information
- Notification recipients
- Parameters
- Language
- Submission time and Completion time
- Schedule
- Parent Request - if the request had a parent click this button to view details information about this request

Use the buttons to perform the following:

- View Diagnostics for the request.
- Launch the Request Log in a separate browser window.
- Launch the Manager Log in a separate browser window.
- View the Request Output.

Inactive Requests

The list of inactive requests is shown with the following information:

- Request ID
- Short Name
- Program Name
- Status - possible values are Disabled, On Hold, or No Manager.
- Requestor - drills down to the User Details page.
- Priority - The priority of the concurrent program to be run. A concurrent program may be given a priority when it is initially defined. However, you can assign a new priority to a request here by typing in the new value and clicking the Apply button.
- Requested Start

Also, you can click on "Show" under the Details column to see additional details for a request, such as

- Printing information

- Notification recipients
- Parameters
- Language
- Submission time
- Schedule

Use the Remove Hold button to remove a hold on the inactive request.

Use the buttons to perform the following:

- View Diagnostics for the request.
- View Managers for the request.
- Cancel the request.

Pending Requests

Choose either Table View or Chart View. The Chart View displays a graph of the completed requests by Status.

The Table View displays the following fields:

- Request ID
- Short Name
- Program Name
- Status - possible values are Normal, Standby, Scheduled, and Waiting.
- Requestor - drills down to the User Details page.
- Priority - The priority of the concurrent program to be run. A concurrent program may be given a priority when it is initially defined. However, you can assign a new priority to a request here by typing in the new value and clicking the Apply button.
- Wait Time - the amount of time after the Requested Start time that the program has been waiting to run.
- Requested Start

Also, you can click on "Show" under the Details column to see additional details for a request, such as

- Printing information

- Notification recipients
- Parameters
- Language
- Submission time
- Schedule

Use the buttons to perform the following:

- View Diagnostics for the request.
- View Managers for the request.
- Place the request on Hold.
- Cancel the request.

Running Requests

Choose either Table View or Chart View. The Chart View displays a graph of the completed requests by Status.

The Table View displays the following fields:

- Request ID
- AUDSID - The database session ID for the request. Drills down to the Database Session Information page.
- Short Name
- Program Name
- Requestor - drills down to the User Details page.
- Responsibility
- Duration

Also, you can click on "Show" under the Details column to see additional details for a request, such as

- Printing information
- Notification recipients
- Parameters

- Language
- Submission time
- Schedule

Use the buttons to perform the following:

- View Diagnostics for the request.
- View the Internal Manager Environment for the request.
- View the Request Log.
- View the Manager Log.
- Cancel the request.

Concurrent Request Diagnostics

For completed, inactive, pending, and running requests, the following information is shown:

Request Status

- Phase - the phase may be Pending, Running, Completed, or Inactive
- Status
  - If the phase is Pending, the status may be: Normal, Standby, Scheduled, or Waiting.
  - If the phase is Running, the status may be: Normal, Paused, Resuming, or Terminating.
  - If the phase is Completed, the status may be: Normal, Error, Warning, Cancelled, or Terminated.
  - If the phase is Inactive, the status may be: Disabled, On Hold, or No Manager.
- Request ID
- Diagnostics
  - For completed requests - provides a completion message and reports the begin and end times for the request.
  - For inactive requests - reports the date and time that the request became inactive and the reason for this status. Provides options based on the status.

- For pending requests - reports the reason for the status of the request and options available to the system administrator.

Run Times

This portion of the screen shows run time statistics for running, completed, and pending requests. All times are displayed in seconds.

- Average - the average time required to run this request.
- Minimum - the minimum time reported for the completion of this request.
- Maximum - the maximum time reported for the completion of this request.
- Estimated Completion - (displayed for running requests only) based on the statistics recorded for this request, the estimated time that the request will finish. If you need to shut down the system, use this indicator as a guide.
- Actual - (displayed for completed requests only) the actual time required for this request to run.

Waiting on Following Requests

This region of the page displays requests that are incompatible with the selected pending, running, or inactive request. Shown for each request are the following fields:

- Show Details - click this link to drill down to request details.
- Request ID
- Program
- Phase
- Status
- Requestor - click this link to drill down to the User Details page.
- Reason - the reason the selected request is waiting on this request.

You can perform the following actions on the requests listed:

- Hold - place the request on hold to allow the selected request to run.
- Cancel - cancel the request to allow the selected request to run.
- View - view the request details.

Internal Manager Environment

This page shows the environment variables and their values for the ICM environment.

You can search for a particular variable using the filter.

Multilingual Support for Concurrent Requests

Users can submit a single concurrent request for a single concurrent program to be run multiple times, each time in a different language. Any output that is produced can be routed to different printers based on language. Users can also route completion notifications based on the language of the output.

For example, a user could submit a request for a Print Invoices program that would cause that program to run several times, each time in a different language, with each set of invoices printed on a different printer.

When submitting requests with multilingual support (MLS), separate requests are actually submitted; one request for each language. To distinguish these requests in the UI, such as the Monitor Requests page, the request names are prefixed with "<ISO language code>-<territory>".

Request Submission

A concurrent program can have a Multilingual Support (MLS) function associated with it. This function determines the set of languages over which the concurrent program will run. For example, the developer might associate a function with a Print Invoices program that would cause any request for that program to run in the preferred languages of the customers who have pending invoices.

If the concurrent program does not have an MLS function associated with it, then a user can choose when submitting the request the list of languages in which the program should run. The language of the current session is the default language.

If a concurrent program does have an MLS function associated with it, users will not be able to select languages for their requests. The associated MLS function determines the languages in which the request will run.

> **Note:** A concurrent program with an associated MLS function should have the "Use in SRS" box checked. If the "Use in SRS" box is not checked then the MLS function will be ignored. See: Concurrent Programs Window, page 4-71.

Runtime Behavior

Multilingual requests behave similarly to request sets. A user submits a single request. When that request runs, it submits a child request for each language in its list of languages. The parent request remains in the Running/Waiting state until its child requests are completed. If any child request completes with error status, then the parent request completes with error status. If no children complete with error status, but one or more completes with warning status, then the parent completes with warning status.

Finally, if all children complete with normal status, then the parent completes with normal status.

MLS Functions

Developers can create an MLS function for concurrent programs. The MLS function determines in which of the installed languages a request should run. For example, an MLS function for a Print Invoices program could require that any request for that program to run only in the preferred languages of the customers who have pending invoices. This restriction saves system resources by assuring that the request does not run in languages for which no output will be produced. This restriction also prevents user error by automatically selecting the appropriate languages for a request.

MLS functions are PL/SQL stored procedures, written to a specific API. When the concurrent manager processes a multilingual request for a concurrent program with an associated MLS function, it calls the MLS function to retrieve a list of languages and submits the appropriate child requests for each language. The concurrent program application short name, the concurrent program short name, and the concurrent request parameters are all available to the MLS function to determine the list of languages that the request should be run in.

Beginning with Release 12.1, MLS functions can also support multiple territories and numeric character settings (",." for example).

MLS functions are registered in the Concurrent Program Executable form. A registered MLS function can be assigned to one or more concurrent programs in the Concurrent Programs form.

Related Topics

Oracle E-Business Suite User's Guide

Oracle E-Business Suite Concepts Guide

Oracle E-Business Suite Developer's Guide

Multiple Organizations Reporting

The Oracle E-Business Suite organization model dictates how transactions flow through different organizations and how those organizations interact with each other. You can define multiple organizations and the relationships among them in a single installation of Oracle E-Business Suite. These organizations can be sets of books, business groups, legal entities, operating units, or inventory organizations.

Multiple organizations reporting improve reporting capabilities of Oracle E-Business Suite products by allowing reporting across operating units.

Understanding Operating Units

An operating unit is an organization that uses *Oracle Cash Management*, *Order Management and Shipping Execution*, *Oracle Payables*, *Oracle Purchasing*, and *Oracle Receivables*. An operating unit is associated with a legal entity and may be a sales office, a division, or a department. Information is secured by operating unit for these applications and each user sees information only for their operating unit. To run any of these applications, you choose a responsibility associated with an organization classified as an operating unit.

> **Note:** The profile option *MO:Operating Unit* links an operating unit to a responsibility. You must set this profile option for each responsibility.

Running Reports

To run reports using multiple organizations reporting:

1. Navigate to the **Submit Requests** page.
2. Choose the report that you want to run.

   A list of available operating units displays.
3. Choose the operating unit for this report.
4. Continue scheduling and submitting the request as usual.

Related Topics

Multiple Organizations in Oracle E-Business Suite

The Output Post Processor

Concurrent processing uses the Output Post Processor (OPP) to enforce post-processing actions for concurrent requests. Post-processing actions are actions taken on concurrent request output. An example of a post-processing action is that used in publishing concurrent requests with XML Publisher. For example, say a request is submitted with an XML Publisher template specified as a layout for the concurrent request output. After the concurrent manager finishes running the concurrent program, it will contact the OPP to apply the XML Publisher template and create the final output.

The OPP runs as a service that can be managed through Oracle Applications Manager. One service instance of the OPP service is seeded by default. This seeded OPP service instance has one work shift with one process.

A concurrent manager contacts an available OPP process when a running concurrent

request needs an OPP processing action. A concurrent manager uses a local OPP process (that, is, on the same node) by default, but will choose a remote OPP if no local OPP process is available.

There should always be at least one OPP process active in the system. If no OPP service is available, completed requests that require OPP processing will complete with a status of Warning.

An OPP service is multi-threaded and will start a new thread for each concurrent request it processes. You can control the number of simultaneous threads for an OPP service instance by adjusting the Threads per Process parameter for the instance. If all the OPP services have reached their respective maximum number of threads, the requests waiting to be processed remain in a queue to be processed as soon as threads become available. If request throughput has become slow, you may want to increase the number of threads per process for the OPP. It is recommended that you keep the number of threads per process between 1 and 20.

Delivery Options for Concurrent Request Output

Oracle XML Publisher offers a feature called Delivery Manager which delivers documents through e-mail, fax, and other delivery channels. This capability is present for single requests and request sets using the Forms-based request submission UI, and for single requests in the HTML-based request submission UI.

Users can direct the output of their concurrent requests to any of the channels that Delivery Manager supports. A user can use the Preferences Page to set the email server user and email server password, so that Delivery Manager can connect to the email server on the user's behalf to deliver the reports through email.

For more information on Delivery Manager and related setup in Oracle XML Publisher, see the Oracle E-Business Suite online help.

> **Note:** Delivery options using the HTML-based request submission UI is not supported currently for request sets.

In the Forms-based Standard Request Submission (SRS) window, users can enter their delivery options in a Delivery Options window available from a button in the "Upon Completion ... region in the SRS window. In the HTML-based request submission UI, delivery options in the Delivery step.

The following delivery channels are possible:

- IPP (Internet Printing Protocol) Printer
- E-mail
- Fax
- FTP

- WebDAV (Web-based Distributed Authoring and Versioning)
- Custom (defined using the Create Delivery Option HTML page)

Request Submission

At the time of request submission, a user can enter details for the chosen delivery option(s) as described below.

IPP Printer

- Username/Password - A user can enter a username and password for the selected IPP printer, if required. These will override any default values entered by the system administrator when the IPP printer was registered.
- Copies - The number of copies. This must be greater than 0.
- Orientation - Portrait or Landscape.
- Language - Users can select a specific language or "All languages".

E-mail

The email delivery option requires that an Simple Mail Transfer Protocol (SMTP) host and port be defined in the profiles FND: SMTP Host (FND_SMTP_HOST) and FND: SMTP Port (FND_SMTP_PORT), respectively. The profile values for these can be viewed and updated on the site and user level by a System Administrator, and can be viewed and updated by users themselves.

- From - The user's default e-mail ID.
- Subject - Populated with a default value composed of the Oracle E-Business Suite instance, program name, and name of the user submitting the request.

To add a recipient for the e-mail, the user must click the **Add Another Row** button and add recipients.

- "To" Recipients - Required. Comma-separated e-mail IDs are supported.
- "Cc" Recipients - Optional. Comma-separated e-mail IDs are supported.
- For Language - The language for the report. If different languages are desired, additional rows can be used.

Fax

Printers that support faxing must be registered. See: Managing Delivery Options, page 4-223.

The Fax option here will list only those IPP printers that support faxing.

- IPP Printer/Fax Server - Required.
- Username/Password - A user can enter a username and password for the selected IPP printer, if required. These will override any default values entered by the system administrator when the IPP printer was registered.
- Fax Number
- For Language - The language for the report.

If different fax servers or languages are desired, additional rows can be used.

FTP

Both FTP and SFTP are supported. Secure FTP is indicated by checking the box "Secure FTP". Only password-authenticated SFTP is supported.

- Host Name - Required.
- Port - The default port value is 22.
- User Name - Required.
- Password - Required.
- Remote Directory - Required. If this is left blank, the file is transferred to the remote home directory.
- For Language - The language for the report.
- Secure FTP

Additional rows can be used for sending output to additional servers.

WebDAV

Enter the following for WebDAV.

- Host Name - Required.
- Port - The default port value is 22.
- Encoding - Optionally choose 'SSL'.
- Remote Directory - Required. If this is left blank, the file is transferred to the remote home directory.
- For Language - The language for the report.

Additional rows can be used for sending output to additional servers.

Custom

This option can be used if a custom delivery type has been registered in the Create Delivery Option page.

Enter the following:

- Custom Delivery Command - Use the LOV to specify the custom delivery option command.
- For Language - The language for the report.

Additional rows can be used to specify more custom delivery commands.

Managing Delivery Options

Use the Manage Delivery Options page available under the System Administration responsibility to search for, register, update, or delete these options.

To search for a delivery option:

You can search by delivery type.

You can search by the delivery name given by the user at the time of the delivery option creation.

To create (register) a delivery option of type IPP:

1. Enter a delivery name.
2. Enter a delivery type. Choose 'IPP Printer'.
3. Enter a host name an port. (Required)
4. Enter a username and password.

   A system administrator can enter a default username and password which can be overridden by a user during request submission.
5. Enter a printer name. (Required)
6. Enter the number for sided printing.

   The lookup codes in the following table are used:

Lookup Code	Meaning
1	One-sided printing
2	Two-sided printing

7. For the options Authentication, Encryption, Use Full URL, and Use Chunked Body, check the boxes as needed. These features are documented in the Delivery Manager documentation in the Oracle XML Publisher online help.

8. If the IPP Printer supports faxing, then the Support Fax box should be checked. This option will be used to enable the LOV for the fax server in the SRS Fax tab.

To create (register) a delivery option of type Custom:

1. Enter a delivery name.

2. Enter a delivery type. Choose 'Custom'.

3. Provide a custom command, using the syntax {file} for the output filename.

To update a delivery option:

You can update a delivery option definition except for the delivery name and delivery type. These fields are read-only.

You can select a delivery option to update from the search results table in the Search page.

To delete a delivery option:

You can delete a delivery option by selecting the Delete icon for it in the Search page. A confirmation message will be shown before deletion.

Reviewing Requests, Request Log Files, and Report Output Files

This essay explains how you, as System Administrator, can view and change the status of concurrent requests, and how to view request log and report output files.

How To View Request Status and Output

Use any of the following methods to view the status and output of concurrent requests.

Use the Requests Window

Use the Requests window to view the status of concurrent requests, and to view request log and report output files.

The System Administrator and Oracle Alert Manager have a privileged version of the Requests window that provides you with more capabilities than your end users. For example, using the Requests window, you can view the status of and log files for *all* concurrent requests (not just your own), including requests that completed unsuccessfully. On some platforms, you can even view the log files of running requests.

Using the same window, you can view your own report output online. You cannot, however, view report output from other users' requests.

From the Requests window, you can also:

- place and remove holds from any pending or inactive request
- cancel a pending request, or terminate a running request
- restart a request set.
- change the priority of any pending request
- view the manager log file
- determine where *any* pending request stands in the queue for each manager defined to accept the request
- determine when the concurrent manager is inactive and needs to be restarted.

Run the Completed Concurrent Requests Report

You can run a report that lists parameters and any error messages associated with concurrent requests that have completed running. See: Completed Concurrent Requests Report, page 4-114.

Setting End User Report and Log File Access Privileges

The user profile option *Concurrent:Report Access Level* determines report output file and log file access privileges for your end users. As System Administrator, you can set this profile option to either "User" or "Responsibility."

All users can can review the log and report output files from requests that they submitted.

If you set the *Concurrent:Report Access Level* option to "Responsibility" at the User level, that user can also review the log and report output files from all requests submitted from the current responsibility.

If you set the *Concurrent:Report Access Level* option to "Responsibility" at the Responsibility level, any user of that responsibility can also view the log and report output files from all requests submitted by any other user of that responsibility.

Defining the Reports Viewer

The Oracle E-Business Suite Report File Viewer is used by default for viewing your text report files. You can also display text files in a browser or use another application such as Microsoft Word. You define your default viewer by setting a profile option.

Set the Viewer:Text Profile Option

If the Viewer:Text profile option is set to "Browser" then reports are sent to a web browser. If this profile option is left blank, the Report File Viewer is used instead.

If this profile option is left blank, a report or log file can still be viewed in a browser by first viewing it using the Report File Viewer, and then choosing "Copy File..." from the Tools menu.

Viewing HTML Report Output

You can view your reports with HTML output in a browser. Once an HTML report has been sent to a browser, it can be saved to the desktop by using the Save As functionality of the browser.

> **Note:** HTML reports are displayed by the browser in the character set of the server. This character set may or may not match the character set on the client. Therefore, it may be necessary to convert the output to the client character set when saving the report. If the browser supports character set conversion with Save As, there will be a poplist in the Save As dialog box. The user can then choose an encoding which matches the client character set.

Online Report Review using Other Applications

You can set up your Online Report Review implementation to enable viewing output files in other applications, such as Microsoft Word or Excel. To do this you associate MIME types with file output formats.

Users can then set their preferred MIME types for particular output formats using profile options, or the users may be prompted to choose the appropriate MIME type for a file at runtime.

You can register more than one MIME type file format with each output format. In the Viewer Options window, you enter in the file format, the MIME type, whether you want to utilize the value of the FND: Native Client Encoding profile option, and a description. The description is displayed to the user in the Profile Values window and

the Submit Request form.

If the Allow Native Client Encoding box for the associated MIME type has been checked in the Viewer Options window, the Report Viewer will convert the output file into the character set specified by the profile option FND: Native Client Encoding.

When the report is viewed, it is first sent to a browser. The browser then uses the associated MIME type to display the report.

> **Important:** For printing, if users choose either HTML or PDF as the output type with Oracle Report programs, they must use appropriate printer drivers to handle the PDF and HTML file for printing their output. See: Overview of Printers and Printing, page 6-1.

> **Note:** For PDF files, the Adobe Acrobat Reader application must have options set as described below:
>
> - For Acrobat 4, under File > Preferences > General: uncheck Web browser integration.
> - For Acrobat 5 and 5.1, under Edit > Preferences: under options, unselect "Display PDF in Browser".
> - For Acrobat 6 and higher, under Edit > Preferences > Internet: under Web browser options, unselect "Display PDF in browser".

See: Viewer Options Window, page 4-173

Changing the Status of Concurrent Requests and Request Sets

This essay explains how to change a request's phase and status, and how to change the priority of a Pending or Inactive request. It also discusses how to restart request sets and how to prioritize requests by placing request sets on hold.

Changing a Request's Phase and Status

A request is in one of four phases: Pending (waiting to be run), Running, Completed, or Inactive (unable to run). Within each phase, a request's condition is referred to as its status.

You can change the phase of a Pending, Running, or Inactive request by changing its status.

Pending and Inactive Requests

You may cancel Pending and Inactive requests. The request's phase and status becomes *Completed - Cancelled*.

You may place on hold Pending and Inactive requests. The request's phase and status becomes *Inactive - On Hold*. You can reverse this action by later selecting the request removing the hold.

Running Requests

You can terminate Running requests. The request's phase and status becomes *Completed - Terminated*.

Changing a Request's Status

You can change the status of a request, and its resulting phase, using the Requests window.

Changing the Priority of a Pending or Inactive request

Requests normally run according to start time, on "first-submitted, first-run" basis. However, a higher priority request starts before an earlier request.

As System Administrator, you can change the priority of any Pending or Inactive request using the Requests window.

Request Priority is associated with an application User

The priority of a user's requests defaults to the value you, as System Administrator, set for their *Concurrent:Priority* user profile option. Users cannot change the priority of their requests.

If a concurrent program has a defined priority, that priority overrides the user's profile option.

- Priorities range from 1 (highest) to 99 (lowest).
- The standard default is 50.
- Concurrent programs submitted by the Internal Concurrent Manager have a priority of zero (0), and override all other requests.

> **Tip:** If you need to change the priority of a request frequently, you should consider assigning that concurrent program its own priority.

Related Topics

Overview of Concurrent Processing, page 4-1

Life cycle of a concurrent request, page 4-2

Concurrent Processing User Profile Settings, page 4-231

Managing Request Sets

This section discusses how to restart request sets and how to yield a request set to higher priority requests.

Restarting Request Sets

If a request set completes with a status of *Error*, the **Restart** button, on the Oracle Applications Manager - View Completed Requests page is enabled. The system also automatically captures, records, and saves the information of the first stage that fails so that when the user clicks on the **Restart** button the request set can restart from that point.

Once the stage has been identified, the request set program submits the stage program in resubmit mode. In this mode, the program looks at the same stage from the previous run and determines which programs need to be rerun, (only those that ended in error), and runs those programs. If this stage completes successfully or has a *Warning* status, the system proceeds to the next stage using the normal mechanism of restarting the request set program.

> **Note:** Users may restart a request set multiple times. The logs for each stage and individual programs are maintained independent of the number of runs as each stage and program submission generates a new request. However, the logs and associated files for a request set are rewritten each time the set is restarted.

Holding Request Sets

In some circumstances, such as when a request set has a large number of stages and takes a long time to execute, administrators may want to yield a request set to higher priority requests. By utilizing the *Hold Request Set* feature, users can place a running request set on hold and effectively control the execution of request set stages.

The **Hold** and **Remove Hold** buttons are available on the OAM View Running Requests page. To hold a request set, simply select the request set and click the **Hold** button. Click **Remove Hold** when you want the request set to continue executing.

Managing Concurrent Processing Files and Tables

This section explains how to maintain the number of log and output files the operating system retains, and how to manage Application Object Library database tables that store information about concurrent requests and concurrent manager processes.

The database tables that are affected by running the Purge Concurrent Request and/or Manager Data program are:

FND_CONCURRENT_REQUESTS

This table contains a complete history of all concurrent requests.

FND_RUN_REQUESTS

When a user submits a report set, this table stores information about the reports in the report set and the parameter values for each report.

FND_CONC_REQUEST_ARGUMENTS

This table records arguments passed by the concurrent manager to each program it starts running.

FND_DUAL

This table records when requests do not update database tables.

FND_CONCURRENT_PROCESSES

This table records information about Oracle E-Business Suite and operating system processes.

FND_CONC_STAT_LIST

This table collects runtime performance statistics for concurrent requests.

Maintenance Suggestions

Your MIS department and application users should agree on an archiving and file retention policy that is appropriate for your organization. To avoid running out of space on your disk drives, you should periodically delete Oracle E-Business Suite log files and output files.

> **Tip:** You can run the program "Purge Concurrent Request and/or Manager Data" once and automatically resubmit the program for you at specific time intervals.

There are some sample guidelines for when to run the Purge Concurrent Requests and/or Manager Data program. Adopt these guidelines according to your user community's usage of Oracle E-Business Suite.

- every 30 days for normal usage
- every two weeks (14 days) for heavy usage

- if using the AGE mode, set the Mode Value to 5 to retain the five most recent days of concurrent request data, log files, and report output files.

Purging removes Audit data

When you purge concurrent request information, you lose audit details. The Signon Audit Concurrent Requests report uses this audit information.

Concurrent Processing User Profile Settings

This essay explains the user profile option settings relevant to submitting concurrent requests.

Setting Concurrent Processing Options

End users can control certain runtime options for their concurrent requests. For example, you can choose a specific date on which to start a request.

If a user does not explicitly enter these options at the time of the request, concurrent processing options default to their user profile values.

As System Administrator, you set user profile values for your end users with the System Profile Values window. Both you and your end users can set some of your own profile values using the Personal Profile Values form.

Changing Concurrent Processing Options for submitted requests

You or your users can use the Requests window to change the concurrent processing options for a submitted request up until the time it starts running.

- As System Administrator you can change all concurrent options for any request.
- Your users can change most of their request's concurrent options.

  End users cannot change (nor set) the priority of their request, or the report access level for viewing request log files and report output files online.

The following table lists the concurrent processing user profile options and an explanation of each:

User Profile Option	Explanation
Concurrent: Hold Requests	"Yes" places concurrent requests on hold. "No" starts programs according to the request's priority and start time.

User Profile Option	Explanation
Concurrent: Multiple Time Zones	"Yes" ensures that requests are scheduled immediately regardless of the time zone your client is running in.
Concurrent: Report Access Level	Viewing a request's output/log files online and reprinting reports can be accessed according to: "Responsibility" - by anyone using the responsibility that submitted the request "User ″ - by only the user who submitted the request.
Concurrent: Report Copies	The number of output copies that print for each report.
Concurrent: Request Priority	Requests normally run according to start time, on a "first-submitted, first-run" basis. Priority overrides request start time. A higher priority request starts before an earlier request. Priorities range from 1 (highest) to 99 (lowest). The standard default is 50.
Concurrent: Request Start Time	The date and time requests are available to start running. If the start date and time is at or before the current date and time, requests may be run immediately.
Concurrent: Save Output	"Yes" saves concurrent program outputs in a standard file format. Some concurrent programs do not generate an output file.
Concurrent: Sequential Requests	"Yes" forces requests to run one at a time (sequentially) according to the requests' start dates and times. "No" means requests canrun concurrently when their concurrent programs are compatible.
Concurrent: Wait for Available TM	You can specify the maximum number of seconds that the client will wait for a given transaction manager (TM) to become available before moving on to try a different TM.
Concurrent: URL Lifetime	This profile option determines the length of time in minutes a URL for a request ouput is retained before it is deleted from the system.

User Profile Option	Explanation
Printer	The printer which prints your reports.

Updating Concurrent Request Profile Options

Most concurrentuser profile options may be set by the System Administrator at all four levels: site, application, responsibility, and user. The user profile *Concurrent:Report Access Level* may not be set at the application level.

Your users can change the default values for most of the concurrent processing profile options. However, they cannot set Concurrent: Request Priority, or Concurrent: Report Access Level.

Related Topics

Overview of Concurrent Processing, page 4-1

Purge Concurrent Request and/or Manager Data Program

Use this program to delete:

- request log files, concurrent manager log files, and report output files from your product directories maintained by the operating system
- records (rows) from Application Object Library database tables that contain history information about concurrent requests and concurrent manager processes.

Report Options

Report options are:

Entity

All Purges records from database tables that record history information for concurrent requests, history information for concurrent managers, and purges request log files, manager log files, and report output files from the operating system.

Manager Purges records from database tables that record history information for concurrent managers, and purges manager log files from the operating system.

Request Purges records from database tables that record history information for concurrent requests, and purges request log files and report output files from the operating system.

Mode

Age Enter the number of days for which you want to save concurrent request history, log files, and report output files. The purge program deletes all records older (in days) than the number you enter.

For example, if you enter "5", then all concurrent request history, log files, and report output files older than five days is purged.

Count Enter the number of (most recent) records for which you want to save concurrent request history, log file, and report output files. The purge program starts from the most recent records, retains the number you enter, and purges all remaining records.

For example, if you enter "5", then the five most recent concurrent request history records, request log files, manager log files, report output files are saved, and all remaining records are purged.

Mode Value

Enter a value to define the number of days for Mode=Age or the number of records for Mode=Count. The valid values are 1 - 9999999.

Oracle ID

Enter the Oracle ID that concurrent programs connect to for which you want to purge concurrent request records, and associated log files and report output files. Oracle ID has relevance when the Entity is either "Request" or "All".

For example, if you enter AP1, then the program purges all request records, log files, and report output files associated with requests to run programs that connect to the AP1 Oracle ID.

User Name

Enter the application username whose concurrent request records and associated log files and report output files you wish to purge. Username has relevance when the Entity is either "Request" or "All".

For example, if you enter JSMITH, then the program purges all request records, log files, and report output files associated with requests submitted by user JSMITH.

Select the application associated with the responsibility for which you want to purge concurrent request records, and associated log files and report output files. Responsibility Application is used with the *Responsibility* option, and has relevance when the Entity is either "Request" or "All".

Responsibility

Select the responsibility for which you want to purge concurrent request records, and associated log files and report output files. Responsibility has relevance when the Entity is either "Request" or "All".

For example, if you select the System Administrator responsibility, then the program purges all request records, log files, and report output files associated with requests submitted by users operating under the System Administrator responsibility.

Program Application

Select the application for which you want to purge concurrent request records, and associated log files and report output files. Program Application has relevance when the Entity is either "Request" or "All".

For example, if you select Oracle Payables, then the program purges all request records, log files, and report output files associated with requests to run Oracle Payables programs.

Program

Select the program for which you want to purge concurrent request records, and associated log files and report output files. Program has relevance when the Entity is either "Request" or "All".

For example, if you select Program X, then the purge program purges all request records, log files, and report output files associated with requests to run Program X.

Manager Application

Select the application associated with the concurrent manager for which you want to purge concurrent request records, and associated log files and report output files.

Manager Application is used with the *Manager* option, and has different effects when Entity is set to "Request, and when Entity is set to "Manager" or "All".

- When Entity is set to "Request", the program purges all request records, log files, and report output files associated with requests run by the concurrent manager named in the *Manager* option.

- When Entity is set to either "Manager" or "All", in addition to the above, the program also purges all manager log files associated with the concurrent manager named in the *Manager* option.

Manager

Select the concurrent manager for which you want to purge concurrent request records, and associated log files and report output files.

Manager is used with the *Manager Application* option, and has different effects when Entity is set to "Request," and when Entity is set to "Manager" or "All".

- When Entity is set to "Request", the program purges all request records, log files, and report output files associated with requests run by the concurrent manager named in the *Manager* option.
- When Entity is set to either "Manager" or "All", in addition to the above, the program also purges all manager log files associated with the concurrent manager named in the *Manager* option.

Report

Select whether you want a report listing the number of records purged by the Purge Concurrent Request and/or Manager Data program.

No Run the program but do not generate a report.

Yes Run the program and generate a report.

Purge Other

Select whether you want to delete records from the FND_DUAL table.

No .Do not delete records from FND_DUAL.

Yes Delete records from FND_DUAL.

Related Topics

Overview of Concurrent Processing, page 4-1

Life cycle of a concurrent request, page 4-2

Reviewing Requests, Request Log Files, and Report Output Files, page 4-224

5

User Profiles and Profile Options in Oracle Application Object Library

Overview of Setting User Profiles

A profile is a set of changeable options that affect the way your application looks and behaves. As System Administrator, you control how Oracle E-Business Suite applications operate by setting user profile options to the values you want. You can set user profile options at different levels: site, application, responsibility, user, server, and organization, depending on how the profile options are defined.

See: Defining Preferences with User Profile Options, *Oracle E-Business Suite User's Guide*.

Profile Hierarchy

A profile option can be set at one or more levels, depending on its hierarchy type. Most profile options use the *Security* hierarchy type, meaning that they can potentially be set at the four levels: Site (lowest level) , Application, Responsibility, and User (highest level).

Note: A higher-level option value overrides a lower-level value.

Hierarchy Types

Hierarchy types enable system administrators to group and set profile options according to their business needs or the needs of the installation.

There are several hierarchy types: Security, Organization, Server, and Server+Responsibility.

Security

Security is the default hierarchy type. Profiles that use this hierarchy type follow the

hierarchy: Site - Application - Responsibility - User.

> **Note:** Most profile options that existed before hierarchy type was introduced use Security.

Organization

Organization refers to operating unit. For example, clerks in different organizations may need to have different values for a given profile option, depending on their organization, but clerks in the same organization would use the same value. The Organization hierarchy type allows system administrators to set a profile option at the organization level, so that all users within that organization will use the profile option value set once at the organization level. Profiles using this hierarchy type follow the hierarchy Site - Organization - User.

Server

The Server hierarchy type is used when the system needs to determine the middle-tier server on which the user's session is running. For example, the profile "Applications Web Agent" can be defined using the Server type. The setting of this profile option can differ for an internal server versus an external one. Cookie validation, can then be done against the value of this profile option. Profiles using this hierarchy type follow the hierarchy Site - Server - User.

Server+Responsibility

The Server+Responsibility hierarchy type allows you to set distinct profile values for specific combinations of server and responsibility. When evaluating profile values to use, the value found with the most specific match across all levels is chosen. At any level, a special default value can be chosen in case no other specific match at that level is found.

Either or both of the responsibility or server may have specific values, or may be the default value. For purposes of evaluating default matches, the server is considered to be at a lower level and less specific than the responsibility.

When evaluating profile values at this Server+Responsibility level, the system first looks for a specific match for both the responsibility and server level values. If no such match is found, it looks for a row matching responsibility and default for the server level. If no such match is found, it will next look for a row matching the server with default for the responsibility level. If no such match is found, it will continue up the hierarchy to the Site level.

The following table describes how the values of a profile using this hierarchy could be set up ("-" indicates default):

Level	Responsibility	Server	Profile Value
Site			A
Server+Responsibility	System Administrator	External	B
Server+Responsibility	-	External	C
Server+Responsibility	-	Internal	D
Server+Responsibility	System Administrator	-	E
Server+Responsibility	General Ledger Superuser	-	F
User	Joe Smith	-	G

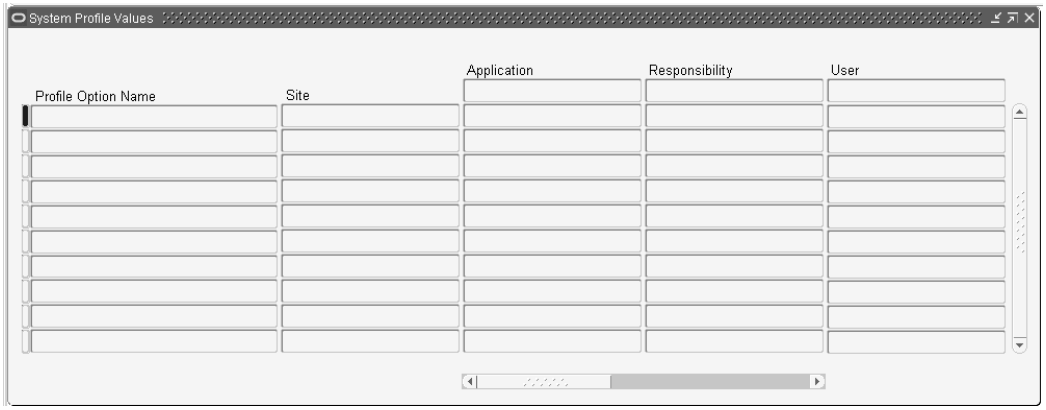
The following table lists the values of the profile that would be used in the given contexts:

Server	Responsibility	User	Profile Value	Explanation
External	System Administrator	Joe Smith	G	User matches.
External	System Administrator	Yali Xu	B	Responsibility plus Server match.
External	Human Resources Manager	Yali Xu	C	Server matches.
Internal	System Administrator	Yali Xu	E	Responsibility matches.
Custom	Human Resources Manager	Yali Xu	A	No match. Use Site level value.

Setting Profile Options

As System Administrator, you can use the Define Profile Values window to set profile options for your user community. If you change a user profile option value, your change takes effect as soon as your users log on again or changes responsibilities.

> **Note:** Profile option values are cached. Setting or unsetting a profile option value raises a cache invalidation business event. When this event is processed, the middle-tier profile cache is invalidated. If this business event takes an unusual amount of time to process, the invalidation might not occur as expected. In this case, the profile option cache can be cleared manually by navigating to Functional Administrator (seeded responsibility) > Core Services > Caching Framework, selecting the cache object 'PROFILE_OPTION_VALUE_CACHE' , and clicking the Clear Cache button.



You can also view how site-level profile options are set using Oracle Applications Manager (Site Map > Monitoring tab).

When you set a user profile, you provide Oracle E-Business Suite with standard information (such as printer) that describes a user, responsibility, application, or site. You can set values for user profile options at each profile level.

For the Security, Organization, and Server hierarchy types, the following describes how option settings are used:

Level	Option Settings
Site	All users at an installation site.

Application	All users of any responsibility associated with the application.
Responsibility	All users currently signed on under the responsibility.
User	An individual user, identified by their application username.
Server	An individual server.
Organization	A particular organization.

The values you set at each level provide runtime values for each user's profile options. An option's runtime value is the highest-level setting for that option.

When a profile option is set at more than one level, Site has the lowest priority, superseded by Application, then Responsibility, with User having the highest priority. A value entered at the Site level may be overridden by values entered at any other level. A value entered at the User level has the highest priority and overrides values entered at any other level.

Example

For example, assume the Printer option is set only at the Site and Responsibility levels. When the user logs on, the Printer option assumes the value set at the Responsibility level, since it is the highest-level setting for the option.

For the Server+Responsibility hierarchy type, option settings pertain to specific combinations of server and responsibility. The system first looks for a specific match for both server and responsibility values. If no such match is found, the system next looks for a profile option value matching responsibility, and with a default value for the server. If no such match is found, the system then looks for a profile option value matching the server, with default value for the responsibility. If no such match is found, the system will continue up the hierarchy to the Site level profile option value.

> **Tip:** As System Administrator, you should set site-level option values before specifying values at the other three levels after the installation of Oracle E-Business Suite. The option values specified at the site-level work as defaults until option values are specified at the other levels.

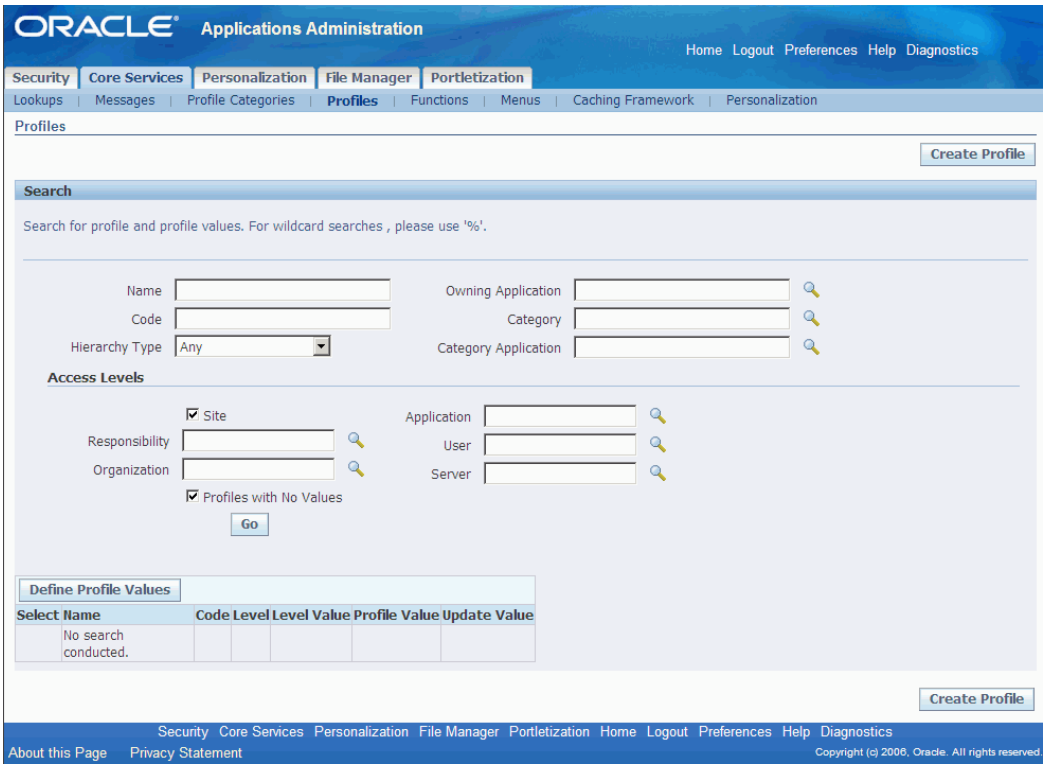
Oracle E-Business Suite users may use the Personal Profile Values window to set their own personal profile options at the user level. Not all profile options are visible to users, and some profile options, while visible, may not be updated by end users.

> **Note:** The display of NULL values has changed from that in previous

releases. In the Personal Profiles window, the default value column no longer displays a NULL value if a NULL value exists in the database. In the Examine window, NULL database values are not shown; the value set at a lower level is shown instead. If there are no non-NULL values set at a particular level, then a message appears stating that no values exist.

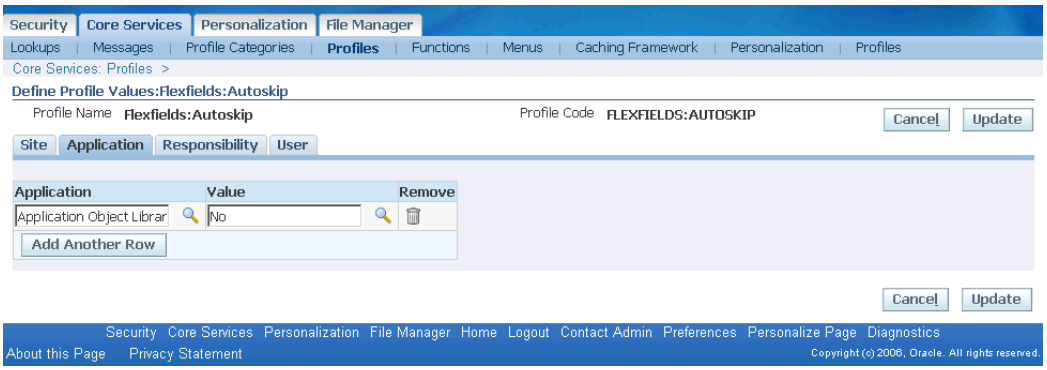
HTML-based Profile Pages

You can also use the HTML-based Profile pages to manage your profile values. These pages are available from the Functional Administrator responsibility under Core Services.



The **Site** and **Profiles with No Values** check boxes on the Profiles page are selected by default. Therefore, when a search is performed and a profile option is selected, the system lists the values defined only at site level. To see the values defined at all the levels, uncheck these checkboxes before performing a search and selecting a profile option.

To define a value for a profile option at some level, then select that profile option and click **Define Profile Values**. This will navigate you to the Define Profile Values page where you can add the value for all the relevant levels by navigating through the sub-tabs in that page.



Using Profile Options in Other Oracle E-Business Suite Features

Profile option settings may be used as a default value for a concurrent program's parameter or a flexfield's segment in the following forms:

- Concurrent Programs form, Parameters window, Parameter Detail region. See: Concurrent Programs Form, page 4-71.
- Request Set form, Report Parameters window. See: Defining Request Sets, page 4-11 .
- Key Flexfield Segments form, Segment window, Validation Information region. See: Defaulting Segment Values, *Oracle E-Business Suite Flexfields Guide*.
- Descriptive Flexfield Segments form, Segment window, Validation Information region. See: Defaulting Segment Values, *Oracle E-Business Suite Flexfields Guide*.

To use a profile option's setting as a default value, navigate to the form's Default Type field and select *Profile*. Then, enter the profile option's internal name in the Default Value field.

Profile options can also be used in value set definitions. See: Overview of Values and Value Sets, *Oracle E-Business Suite Flexfields Guide*.

Examples of User Profile Options

Example

Your Accounts Payable department recently purchased a printer, and you want all the reports from that department to print on the new printer. Simply change the Printer profile option for Oracle Payables to reflect the purchase of the new printer.

Tip: This example highlights the importance of default profile options.

If an application user of Oracle Payables or a responsibility associated with Oracle Payables already has a value specified for the printer profile option, that value will override the value you set at the application level. We suggest you first set user profile options at the site level, and then work your way up the hierarchy to other levels when appropriate.

Profile Categories

Profiles can be grouped into categories based on their functional area. Administrators can categorize profiles and then easily search on the profiles by category in the Profiles HTML-based page when they need to view or update them.

The Profiles and Profile Categories HTML-based pages can be accessed from the Functional Administrator and Functional Developer responsibilities. For more information on these, see: Overview of Functional Administrator and Functional Developer Responsibilities, *Oracle E-Business Suite Maintenance Guide*.

Profile Categories Search Page

From the Profiles Categories Search page, you can search based on the following criteria:

- Name
- Code (internal name)
- Application

The search results will list the profile categories that meet your criteria. You can click on a profile category name to view the profile options included in that category, and in turn, click on a profile option name to view and update its definition.

Creating a Profile Category

In creating a profile category, you specify a name, code, owning application, and description. You then add profiles to the category.

After creating a profile category, you can duplicate, update, or delete it.

Exportable Profiles for iSetup

Some profiles in the Applied Technology area are in a product-specific category called "Exportable" (internal name <application short name>_AZ_EXPORTABLE) for Oracle iSetup. To find out which profiles are in such a category, query for the profile category "Exportable" for the given application (for example, "Application Object Library").

User Profile Option Values Report

This report documents user profile option settings. Use this report when defining different profile option values for several responsibilities, or users, or for different applications.

Report Parameters

Profile Option Name

Choose the profile option name whose values you wish to report on. If you do not select a profile option name, then this report will document all profile options.

User Name

Choose the name of a user whose profile option values you wish to report on.

Application Short Name

Choose the name of an application whose profile option values you wish to report on.

Responsibility Name

Choose the name of a responsibility whose profile option values you wish to report on.

Report Headings

The report headings display the specified report parameters and provide you with general information about the contents of the report.

Profile Options in Oracle Application Object Library

This section lists profile options in Oracle Application Object Library. These profile options are organized according to their functional area and are available to every product in Oracle E-Business Suite. For each profile option, we give a brief overview of how Oracle Application Object Library uses the profile's setting.

Unless otherwise noted, a profile option uses the Security hierarchy type.

A table is provided for most profile options that lists the access levels for the profile option (at which levels the system administrator can set the profile option). For Security profile options, there are four possible levels at which system administrators can view and update a profile option value: site, application, responsibility, and user. This table lists whether the profile option's value is visible at each of these levels, and whether it is updatable at each level.

Note: For information on profile options related to Oracle Application Framework, see My Oracle Support Knowledge Document 1315510.1, *Oracle Application Framework Profile Options*.

ADF Integration

The following profile is used for linking Application Development Framework (ADF) 11*g* applications deployed on an Oracle Application Server 11*g* container from the Oracle E-Business Suite home page. The ADF application should be run on a different middle tier than the Oracle E-Business Suite.

External ADF Application URL

Use this profile to specify the ADF application base URL.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_EXTERNAL_ADF_URL.

Calendar Support

These profile options are used in supporting non-Gregorian calendars in Oracle Forms-based products.

FND: Calendar Week Start Day

With the Hijrah calendar, users can choose the first day of week in a Date Picker by setting this profile option.

This profile option is visible and updatable on all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_CALENDAR_WEEK_START_DAY.

FND: Forms User Calendar Profile Option

Users can set the FND: Forms User Calendar profile option to their preferred calendar. Valid values are: Arabic Hijrah. English Hijrah , Gregorian and Thai Buddha. By default, the user calendar displays the Gregorian calendar, but if this profile option is set to another value, that calendar is used.

Users can see and update this profile option.

This profile option is visible and updatable on all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_FORMS_USER_CALENDAR.

FND: Tooltip Calendar

In Oracle Forms-based applications, users can set the FND: Tooltip Calendar to a calendar other than the preferred calendar. Within the Date Picker, a given date will be displayed in this calendar's format as a tooltip.

Users can see and update this profile option.

This profile option is visible and updatable on all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_TOOLTIP_CALENDAR.

Concurrent Processing Execution

The internal name for this profile category is FND_CP_EXECUTION.

Concurrent:Active Request Limit

You can limit the number of requests that may be run simultaneously by each user. or for every user at a site. If you do not specify a limit, no limit is imposed.

Users cannot see or update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is CONC_REQUEST_LIMIT.

Concurrent:Attach URL

Setting this option to "Yes" causes a URL to be attached to request completion notifications. When a user submits a request, and specifies people to be notified in the Defining Completion Options region, everyone specified is sent a notification when the request completes. If this profile option is set to Yes, a URL is appended to the notification that enables them to view the request results online.

Only the System Administrator can update this profile option.

Users can see but not update this profile option.

This profile options is visible at all levels but can only updated at the Site level.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_ATTACH_URL.

Concurrent:Conflicts Domain

Specify a conflict domain for your data. A conflict domain identifies the data where two incompatible programs cannot run simultaneously.

Users can see but not update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_CD_ID.

Concurrent:Date Parameter Increment Option

Use this profile to control how date parameters are automatically incremented for concurrent requests. In the Standard Request Submission window, the user can specify if to run a request periodically. The user can then specify that the interval be based on the start date of the requests, or specify the interval using a unit of time and number of units.

If this profile is set to "Start Date" then the date parameters for a given request will be

incremented according to the difference between the requested start date of the request and the requested start date of the previous request. If this profile is set to "Resubmit" any date parameters are incremented according to the current request's date parameter and the amount of time represented by the number of units (RESUBMIT_INTERVAL) and the unit of time (RESUBMIT_INTERVAL_UNIT_CODE).

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	No
Responsibility	Yes	No
User	Yes	No

The internal name for this profile option is CONC_DATE_INCREMENT_OPTION.

Concurrent:Hold Requests

You can automatically place your concurrent requests on hold when you submit them.

The default is "No". The concurrent managers run your requests according to the priority and start time specified for each.

Changing this value does not affect requests you have already submitted.

"Yes" means your concurrent requests and reports are automatically placed on hold. To take requests off hold, you:

- Navigate to the Requests window to select a request
- Select the Request Control tabbed region
- Uncheck the Hold check box

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes

Level	Visible	Allow Update
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_HOLD.

Concurrent:Multiple Time Zones

"Yes" sets the default value to 'Sysdate-1' for the 'Schedules Start Date' used by request submissions. Sysdate-1 ensures that you request is scheduled immediately regardless of which time zone your client session is running in. You should use this profile option when the client's session is running in a different time zone than the concurrent manager's session.

Users cannot see nor change this profile option.

This profile option is visible at all four levels and updatable at the Site level.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	No	No

The internal name for this profile option is CONC_MULTI_TZ.

Concurrent:Print on Warning

Set this profile option to "Yes" if you want concurrent request output to be printed if the requests completes with a status of Warning.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_PRINT_WARNING.

Concurrent:Report Copies

You can set the number of output copies that print for each concurrent request. The default is set to 1.

- Changing this value does not affect requests that you have already submitted.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_COPIES.

Concurrent:Request Priority

This displays the default priority number for your concurrent requests. Only a system administrator can change your request priority.

Requests normally run according to start time, on a "first-submitted, first-run" basis. Priority overrides request start time. A higher priority request starts before an earlier request.

Priorities range from 1 (highest) to 99 (lowest). The standard default is 50.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_PRIORITY.

Concurrent:Save Output

The Concurrent: Save Output profile is used to determine whether the default behavior of certain concurrent programs should be to save or delete their output files. This only affects concurrent programs that were created in the character mode versions of Oracle E-Business Suite (formerly Oracle Applications) and that have a null value for "Save Output".

- "Yes" saves request outputs.
- Some concurrent requests do not generate an output file.
- If your request output is saved, you can reprint a request. This is useful when requests complete with an Error status, for example, the request runs successfully but a printer malfunctions.
- Changing this value does not affect requests you have already submitted.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes

Level	Visible	Allow Update
User	Yes	Yes

The internal name for this profile option is CONC_SAVE_OUTPUT.

Concurrent:Sequential Requests

You can force your requests to run one at a time (sequentially) according to the requests' start dates and times, *or* allow them to run concurrently, when their programs are compatible.

- Concurrent programs are incompatible if simultaneously accessing the same database tables incorrectly affects the values each program retrieves.
- When concurrent programs are defined as incompatible with one another, they cannot run at the same time.

"Yes" prevents your requests from running concurrently. Requests run sequentially in the order they are submitted.

"No" means your requests *can* run concurrently when their concurrent programs are compatible.

Changing this value does not affect requests you have already submitted.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_SINGLE_THREAD.

Concurrent:Wait for Available TM

You can specify the maximum number of seconds that the client will wait for a given transaction manager (TM) to become available before moving on to try a different TM.

Users can see and update this profile option.

This profile option is visible and updatable at the site and application levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	No	No
User	No	No

The internal name for this profile option is CONC_TOKEN_TIMEOUT.

Concurrent Processing File Server

The internal name for this profile category is FND_CP_FILE_SERVER.

RRA:Delete Temporary Files

When using a custom editor to view a concurrent output or log file, the Report Review Agent will make a temporary copy of the file on the client. Set this profile to "Yes" to automatically delete these files when the user exits Oracle E-Business Suite.

Only the System Administrator can update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FS_DELETE.

RRA:Enabled

Set this user profile to "Yes" to use the Report Review Agent to access files on

concurrent processing nodes.

Only the System Administrator can update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FS_ENABLED.

RRA: Service Prefix

Using this new profile option allows you to override the default service name prefix (FNDFS_) assigned to the Report Review Agent. By assigning a new prefix to the Report Review Agent you can avoid having multiple instances of the Applications share executables.

Valid values for this option must be nine characters or less and use only alphanumeric characters or the underscore. We recommend using the underscore character as the last character of your value as in the default value "FNDFS_".

Users cannot see or update this profile option.

This profile option is visible and updatable at the site level only.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	No	No

The internal name for this profile option is FS_SVC_PREFIX.

RRA:Maximum Transfer Size

Specify, in bytes, the maximum allowable size of files transferred by the Report Review Agent, including those downloaded by a user with the "Copy File..." menu option in the Oracle E-Business Suite Report File Viewer and those "temporary" files which are automatically downloaded by custom editors. For example, to set the size to 64K you enter 65536. If this profile is null, there is no size limit.

Only the System Administrator can update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FS_MAX_TRANS.

Concurrent Processing Manager

The internal name for this profile category is FND_CP_MANAGER.

Concurrent:Debug Flags

Your Oracle support representative may access this profile option to debug Transaction Managers. Otherwise, it should be set to null.

Users cannot see nor change this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes

Level	Visible	Allow Update
User	Yes	Yes

The internal name for this profile option is CONC_DEBUG.

Concurrent:OPP Process Timeout

This profile option specifies the amount of time the manager waits for the OPP to actually process the request.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_PP_PROCESS_TIMEOUT.

Concurrent:OPP Response Timeout

This profile option specifies the amount of time a manager waits for the OPP to respond to its request for post processing.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_PP_RESPONSE_TIMEOUT.

Concurrent:PCP Instance Check

This profile option controls whether Parallel Concurrent Processing (PCP) will be sensitive to the state (up or down) of the database instance connected to on each middle-tier node.

When this profile option is set to "OFF", PCP will not provide database instance failover support; however, it will provide middle-tier node failover support when a node goes down.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	No	No

The internal name for this profile option is CP_INSTANCE_CHECK.

Concurrent Processing Submission

The internal name for this profile category is FND_CP_SUBMISSION.

Concurrent:Allow Debugging

This profile option allows debug options to be accessed by the user at submit time.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_CONC_ALLOW_DEBUG.

Concurrent:Enable Request Submission in View Mode

Use this profile option to enable Request Submission in View Requests mode.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_FNDRSRUN_MODE.

Concurrent:Request Start Time

You can set the date and time that your requests are available to start running.

- If the start date and time is at or before the current date and time, requests are available to run immediately.
- If you want to start a request in the future, for example, at 3:45 pm on June 12, 2002, you enter 2002/06/12 15:45:00 as the profile option value.

  Important: You must ensure that this value is in canonical format (YYYY/MM/DD HH24:MI:SS) to use the Multilingual Concurrent Request feature.

- You must include both a date and a time.
- Changing this value does not affect requests that you have already submitted.
- Users can override the start time when they submit requests. Or, this profile option can be left blank and users will be prompted for a start time when they submit requests.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_REQ_START.

Concurrent: Show Requests Summary After Each Request Submission

Using this new profile option, you can choose to either have the Requests Summary displayed each time you submit a request, or retain the request submission screen.

The default is "Yes". "Yes" means the Requests Summary screen is displayed each time you submit a request.

If you choose "No", a decision window is opened asking you if you wish to submit another request. When you choose to submit another request you are returned to the submission window and the window is not cleared, allowing you to easily submit copies of the same request with minor changes.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_REQ_SUMMARY.

Concurrent:Validate Request Submission

This profile option prompts users in SRS form if no options or parameters have been changed from their defaults.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_VALIDATE_SUBMISSION.

Printer

You can select the printer which prints your reports. If a printer cannot be selected, contact your system administrator. Printers must be registered with Oracle E-Business Suite.

This profile option can be managed by AutoConfig.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is PRINTER.

Concurrent Processing View Requests

The internal name for this profile category is FND_CP_VIEW_REQUESTS.

Concurrent:Show Request Set Stages

Set this profile option value to *Yes* to show request set stages in the concurrent request screens.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CONC_SHOW_STAGES.

Concurrent:URL Lifetime

The numeric value you enter for this profile option determines the length of time in minutes a URL for a request ouput is maintained. After this time period the URL will be deleted from the system. This profile option only affects URLs created for requests where the user has entered values in the notify field of the Submit Request or Submit Request Set windows.

> **Important:** All request ouput URLs are deleted when the Purge Concurrent Requests and Manager... program is run even if the URL liftime has not expired.

Users can see and update this profile option.

This profile option is visible and updatable at the all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	No
Responsibility	Yes	No
User	Yes	No

The internal name for this profile option is CONC_URL_LIFETIME.

FND: Default Request Days

This profile option specifies the default number of days to view requests.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_DEFAULT_REQUEST_DAYS.

Maximum Page Length

Determines the maximum number of lines per page in a report.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is MAX_PAGE_LENGTH.

Viewer: Application for HTML, PCL, PDF, Postscript, Text, and XML

These profile options determine the applications a user will use to view reports in the given output formats. For example, you could set Viewer: Application for Text to 'application/word' to view a Text report in Microsoft Word.

Valid values are defined by the system administrator in the Viewer Options form.

Users can see and update these profile options.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal names for these profile options are FS_MIME_HTML, FS_MIME_PCL, FS_MIME_PDF, FS_MIME_PS, FS_MIME_TEXT, and FS_MIME_XML.

Viewer:Default Font Size

Using this new profile option, you can set the default font size used when you display report output in the Report Viewer.

The valid values for this option are 6, 8, 10, 12, and 14.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FNDCPVWR_FONT_SIZE.

Viewer: Text

The Viewer: Text profile option allows you to send report output directly to a browser window rather than using the default Report Viewer. Enter "Browser" in this profile option to enable this feature.

Users can see and update the Viewer:Text profile option.

This profile option is both visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is EDITOR_CHAR.

Database

The internal name for this profile category is FND_DATABASE.

Database Instance

Entering a valid two_task connect string allows you to override the default two_task. This profile is specifically designed for use with Oracle Parallel Server, to allow different responsibilities and users to connect to different nodes of the server.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is INSTANCE_PATH.

FND: Resource Consumer Group

Resource consumer groups are used by the Oracle8i Database Resource Manager, which allocates CPU resources among database users and applications. Each form session is assigned to a resource consumer group. The system administrator can assign users to a resource consumer group for all of their forms sessions and transactions. If no resource

consumer group is found for a process, the system uses the default group "Default_Consumer_Group".

Users can see this profile option, but they cannot update it.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_RESOURCE_CONSUMER_GROUP.

Two Task

This profile option should be set by AutoConfig. only.

The TWO_TASK for the database. This profile is used in conjunction with the Gateway User ID profile to construct a connect string for use in creating dynamic URLs for the Web Server. This should be set to the SQL*NET. alias for the database.

> **Note:** The TWO_TASK must be valid on the node upon which the WebServer is running

Users can see and but not update this profile option.

This profile option is visible at all levels but may only be updated at site level.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	No
Responsibility	Yes	No
User	Yes	No

The internal name for this profile option is TWO_TASK.

Debug

The internal name for this profile category is FND_DEBUG.

Account Generator:Debug Mode

This profile option controls Oracle Workflow process modes for the Account Generator feature in flexfields. This profile option should normally be set to "No" to improve performance. If you are testing your Account Generator implementation and using the Oracle Workflow Monitor to see your results, set this profile option to "Yes".

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is ACCOUNT_GENERATOR:DEBUG_MODE.

FND: Override Directory

The FND:Override Directory profile option is used by the Work Directory feature. The value of FND: Override Directory should be the directory containing your alternate files. Typically, this profile option should be set at the User level only.

Using the Work Directory and this profile option should be done for debugging only, as they present a security risk.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is APPLWRK.

Utilities: Diagnostics

Utilities: Diagnostics determines whether a user can automatically use the Diagnostics features. If Utilities:Diagnostics is set to Yes, then users can automatically use these features. Otherwise, certain Diagnostics features will be accessible only if the users have the necessary permissions granted to them. See: Controlling Access to the Oracle Forms-based Applications Diagnostics Menu, *Oracle E-Business Suite Maintenance Guide* for more information.

Users cannot see nor change this profile option.

This profile option is visible and updatable at the all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is DIAGNOSTICS.

Utilities:SQL Trace

This profile option is used by concurrent processing only. SQL trace files can be generated for individual concurrent programs. The trace can be enabled at the user level by setting the profile "Utilities:SQL Trace" to "Yes". This profile can be enabled for a user only by System Administrator so that it is not accidentally turned on and disk usage can be monitored.

For more information on SQL trace, see the Oracle database documentation.

Users cannot see nor change this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes

Level	Visible	Allow Update
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is SQL_TRACE.

Deployment

The internal name for this profile category is FND_DEPLOYMENT.

Forms Runtime Parameters

Use this profile to specify certain forms runtime parameters. The profile value must be entered in as parameter=value. Each parameter-value pair must be separated by a single space. For example:

record=collect log=/tmp/frd.log debug_messages=yes

In order for the parameters updated in this profile option to go into effect, you must exit and log back in to Oracle E-Business Suite.

Users can see but not update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_MORE_FORM_PARAMS.

Gateway User ID

Oracle login for gateway account. This should be the same as the environment variable GWYUID. For example, *APPLSYSPUB/PUB*.

This profile option can be managed by AutoConfig.

Users cannot see or update this profile option.

This profile option is visible at all levels but can only be updated at the site level.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	No
Responsibility	Yes	No
User	Yes	No

The internal name for this profile option is GWYUID.

Site Name

Site Name identifies an installation of Oracle E-Business Suite.

The Site Name appears in the title of the MDI window. If you want additional information on your installation to appear in the title, for example, "Test" or "Production", you can add that information here.

Users cannot see nor change this profile option.

This profile option is visible and updatable at the site level.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	No	No

The internal name for this profile option is SITENAME.

Socket Listener Port

This profile option defines the port number used by the Forms Client Controller.

The default value for this profile option is '6945'.

The E-Business Suite Home page uses the Socket Listener Port profile for launching

forms from Framework HTML sessions. With this architecture, a user navigating through different forms/responsibilities in a Framework session will reuse the same Oracle Forms session instead of opening multiple ones. So a user will never have more than one Forms session open on his/her PC at any given time, for a given database.

It is possible to have multiple Oracle Forms sessions open where each is connected to a different database, but the Socket Listener Port profile must be set to a different value beforehand on each database. For example, set it to 6945 on database A, 6946 on database B, and 6947 on database C. This profile option must be set at the site level in advance of any users attempting to use this functionality, as it cannot be set on a per-user basis.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	No	No

The internal name for this profile option is SOCKET_LISTENER_PORT.

TCF: HOST

Set this to the name of the host running the TCF Socket Server.

This profile option is visible at all levels and updatable at the site and application level only.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	No
User	Yes	No

The internal name for this profile option is TCF:HOST.

TCF: PORT

Set this profile option to the port number at which TCF Socket Server accepts connections.

This profile option can be managed by AutoConfig.

Users can see and but not update this profile option.

This profile option is visible at all levels and updatable at the site and application level only.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	No
User	Yes	No

The internal name for this profile option is TCF:PORT.

Discoverer

The internal name for this profile category is FND_DISCOVERER.

ICX: Discoverer Launcher, Forms Launcher

These profile options are used by the Oracle E-Business Suite Personal Homepage.

Set the site level value of each of these profile options to the base URL for launching each application. The profile option value should be sufficient to launch the application, but should not include any additional parameters which may be supplied by the Personal Homepage.

These profile options can be managed by AutoConfig.

Users can see these profile options, but they cannot update them.

These profile options are visible and updatable at all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for these profile options are ICX_DISCOVERER_LAUNCHER and ICX_FORMS_LAUNCHER.

Document Sequencing

The internal name for this profile category is FND_DOC_SEQ.

Sequential Numbering

Sequential Numbering assigns numbers to documents created by forms in Oracle financial products. For example, when you are in a form that creates invoices, each invoice document can be numbered sequentially.

Sequential numbering provides a method of checking whether documents have been posted or lost. Not all forms within an application may be selected to support sequential numbering.

Sequential Numbering has the following profile option settings:

Always Used You may not enter a document if no sequence exists for it.

Not Used You may always enter a document.

Partially Used You will be warned, but not prevented from entering a document, when no sequence exists.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at the site, application, and responsibility levels.

> **Note:** If you need to control Sequential Numbering for each of your set of books, use the 'Responsibility' level. Otherwise, we recommend that you use either the 'Site' or 'Application' level to set this option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	No	No

The internal name for this profile option is UNIQUE:SEQ_NUMBERS.

Flexfields

The internal name for this profile category is FND_FLEXFIELDS.

Flexfields:AutoSkip

You can save keystrokes when entering data in your flexfields by automatically skipping to the next segment as soon as you enter a complete valid value into a segment.

- "Yes" means after entering a valid value in a segment, you automatically move to the next segment.
- "No" means after entering a valid value in a segment, you must press [Tab] to go to the next segment.

> **Note:** You may still be required to use tab to leave some segments if the valid value for the segment does not have the same number of characters as the segment. For example, if a segment in the flexfield holds values up to 5 characters and a valid value for the segment is 4 characters, AutoSkip will not move you to the next segment.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes

Level	Visible	Allow Update
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FLEXFIELDS:AUTOSKIP.

Flexfields:BiDi Direction

This profile option controls the appearance of the flexfields window in Applications running in Semitic languages. Possible values are "Left To Right" and "Right To Left".

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FLEXFIELDS:BIDI_DIRECTION.

Flexfields:Open Descr Window

You can control whether a descriptive flexfield window automatically opens when you navigate to a customized descriptive flexfield.

- "Yes" means that the descriptive flexfield window automatically opens when you navigate to a customized descriptive flexfield.
- "No" means that when you navigate to a customized descriptive flexfield, you must choose **Edit Field** from the Edit menu or use the List of Values to open the descriptive flexfield window.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FLEXFIELDS:OPEN_DESCR_WINDOW.

> **Note:** This profile option does not apply to descriptive flexfields in folders.

Flexfields:Open Key Window

You can control whether a key flexfield window automatically opens when you navigate to a key flexfield.

- "Yes" means that the key flexfield window automatically opens when you navigate to a key flexfield.
- "No" means that when you navigate to a key flexfield, you must choose **Edit Field** from the Edit menu or use the List of Values to open the key flexfield window.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FLEXFIELDS:OPEN_KEY_WINDOW.

Flexfields:Shorthand Entry

If shorthand flexfield entry is defined for your flexfield, you can use a shorthand alias to automatically fill in values for some or all of the segments in a flexfield.

Not Enabled Shorthand Entry is not available for any flexfields for this user, regardless of whether shorthand aliases are defined.

New Entries Only Shorthand Entry is available for entering new records in most foreign key forms. It is not available for combinations forms, updating existing records, or entering queries.

Query and New Entry Shorthand Entry is available for entering new records or for entering queries. It is not available for updating existing records.

All Entries Shorthand Entry is available for entering new records or updating old records. It is not available for entering queries.

Always Shorthand Entry is available for inserting, updating, or querying flexfields for which shorthand aliases are defined.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FLEXFIELDS:SHORTHAND_ENTRY.

Flexfields:Show Full Value

If an alias defines valid values for *all* of the segments in a flexfield, and Flexfields: Shorthand Entry is enabled, when you enter the alias the flexfield window does not appear.

"Yes" displays the full flexfield window with the cursor resting on the last segment.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FLEXFIELDS:SHOW_FULL_VALUE.

Flexfields:Validate On Server

This profile option is set to "Yes" to enable server side, PL/SQL flexfields validation for Key Flexfields. This improves performance when using Key Flexfields over a wide area network by reducing the number of network round trips needed to validate the entered segment combinations.

You may find, however, that your validation's performance is better with client side validation. In this case, set this profile option to "No".

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FLEXFIELDS:VALIDATE_ON_SERVER.

Folders

The internal name for this profile category is FND_FOLDERS.

Folders:Allow Customization

Your system administrator controls whether you can create or customize a folder definition layout in folder block.

- "Yes" means that you can create or customize a folder definition, that is, the entire Folder menu is enabled in the folder block.
- "No" means that you can only open an existing folder definition in a folder block, that is, only the Open option is enabled in the Folder menu.

Users can see this profile option, but they cannot update it.

Level	Visible	Allow Update
Site	No	No
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is FLEXVIEW:CUSTOMIZATION.

Forms UI

The internal name for this profile category is FND_FORMS_UI.

Flexfields:LOV Warning Limit

Use Flexfields:LOV Warning Limit to improve efficiency when retrieving a list of values.

Sometimes, particularly when no reduction criteria has been specified, an LOV can take a very long time to run if there is a very significant amount of data in it. Set this profile option to the number of rows to be returned before the user is asked whether to continue retrieving the entire list.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is QUICKPICK_ROWS_BEFORE_WARN.

FND: Enable Cancel Query

Oracle E-Business Suite allows end users to cancel certain long-running queries, such as retrieving data in a block. When these operations exceed a threshold of time, approximately ten seconds, a dialog will display that allows the user to cancel the query.

Set the FND: Enable Cancel Query profile option to Yes if you wish to enable the ability to cancel a form query. This profile option may be set at the site, application, responsibility or the user level.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_ENABLE_CANCEL_QUERY.

FND: Indicator Colors

The default for this profile option is null, which means "Yes." When this profile option is set to Yes:

- Required fields are displayed in yellow.

- Queryable fields are displayed in a different color while in enter-query mode.
- Fields that cannot be entered (read-only) are rendered in dark gray.

Users can see and update this profile option.

Level	Visible	Allow Update
Site	No	No
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is FND_INDICATOR_COLORS.

Forms Keyboard Mapping File

Use this profile option to define the path of the Keyboard Mapping File.

The "Keys" window displays the keystrokes to perform standard Forms operations, such as "Next Block" and "Clear Record." This window can be viewed at anytime by pressing Ctrl+k. The keyboard mappings can be customized as follows:

- The System Administrator must locate the Oracle Forms resource file on the middle tier, typically called fmrweb.res.
- Make a copy of the file, name it as desired, and locate it in the same directory as the original
- Open the new file in any text editor and make the desired keystroke mapping changes. Comments at the top of the file explain how the mappings are performed.
- To run the new mapping file, specify the complete path and file name in this profile option.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes

Level	Visible	Allow Update
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_FORMS_TERM.

Indicate Attachments

This profile option allows you to turn off indication of attachments when querying records (for performance reasons).

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is ATCHMT_SET_INDICATOR.

Java Color Scheme

If the Java Look and Feel profile option is set to Oracle, the Java Color Scheme can be specified as follows:

- Standard (default)
- Teal
- Titanium
- Red
- Khaki

- Blue
- Olive
- Purple

The Java Color Scheme profile has no effect if the Java Look and Feel is set to Generic.

> **Important:** Setting the Java Color Scheme profile option to a value other than 'standard' (the default value) can have a considerable impact on forms user response time performance.
>
> For some users, setting this profile option to a value other than 'standard' may be desirable for accessibility reasons. See: Oracle E-Business Suite Accessibility Features, *Oracle E-Business Suite User's Guide* and "Accessibility in Oracle Forms Applications" at http://www.oracle.com/accessibility/apps02.html.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_COLOR_SCHEME.

Java Look and Feel

Oracle E-Business Suite Professional User Interface (Forms-based applications) can be run with either the Oracle Look and Feel or the Generic Look and Feel. The Oracle Look and Feel consists of a new look and feel for each item, and a predefined set of color schemes. The Generic Look and Feel adheres to the native interface and color scheme of the current operating system.

To specify the look and feel set this profile to "generic" or "oracle".

If the Oracle Look and Feel is used, the profile Java Color Scheme can be set. The Java Color Scheme profile has no effect if the Java Look and Feel is set to Generic.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_LOOK_AND_FEEL.

Framework Logging and Alerting

The internal name for this profile category is FND_FWK_LOGGING_ALERTING.

FND: Log Filename for Middle-Tier

The file name for the file to hold debugging messages used in the Logging Service. If the value of this profile option is null, then the Logging Service is turned off.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is AFLOG_FILENAME.

FND: Log Level

The Logging Service can filter out debugging messages depending on their priority level.. There are five levels of the Debug/Trace Service:. In order from highest priority to lowest priority, they are: Errors, Exceptions, Events, Procedures, and Statements. The Debug Log Level is the lowest level that the user wants to see messages for.. The possible profile option values are Null (which means off), and the five priority levels above. For instance, if the "FND: Debug Log Level" profile is set to "EVENT", then the

file will get the messages that the programmer had marked as "EVENT", "EXCEPTION", or "ERROR".

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is AFLOG_LEVEL.

FND: Log Module

The Logging Service can filter out debugging messages depending on their module. Module names are unique across applications and coding languages. If a module is specified for this profile option, then only messages for that module will be written to the log file. If this profile option is left blank then messages for all modules will be written to the log file.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is AFLOG_MODULE.

Help System

The internal name for this profile category is FND_IHELP.

Applications Help Web Agent

Applications Help Web Agent is optional and should only be used if you want to launch online help on a web server different from the one specified by the Applications Servlet Agent.

> **Important:** For most installations, this profile should be set to NULL. Only specify a value if you want to use a different web server than that for the Applications Servlet Agent.

Specify the entire online help URL for this profile's value.

If this profile option is not set, the online help tree navigator will default to starting up at the host name and port number that is specified by the Applications Servlet Agent profile option. The DBC file used will be that of the database where online help was invoked.

This profile option can be managed by AutoConfig.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is HELP_WEB_AGENT.

Help Localization Code

This code determines which localized context-sensitive help files a user accesses.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at the responsibility and user levels.

Level	Visible	Allow Update
Site	No	No
Application	No	No
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is HELP_LOCALIZATION_CODE.

Help Tree Root

This profile option determines which tree is shown in the navigation frame when context-sensitive help is launched.

If Help Tree Root is set to "null" or "NULL" (case insensitive), then the online help is launched in a single frame, without the navigation and search features.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is HELP_TREE_ROOT.

Help Utility Download Path

Use this profile option to define the directory into which the Help Utility downloads help files from the Oracle E-Business Suite Help System.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is HELP_UTIL_DL_PATH.

Help Utility Upload Path

Use this profile option to define the directory from which the Help Utility uploads help files to the Oracle E-Business Suite Help System.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is HELP_UTIL_UL_PATH.

Multi Organization Setup

The internal name for this profile category is FND_MULTI_ORG.

MO:Operating Unit

In Multiple Organization installations, Oracle E-Business Suite uses the profile option MO: Operating Unit to link an operating unit to a responsibility. You must set this profile option for each responsibility. For more information on setting this profile option, see: *Multiple Organizations in Oracle E-Business Suite*.

Users can see but not update this profile option.

This profile option is visible and updatable at the responsibility level only.

Level	Visible	Allow Update
Site	No	No
Application	No	No
Responsibility	Yes	Yes
User	No	No

The internal name for this profile option is ORG_ID.

NLS

The internal name for this profile category is FND_NLS.

Currency:Mixed Precision

Use Mixed Currency Precision to specify how many spaces are available to the right of the decimal point when displaying numbers representing different currencies.

- Normally, currency numbers are right-justified.
- Each currency has its own precision value that is the number of digits displayed to the right of a decimal point. For U.S. dollars the precision default is 2, so an example display is 345.*70*.
- Set Mixed Currency Precision to be equal to or greater than the *maximum* precision value of the currencies you are displaying.

  For example, if you are reporting on rows displaying U.S. dollars (precision=2), Japanese yen (precision=0), and Bahraini dinar (precision=3), set Mixed Currency Precision=3.

  Note: The Currency profile options pertain to currency only, not to other numeric fields.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CURRENCY:MIXED_PRECISION.

Currency:Negative Format

You can use different formats to identify negative currency. The default identifier is a hyphen (-) preceding the currency amount, as in "-xxx". You can also select:

Angle brackets < > < xxx >

Trailing hyphen - xxx -

Parentheses ( ) (xxx)

Square Brackets [ ] [xxx]

> **Note:** The Currency profile options pertain to currency only, not to other numeric fields.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CURRENCY:NEGATIVE_FORMAT.

Note: Currency:Negative Format only affects the display of currency values . Non-currency negative numbers appear with a preceding hyphen regardless of the option selected here.

Currency:Positive Format

You can use different formats to identify positive currency values. The default condition is no special identifier.

Note: The Currency profile options pertain to currency only, not to other numeric fields.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CURRENCY:POSITIVE_FORMAT.

Currency:Thousands Separator

You can separate your currency amounts in thousands by placing a thousands separator. For example, one million appears as 1,000,000.

Users can see and update this profile option.

Note: The Currency profile options pertain to currency only, not to other numeric fields.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is CURRENCY:THOUSANDS_SEPARATOR.

Default Country

This is the default source for the Country field for all address zones and is used by the Flexible Address Formats feature, the Flexible Bank Structures feature and the Tax Registration Number and Taxpayer ID validation routines.

The profile can be set to any valid country listed in the Maintain Countries and Territories form and can be set to a different value for each user.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is DEFAULT_COUNTRY.

FND: Native Client Encoding

FND: Native Client Encoding indicates the character set that a client machine uses as its native character set. The value must be one of the Oracle character sets and should correspond to the client native character set. The character set used in a client machine varies depending on language and platform. For example, if a user uses a Windows machine with Japanese, the value should be JA16SJIS. But if a user uses a Solaris

machine with Japanese, the value should be JA16EUC. The value is normally set in the user level since each user uses different machine, but it can be set in every level for a default value.

This profile option is used when storing text data. When a user uploads text files as attachments, the current value of FND: Native Client Encoding is stored along with the text data. With the value of this profile option, the server can then convert the text data to another character set as necessary when the text data is downloaded.

Users can see and update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_NATIVE_CLIENT_ENCODING.

ICX: HTML directory

This profile is used by some applications to construct URLs for certain pages. It is usually set to 'OA_HTML'.

Users can see but not update the profile value.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is ICX_OA_HTML.

ICX: Preferred Currency

This profile determines in which currency a user will see the currency number in the UI.

For example, the source currency number might be stored in database such as 10.00 as US Dollar (USD), but the displayed currency number is based on the currency set in this profile option such as 1,200 as Japanese Yen (JPY). In this multi-currency conversion, USD is source currency and JPY is the profile option value.

This profile option is for currency display purpose especially for self-service type applications.

This profile option is a generic preference that a user can set through the Oracle Application Framework Preferences page. The profile option value is used across Oracle E-Business Suite so that the user sees currency numbers in all applications based on the currency chosen.

The currencies must be set up through the Oracle General Ledger application properly (the following must be set properly: Enabled/Disabled, Active Date and Exchange ratio between currencies). Proper setup ensures that the currency chosen is available in the system, and the currency number can be converted from the source (functional) currency to the target currency (the currency chosen by a user as this profile option value) with the specified exchange ratio. This profile option is tightly linked to GL currency setup. For more information, see: Defining Currencies, *Oracle General Ledger User's Guide*.

Users can see and update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is ICX_PREFERRED_CURRENCY.

Server Timezone

The time zone of the database server.

Users can see this profile option, but they cannot update it.

Level	Visible	Allow Update
Site	Yes	Yes

Level	Visible	Allow Update
Application	No	No
Responsibility	No	No
User	No	No

The internal name for this profile option is SERVER_TIMEZONE_ID.

Personalization

The internal name for this profile category is FND_PERSONALIZATION.

Initialization SQL Statement - Custom

This profile option allows you to add site-specific initialization code (such as optimizer settings) that will be executed at database session initialization. The value of this profile option must be a valid SQL statement.

The system administrator may set this profile option at any level.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_INIT_SQL.

Security

The internal name for this profile category is FND_SECURITY.

AuditTrail:Activate

You can turn AuditTrail on or off (Yes or No). The default setting is No (Off).

When you enter or update data in your forms, you change the database tables underlying the forms you see and use.

AuditTrail tracks which rows in a database table(s) were updated at what time and which user was logged in using the form(s).

- Several updates can be tracked, establishing a trail of audit data that documents the database table changes.
- AuditTrail is a feature enabled on a form-by-form basis by a developer using Oracle's Application Object Library.
- All the forms that support AuditTrail are referred to as an *audit set*.
- Not all forms may be enabled to support AuditTrail.
- To enable or disable AuditTrail for a particular form, you need access to Oracle Application Object Library's Application Developer responsibility.

Users cannot see nor change this profile option.

This profile option is visible and updatable at the site and application levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	No	No
User	No	No

The internal name for this profile option is AUDITTRAIL:ACTIVATE.

Enable Security Groups

This profile option is used by the Security Groups feature, which is used by HRMS security only. For more information on Security Groups, see the Oracle HRMS documentation.

The possible values are 'None' (N), and 'Service Bureau' (Y).

Only the System Administrator can update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes

Level	Visible	Allow Update
Application	Yes	Yes
Responsibility	No	No
User	No	No

The internal name for this profile option is ENABLE_SECURITY_GROUPS.

Hide Diagnostics Menu Entry

This profile option determines whether users can access the Diagnostics menu entry from the Help menu. The default value is Yes, with the Diagnostics menu entry is hidden. If it is set to No, the Diagnostics menu entry is visible.

Users cannot see nor change this profile option.

This profile option is visible and updatable at the all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is FND_HIDE_DIAGNOSTICS.

ICX: Limit time

This profile option determines the absolute maximum duration (in hours) of a user's session, regardless of activity.

Users cannot see or update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes

Level	Visible	Allow Update
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is ICX_LIMIT_TIME.

ICX: Session Timeout

This profile option determines the length of time (in minutes) of inactivity in a user's session before the session is disabled. If the user does not perform any operation in Oracle E-Business Suite for longer than this value, the session is disabled. The user is provided the opportunity to re-authenticate and re-enable a timed-out session. If re-authentication is successful, the session is re-enabled and no work is lost. Otherwise, Oracle E-Business Suite ends the session without saving pending work.

If this profile option to 0 or NULL, then user sessions will never time out due to inactivity.

Users can see this profile option, but they cannot update it.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is ICX_SESSION_TIMEOUT.

Node Trust Level

Determines the level of trust assigned to a Web server. This profile option uses the Server hierarchy type. This profile option is used in conjunction with the profile option Responsibility Trust Level. For more information on using these profile options, see: Restricting Access to Responsibilities Based on User's Web Server, page 1-16.

Users can see but not update this profile option.

This profile option is visible and updatable at the site and server level only.

Level	Visible	Allow Update
Site	Yes	Yes
Server	Yes	Yes
User	No	No

The internal name for this profile option is NODE_TRUST_LEVEL.

Responsibility Trust Level

Responsibilities or applications with the specified level of trust can only be accessed by an application server with at least the same level of trust.

This profile option is used in conjunction with the profile option Node Trust Level. For more information on using these profile options, see: Restricting Access to Responsibilities Based on User's Web Server, page 1-16.

Users can see this profile option, but they cannot update it.

The system administrator access is described in the following table:

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	No	No

The internal name for this profile option is APPL_SERVER_TRUST_LEVEL.

Sign-On:Audit Level

Sign-On:Audit Level allows you to select a level at which to audit users who log in to Oracle E-Business Suite. Four audit levels increase in functionality: None, User, Responsibility, and Form.

None is the default value, and means do not audit any users who log in to Oracle E-Business Suite.

Auditing at the User level tracks:

- who signs on to your system
- the times users log on and off

Auditing at the Responsibility level performs the User level audit functions and tracks:

- the responsibilities users choose
- how much time users spend using each responsibility

Auditing at the Form level performs the Responsibility level audit functions and tracks:

- the forms users choose
- how long users spend using each form
- System Administrator visible, updatable at all levels.

Users cannot see nor change this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is SIGNONAUDIT:LEVEL.

Sign-On:Notification

This profile option lets you specify whether to display a tip message in the Oracle E-Business Suite home page that informs the user how many open notifications are in his or her Oracle Workflow worklist. The message also requests the user to view and respond to these notifications.

- Yes - The tip message listing the number of open notifications for the user appears on the Oracle E-Business Suite home page.
- No - The tip message is disabled. This setting is the default value for this profile option.

Users can see and update this profile option.

This profile option is visible and updatable at all four levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is SIGNONAUDIT:NOTIFY.

Signon Password Case

Oracle E-Business Suite gives you the ability to control case sensitivity in user passwords through this profile option. This profile has two possible settings:

- **Sensitive (default)** - All new or changed passwords are treated as case sensitive. Passwords are stored and compared with the password case preserved. During validation, the entered password must match the decrypted version, or an error message is displayed.

  If some existing users still have case insensitive passwords from a previous release, a password expiration utility can be used to require all users to convert to case sensitive passwords upon the next login. This utility expires all passwords in FND_USER, including that of SYSADMIN and default Vision accounts, and can be run either as an SQL script (`$FND_TOP/sql/AFCPEXPIRE.sql`) or as a concurrent program (`FNDCPEXPIRE_SQLPLUS`).

- **Insensitive** - With this setting, passwords are treated as case insensitive, and are stored and compared in uppercase. During validation, the entered password and the decrypted password are compared in uppercase. If the passwords do not match, an error message is displayed.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes

Level	Visible	Allow Update
Application	No	No
Responsibility	No	No
User	No	No

The internal name for this profile option is SIGNON_PASSWORD_CASE.

Signon Password Custom

This profile specifies the full name of the class containing custom password validation logic.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is SIGNON_PASSWORD_CUSTOM.

Signon Password Failure Limit

The Signon Password Failure Limit profile option determines the maximum number of login attempts before the user's account is disabled.

Users cannot see or update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No

Level	Visible	Allow Update
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is SIGNON_PASSWORD_FAILURE_LIMIT.

Signon Password Hard to Guess

The Signon Password Hard to Guess profile option sets rules for choosing passwords to ensure that they will be "hard to guess." A password is considered hard-to-guess if it follows these rules:

- The password contains at least one letter and at least one number.
- The password does not contain the username.
- The password does not contain repeating characters.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is SIGNON_PASSWORD_HARD_TO_GUESS.

Signon Password Length

Signon Password Length sets the minimum length of an Applications signon password. If no value is entered the minimum length defaults to 5.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is SIGNON_PASSWORD_LENGTH.

Signon Password No Reuse

This profile option specifies the number of days that a user must wait before being allowed to reuse a password.

Users can see but not update this profile option.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is SIGNON_PASSWORD_NO_REUSE.

Single Sign-On Account Settings

The internal name for this profile category is FND_SSO_ACCOUNT_SETTINGS.

ICX: Client IANA Encoding

This profile option is used to determine the character set of text displayed by Java Server pages. The value is the code set of the middle tier. It is used to allow the online help system to support languages other than American English. The default setting is the Western European character set (ISO-8859-1).

This profile option can be managed by AutoConfig.

This profile option should be set only at the site level.

Important: This profile option must not be set to NULL at the site level.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at the all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is ICX_CLIENT_IANA_ENCODING.

Web Server Deployment

The internal name for this profile category is FND_WS_DEPLOYMENT.

Applications Servlet Agent

This profile option must be set to the URL base for the servlet execution engine on Apache. Oracle E-Business Suite uses the value of this profile option to construct URLs for JSP and SERVLET type functions. The syntax is:

```
https://<hostname>:<port>/<servlet_zone>
```

Example:

```
https://ap523sun.us.oracle.com:8888/oa_servlets
```

This profile option can be managed by AutoConfig.

Users can see this profile option, but they cannot update it.

This profile option is visible and updatable at all levels.

Level	Visible	Allow Update
Site	Yes	Yes

Level	Visible	Allow Update
Application	Yes	Yes
Responsibility	Yes	Yes
User	Yes	Yes

The internal name for this profile option is APPS_SERVLET_AGENT.

Applications Web Agent

Provides the base URL for the Applications Schema's WebServer DAD. You set this profile option during the install process.

This profile option can be managed by AutoConfig.

This profile option is visible and updatable at all levels.

Level	Visible	Allow Update
Site	Yes	Yes
Application	No	No
Responsibility	No	No
User	Yes	Yes

The internal name for this profile option is APPS_WEB_AGENT.

6

Setting Up Printers

Printers and Printing

Oracle E-Business Suite offers two printing solutions to handle all your printing requirements. For most printing needs, the Pasta Utility offers quick setup and easy maintenance. For additional flexibility, Oracle E-Business Suite allows you to define your own printer drivers and print styles.

- To set up your printers using Pasta, see Printer Setup with Pasta, page 6-10.

  Note: Pasta is required to print using UTF8.

- To set up your printers using a custom solution, see Customizing Printing Support in Oracle E-Business Suite, page 6-31.

Overview

When you run an Oracle E-Business Suite report, Oracle Reports generates and formats the output. A completed report is sent to the operating system by the concurrent manager, which issues an operating system print command, or calls a custom print program or subroutine that issues an operating system print command.

Oracle Reports and Report Generation

Oracle Reports includes page break, carriage return, line feed, text bold on/off, and text underline on/off instructions within the output file. The values are retrieved from a SQL*ReportWriter (SRW) driver file.

When the report is generated for online viewing, Oracle Reports uses the SRW driver named by the print style in the Print Styles form.

When the report is to be printed, Oracle Reports uses the SRW driver named by the Oracle E-Business Suite printer driver in the Printer Drivers form.

The dimensions of a report are determined by the columns and rows values in the print style, defined using the Print Styles form. These values override the width and height values in an SRW driver file.

Concurrent Manager Issues or Calls a Print Command

When a report program finishes running, the concurrent manager prepends an initialization string, and appends a printer reset string to the output file. Both strings are defined using the Printer Drivers form.

An Oracle E-Business Suite printer driver is invoked by issuing a print command or by calling a print program or subroutine.

When the printer driver method is *Command*, the concurrent manager can issue an operating system print command and arguments, entered in the Arguments field of the Printer Drivers form.

When the printer driver method is *Program*, the concurrent manager can call a custom print program, named (along with its path) in the Name field of the Printer Drivers form. Arguments to the program may be entered in the form's Arguments field.

When the printer driver method is *Subroutine*, the concurrent manager calls a predefined Oracle E-Business Suite subroutine that passes a print command and arguments to the printer via the operating system. The subroutine name is entered in the Program Name field of the Printer Drivers form.

The concurrent manager may provide values for four arguments to an operating system print command or custom print program:

- the name of the file to be printed
- the operating system name of the target printer
- the title of the file, which appears on a header page if it is printed
- the number of copies to be printed

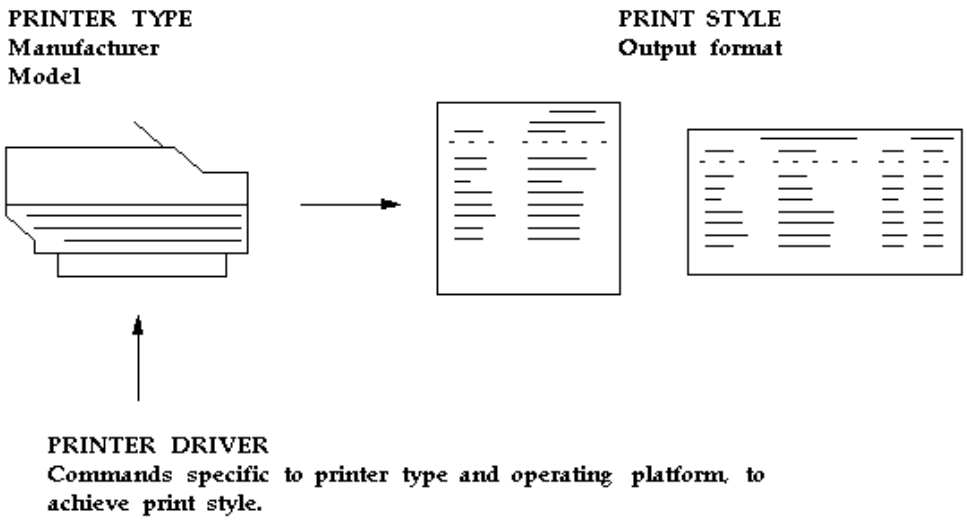
Printer Types, Print Styles, and Printer Drivers

The commands that a printer can understand vary from one type of printer to another. A *printer type* identifies a printer by manufacturer and model.

A *print style* tellsthe printer how a printed output should look. A *printer driver* delivers commands that tell the printer how to output the specified print style.

The ability to print a report in a particular print style depends on the type of printer the report file is sent to.

For each print style that a particular type of printer can print, a printer driver specific to the printer type and the operating system is required.



Printer Types

The printer type is the printer manufacturer and model. Two examples are a DEC LN03 printer and an HP Laserjet III printer.

Print Styles

A Print style defines the page format for a printer, such as the number of lines per page, the width of each line, and whether a header page should be printed.

Each printer type can have one or more associated print styles.

Print styles allow you to set up report dimensions on a variety of printers. You can tailor your page setups while providing consistent-looking reports from printer to printer.

For example, users may wish to print a menu report with a wider left margin to allow for hole punching the paper. As System Administrator, you register this new style, which users can then access if the printer type supports it.

At report submission time, users select the style in which to output the report.

- Only styles available on the destination printer are displayed.
- Some concurrent programs predefine either the printer or the print style, and these values cannot be changed.

Printer Drivers

To print in a particular style from a particular printer type, you define a printer driver.

A printer driver is the mechanism that delivers a report's output along with its commands to the target printer.

Concurrent managers determine what drivers to use depending on what the print style is and what printer type the report is to be sent to.

You need to define a printer driver for each print style that you want to use with a specific printer type on a specific platform.

Sequence of Printing Events

The concurrent manager associates a print style and a printer driver with the destination printer's printer type. This combination of print style and printer driver is defined in the Printer Types form.

A printer driver tells the destination printer how to interpret the format. An SRW Driver formats text and sets page breaks within an Oracle Reports file.

Sequence of Printing Events - Example

The following is an example of the sequence of printing events.

1. A user submits a request to run a report from the Run Reports form.

2. A request to run the report is added to the requests table.

3. A concurrent manager reads the request.

4. The concurrent manager calls Oracle Reports to run the report, and passes the SRW Driver name. If Report Copies = 0 and the Printer field is blank, the Print Style's SRW Driver is used. If Report Copies > 0 and Printer is required, then the Printer Driver's SRW Driver is used.

   > **Note:** For concurrent requests running Oracle Report concurrent programs, a printer needs to be identified or associated even when the number of copies is set to 0.

   The concurrent manager passes Print Style information (Columns and Rows) to Oracle Reports (overrides SRW Driver width and height if the report is to be printed).

5. A report is created using Oracle Reports. The concurrent manager attaches Printer Driver information to the file. It prepends the initialization string and appends the reset string.

   The concurrent manager also passes suppress header option information from the Printer Styles form.

6. The concurrent manager issues an operating system print command with the

arguments Destination Printer, Filename (including path), Number of Copies to print, and Filename for the Title on the banner page.

Related Topics

Overview of Printers and Printing, page 6-1

Printer Types, Print Styles, and Printer Drivers , page 6-2

Setting Character-Mode vs. Bitmap Printing, page 6-5

Setting up Your Printers, page 6-8

Fast-Track Printing with Pasta , page 6-10

Customizing Printing Support in Oracle E-Business Suite, page 6-31

Postscript Printing in UNIX, page 6-43

Setting Character-Mode vs. Bitmap Printing

Running Character mode Oracle Reports Concurrent Programs

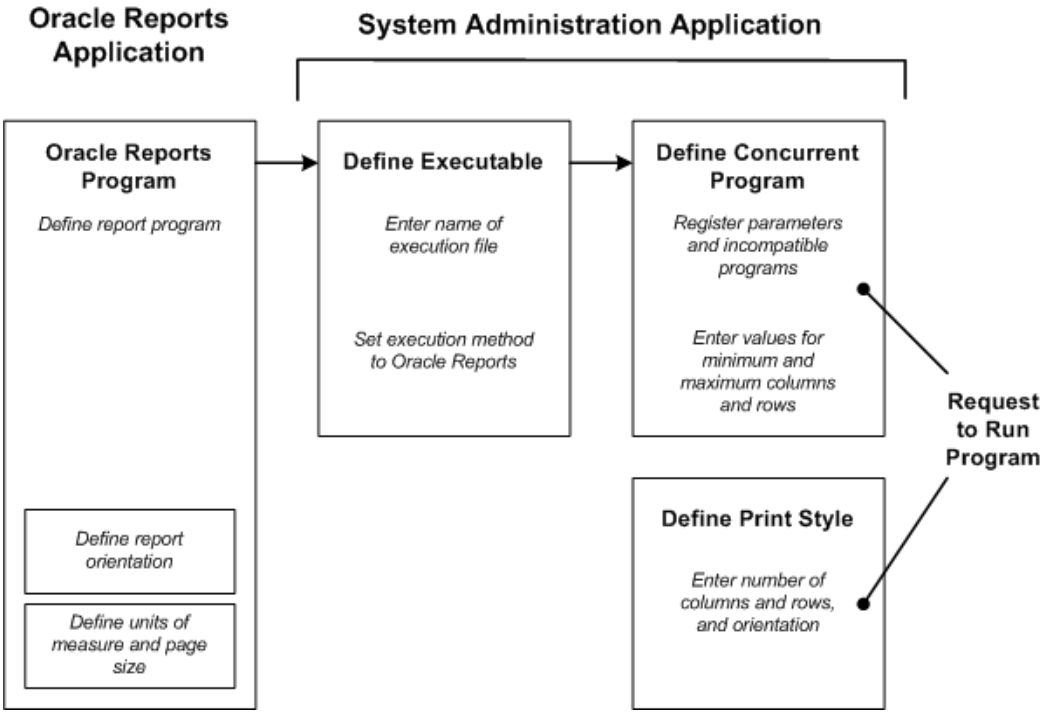
Character mode Oracle Reports programs take their page dimensions and orientation from the print style associated with the request to run the program.

Some print styles are predefined, and a System Administrator can define additional styles, if necessary.

After you create an Oracle Reports program, you create a corresponding concurrent program executable with the Oracle Reports execution method.

You then define a concurrent program for that executable, registering any parameters and incompatible programs. You also enter the minimum column and row length, orientation, and print style.

Running Character Mode Oracle Reports Programs



Running Bitmap Oracle Reports Concurrent Programs

Bitmap Oracle Reports programs are defined similarly in Oracle Reports and in the Concurrent Program Executable form.

To run an Oracle Reports program in bitmap mode, query the concurrent program's definition in the Concurrent Programs form, and choose PostScript in the Format field.

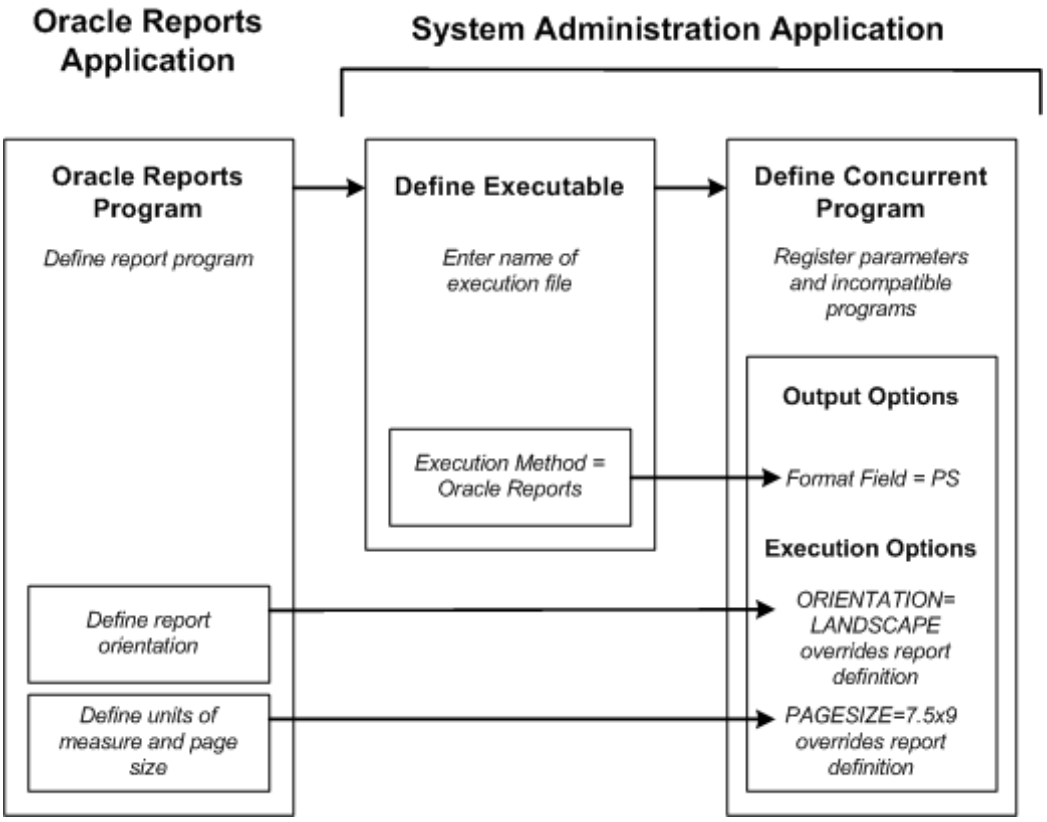
Bitmap Oracle Reports programs take their page dimensions and orientation from the program's definition (note: when printing a bitmap report, a print style is still required).

If you wish to override the program's definitions, you can enter values in the Execution Options field for ORIENTATION and PAGESIZE.

When entering more than one execution option, each option should be separated by a *single* space. There should be no spaces before or after the options. For example:

```
ORIENTATION=LANDSCAPE PAGESIZE=7.5x9
```

Running Bitmap Oracle Reports Programs



Notes about PAGESIZE in the Execution Options field

In Oracle Reports, when defining a report the units and size of the report are specified in the menu under Report->Global Properties->Unit of Measurement.

For bitmapped reports, <width>x<height> for PAGESIZE is usually in inches; however, this depends on the particular report definition.

You can enter the PAGESIZE parameter in the Execution Options field of the Concurrent Programs form (for bitmapped reports only) when you want to override the values specified in the report definition. For example:

```
PAGESIZE=7.5x9
```

If the dimensions specified with the PAGESIZE parameter are smaller than what the report was designed for, you will generate a "REP-1212" error.

Related Topics

Overview of Printers and Printing, page 6-1

Setting up Your Printers, page 6-8

Creating Custom Printer Drivers, page 6-31

Printer Drivers field help, page 6-53

Setting Up Your Printers

Oracle E-Business Suite provides you with predefined printer types, print styles, and printer drivers. Use the Printer Types form to query the combinations of print style and printer driver that support each type of printer you have. Customize the predefined components as necessary. See: Customizing Printing Support in Oracle E-Business Suite, page 6-31.

> **Important:** Predefined printing components may have to be modified for different printer types and/or operating platforms.

Forms for Defining Printer Support

You use four forms to define printer support.

Printer Types

You must define any printer types used at your site that are not shipped with Oracle E-Business Suite. It is on this form that you associate the print style with a printer driver for the particular printer type.

Printers

You register a printer so that Oracle E-Business Suite recognizes the printer and can forward to it the output from a report program.

To register a printer you specify the printer's operating system name, which uniquely identifies the printer, and select the printer type. The printer type must already be defined.

For example, if you want users of Oracle E-Business Suite to be able to print to a newly purchased printer, you:

- Register the operating system name of the new printer (for example, printer39), and select the printer type (for example, LN03).
- If the correct printer type is not defined, you must define the new printer type before you can register the printer.

Print Styles

To generate a report, the print style values for columns and rows are passed by the concurrent manager to Oracle Reports. A print style determines the dimensions of your report, or the number of rows and columns per page.

Printer Drivers

A printer driver includes the initialization and reset strings that format and restart a printer. You need a defined printer driver for each print style that you plan to use with a specific printer type on a specific platform.

Printing Setup Interrelationships

- Many printers can be registered as the same printer type.
- A printer type can support multiple print styles.
- A printer driver must be assigned to a printer type for each print style.
- Many printer drivers can support the same print style.
- Many printer drivers can support the same printer type.

See: Overview of Printers and Printing, page 6-1

Printer Setup Information Is Cached On Demand

Printer setup information such as Printer Type definitions, Print Style definitions, and Printer Driver definitions, are read into memory (cached) the first time the information is required to print a program's output.

The cache area that holds printer setup information is private to the concurrent managers. Printer setup information remains cached in memory until the concurrent managers are restarted, when the values are erased and new values are cached.

> **Important:** You should issue a *Restart concurrent manager* command for all currently active managers whenever you edit an existing Printer Type, Print Style, or Printer Driver (unless the type, style or driver has not been referred to or cached yet).

See: Controlling Concurrent Managers, page 4-134

Related Topics

Overview of Printers and Printing, page 6-1

Printer Types, Print Styles, and Printer Drivers, page 6-2

Customizing Printing Support in Oracle E-Business Suite, page 6-31

Creating Custom Printer Drivers, page 6-31

Printer Setup with Pasta

Overview

Pasta is an Oracle E-Business Suite utility that converts text report files to PostScript and also enables the printing of custom PostScript reports from Oracle E-Business Suite. The reports can then be directed to any PostScript printer.

Setting up your system to use Pasta is much simpler than the standard Oracle E-Business Suite printer setup procedure. The Printer Type, Printer Driver, and SRW driver files are provided. The only setup required to begin printing is the registration of the printer with Oracle E-Business Suite.

Many printing options can be defined using the Pasta configuration file (pasta.cfg). You no longer need to maintain multiple drivers and styles for each printer.

Pasta is provided as an executable named FNDPSTAX.

Related Topics

Setup for Basic Printing with Pasta, page 6-10

Setup for Basic Printing with Pasta

The following setup can be used to enable any PostScript printer to print text or PostScript reports in the following styles: Landscape, Landwide, Portrait, or Dynamic.

Use the Printers window to register your printer:

1. Enter your printer's name as defined in the operating system and applications.
2. Select "--Pasta Universal Printer" from the list of values for the printer Type.

You are now ready to print text and PostScript reports from your PostScript printer using the default Pasta configuration.

For more information on the Printers window, see Printers Window, page 6-49.

For more information on setting options in the Pasta configuration file, see: Configuration File Options, page 6-18.

Related Topics

Defining Configuration Files for Specific Printers, page 6-11

Modify/Add Printer Type to Use Pasta, page 6-12

Setting Margins, page 6-13

Printing a Report Using the noprint Option, page 6-14

Generating Other Formats Using the Preprocessing Option, page 6-15

Font Source, page 6-16

Language-Specific Font Support, page 6-17

Command Line Parameters, page 6-24

Defining Configuration Files for Specific Printers

The pasta.cfg file controls many printing options. You can use the default file for many printers for multiple languages. However, if you have printers that require special setup, you can customize these options by creating a configuration file for each printer.

Copy the pasta.cfg file to pasta_<printer name>.cfg. Make the necessary changes to the file. Pasta automatically looks for a printer-specific file name. If it does not find one, Pasta then uses the default file, pasta.cfg.

For example, suppose you have a printer named "hqprinter" for which you want to set the page height and width to letter size:

1. Copy $FND_TOP/resource/pasta.cfg to $FND_TOP/resource/pasta_hqprinter.cfg where hqprinter is the name of the printer as defined on the operating system.

2. In the pasta_hqprinter.cfg file, edit the paper size options:

   Set pagewidth=8.5

   Set pageheight=11

Pasta will now use the options as defined in the pasta_hqprinter.cfg file when printing to the hqprinter.

For more information on setting options in the Pasta configuration file, see: Configuration File Options, page 6-18.

Using a Different Configuration File as the Default

You can change the file that is defined as the default configuration file for Pasta by using the -F command line parameter.

For example, suppose you create a PCL print-specific configuration file named pcl.cfg. Set the FNDPSTAX command line option as follows:

```
-Fpcl.cfg
```

Pasta will look for pcl_<printer>.cfg first, and if it is not found, Pasta will use pcl.cfg as the default.

These files must be placed under the $FND_TOP/resource directory.

The -F command line parameter can be set in the Arguments field of the Printer Drivers window. See Printer Drivers Window, page 6-53.

Related Topics

Setup for Basic Printing with Pasta, page 6-10

Modify/Add Printer Type to Use Pasta, page 6-12

Setting Margins, page 6-13

Printing a Report Using the noprint Option, page 6-14

Generating Other Formats Using the Preprocessing Option, page 6-15

Font Source, page 6-16

Language-Specific Font Support, page 6-17

Configuration File Options, page 6-18

Command Line Parameters, page 6-24

Modify an Existing Printer Type to Use Pasta

If your printer is already assigned to a printer type that contains styles and drivers that you want to maintain, you can add Pasta to the existing printer type.

To add Pasta to an existing printer type, you must associate one of the Pasta printer drivers with a print style on the Printer Types window.

The seeded Pasta printer drivers are:

- PASTA_LANDSCAPE
- PASTA_PORTRAIT
- PASTA_LANDWIDE
- PASTA_DYNAMIC

You can associate a Pasta driver with an existing print style, or you can create a new print style. To create a new print style, use the Print Styles window. For more information on defining a print style, see Print Styles Window, page 6-50.

1. Query your existing printer type in the Printer Types window.
2. In the Style field, select the style to which you want to assign a Pasta driver.

   Or, if you are assigning Pasta to a style already defined for the printer type, delete the driver in the Driver Name field currently associated with the style.
3. In the Driver Name field, select the appropriate Pasta driver.

   For more information on the Printer Types window, see Printer Types Window, page 6-47.

Add a New Printer Type to Use Pasta

If you want to add a new Printer Type, you can also add Pasta to your new printer type.

1. Navigate to the Printer Types window.
2. Enter the Type of printer.
3. In the Style field use the list of values to select the style to which you want to assign a Pasta driver.
4. In the Driver Name field, select the appropriate Pasta driver from the list of values:
5. PASTA_LANDSCAPE
6. PASTA_PORTRAIT
7. PASTA_LANDWIDE
8. PASTA_DYNAMIC

   For more information on the Printer Types window, see Printer Types Window, page 6-47.

Related Topics

Setup for Basic Printing with Pasta, page 6-10

Defining Configuration Files for Specific Printers, page 6-11

Setting Margins, page 6-13

Printing a Report Using the noprint Option, page 6-14

Generating Other Formats Using the Preprocessing Option, page 6-15

Font Source, page 6-16

Language-Specific Font Support, page 6-17

Configuration File Options, page 6-18

Command Line Parameters, page 6-24

Setting Margins

The margins on your printed output are determined by the margin settings in the pasta.cfg file and the printable area defined by your printer. In order to set your margins properly you must know the printable area for your specific printer and adjust the margin parameter settings in the pasta.cfg file accordingly. The margin parameters are leftMargin, rightMargin, topMargin, and bottomMargin.

For example, suppose you want to set the left margin to one inch. If the printable area for your printer begins at .25 inches from the left, then you must set the leftMargin option to .75 in the pasta.cfg file.

For more information on setting options in the Pasta configuration file, see: Configuration File Options, page 6-18.

Refer to your printer's documentation for information on its printable area.

Related Topics

Setup for Basic Printing with Pasta, page 6-10

Defining Configuration Files for Specific Printers, page 6-11

Modify/Add Printer Type to Use Pasta, page 6-12

Printing a Report Using the noprint Option, page 6-14

Generating Other Formats Using the Preprocessing Option, page 6-15

Font Source, page 6-16

Language-Specific Font Support, page 6-17

Command Line Parameters, page 6-24

Printing a Report Generated Using the noprint Option

When you use the "noprint" printer option to generate a report, you can still have the option of printing it later using Pasta.

Before running the report, associate the noprint printer with the "--Pasta Universal Printer Type" in the Printers form.

> **Note:** You must restart the concurrent manager for this to take effect. See Printer Setup Information is Cached on Demand, page 6-9.

Related Topics

Setup for Basic Printing with Pasta, page 6-10

Defining Configuration Files for Specific Printers, page 6-11

Modify/Add Printer Type to Use Pasta, page 6-12

Setting Margins, page 6-13

Generating Other Formats Using the Preprocessing Option, page 6-15

Font Source, page 6-16

Language-Specific Font Support, page 6-17

Configuration File Options, page 6-18

Command Line Parameters, page 6-24

Generating Other Formats Using the Preprocessing Option

Pasta can use a preprocessing option to invoke any executable that supports an input file and an output file (filter program). Pasta will invoke the filter program to preprocess the Pasta output before passing it to the printing command. By using the preprocessing option, you can generate output formats other than the formats that Pasta currently supports. For example, by invoking products such as Adobe Acrobat Distiller Server or Ghostscript, you can generate PDF output or PCL output.

> **Important:** Ensure that the executable for the preprocess program is placed in your path.

The preprocessing command is a configuration file (pasta.cfg) option. This option uses {infile} and {outfile}.

- {infile} is the file generated by Pasta to be used as input to the preprocessing command. It is a temporary file and will be deleted after it is passed to the preprocessing command.
- {outfile} is the output file generated by the preprocessing command. Pasta names it temporarily and it will be deleted after it is passed to the printing command.

  If you want to keep the {outfile}, you can name it by using the "outFile" pasta configuration file option (see Configuration File Options, page 6-18), or the "-o" command line option (see Command Line Parameters, page 6-24). Pasta will copy {outfile} to the file you specify.

Example for Generating PCL Output

In this example, "gs" is Ghostscript and "pxlmono" is a device used with HP black and white PCL XL printers (Laserjet 5 and 6 family).

In the pasta.cfg file, enter the following for the preprocess option:

```
preprocess=gs -q -dNOPAUSE -dBATCH -sDEVICE=pxlmono
-sOutputFile={outfile} {infile}
```

To get a list of output devices available in Ghostscript go to http://www.gnu.org/software/ghostscript/devices.html

Example for Generating PDF Output

In this example, "ps2pdf" is a shell script bundled with Ghostscript. The ps2pdf script can convert a PostScript file to a PDF file.

In most cases you cannot send a PDF file to the printer command because the printer command cannot understand PDF. Set the noPrint option to "y" or use the "-np" (no print) command line option if you do not want Pasta to send the PDF file to the printer.

Use the outFile option to define the destination on the middle tier for the output file. You can use {inputfile} in the outFile option. Pasta will replace it with the actual input file name (without the path) specified by the "-f" (input file) command line option.

1. Using the basic Pasta setup procedure, define a printer in the Printers window called "PDFfile". Assign the "--Pasta Universal Printer" type to the printer.

   For more information about defining a printer, see: Setup for Basic Printing with Pasta, page 6-10.

2. Create a configuration file for the PDFfile printer called "pasta_PDFfile.cfg".

   For more information about defining a configuration file, see: Defining Configuration Files for Specific Printers, page 6-11.

3. In the pasta_PDFfile.cfg file, enter the following for the preprocess option:

   ```
   preprocess=ps2pdf {infile} {outfile}
   noPrint=y
   outFile=<APPLTMP>/{inputfile}.pdf
   ```

Related Topics

Modify/Add Printer Type to Use Pasta, page 6-12

Setting Margins, page 6-13

Printing a Report Using the noprint Option, page 6-14

Font Source, page 6-16

Language-Specific Font Support, page 6-17

Font Source

If your printer does not have the necessary fonts installed, Pasta embeds the required glyphs as a small font in your report from the font files.

If you do not want to use the fonts provided by Oracle you can specify the font you want by using the font name option in the pasta.cfg file. You can use any TrueType fonts on your middle tier; or, if your printer has a font installed suitable for the language of your report, you can utilize the printer font.

The form of the pasta.cfg option is:

```
Font.Default.<Style>=<TrueType font file name>
```

or

```
Font.Default.<Style>=printer:<Printer font file name>
```

Example using TrueType font file:

```
Font.Default.Plain=<FND_TOP>/<APPLRSC>/ADUO.ttf
Font.Default.Bold=<FND_TOP>/<APPLRSC>/ADUOB.ttf
```

Example using Printer Font:

```
Font.Default.Plain=printer:Courier
Font.Default.Bold=printer:Courier-Bold
```

Related Topics

Setup for Basic Printing with Pasta, page 6-10

Defining Configuration Files for Specific Printers, page 6-11

Modify/Add Printer Type to Use Pasta, page 6-12

Setting Margins, page 6-13

Printing a Report Using the noprint Option, page 6-14

Generating Other Formats Using the Preprocessing Option, page 6-15

Language-Specific Font Support, page 6-17

Configuration File Options, page 6-18

Command Line Parameters, page 6-24

Language-Specific Font Support

The default pasta.cfg file contains font settings for languages. You override the default setting in the language-specific section of the pasta.cfg file.

For example, if you want to override the default setting of Courier font for French language reports to use Helvetica instead, add the following to the end of the pasta.cfg file:

```
[FRENCH]
Font.Default.Plain = printer:Helvetica
Font.Default.Bold = printer:Helvetica-Bold
```

Related Topics

Setup for Basic Printing with Pasta, page 6-10

Defining Configuration Files for Specific Printers, page 6-11

Modify/Add Printer Type to Use Pasta, page 6-12

Setting Margins, page 6-13

Printing a Report Using the noprint Option, page 6-14

Generating Other Formats Using the Preprocessing Option, page 6-15

Font Source, page 6-16

Configuration File Options, page 6-18

Command Line Parameters, page 6-24

Configuration File Options

The configuration file pasta.cfg governs many printing options. The file is a normal ASCII text file that has a simple format described below.

The options are listed in the following two tables. They are divided into Generic Options and Arabic, Hebrew, and Thai options. For each option are listed the Key Name, the Default Value (if applicable), the Description, and the Equivalent Command Line Option (if applicable).

> **Note:** For options that also have command line equivalents, Oracle recommends that you set the value in the configuration file.

Generic Options

Key Name	Default Value	Description	Command Line Equivalent
outputFormat	ps	Two output formats are supported: PostScript ("ps") and text ("text"). When the output format is text, you can specify the output character set by the Oracle character set name (for example, text.WE8ISO8859P1). If you use "auto" as the output character set (text.auto), Pasta uses the appropriate character set according to the NLS_LANGUAGE value in the FND_LANGUAGES table.	-x

Key Name	Default Value	Description	Command Line Equivalent
textAutoCharset	The default value is taken from the FND_LANGUAGES table.	When the outputFormat is set to "text.auto", Pasta uses a default character set for each language based on the language and character set mappings in the FND_LANGUAGES table. To override the default setting for a language, use the textAutoCharset option in the Language-Specific section of the pasta.cfg file. For example, to override the default character set for Japanese to use JA16EUC instead, enter the following: [Japanese] textAutoCharset=JA16SJIS	N/A
preprocess	N/A	Use this option to convert the output file. Enter a preprocessing command to invoke any executable that supports an input file and an output file (filter program). Pasta will invoke the filter program before passing the file to the printing command.	N/A
printCommand	N/A	Specific print command for Unix platform.	N/A

Key Name	Default Value	Description	Command Line Equivalent
ntPrintCommand	N/A	Specific print command for Windows platform.	N/A
outFile	N/A	If you want to save the output file, use this option to define the ouput file name and its destination on the middle tier.	-o
noPrint	N/A	Set this option to "y" if you do not want Pasta to produce printed output.	-np
duplex	default	Specifies duplex printing. Options are "y", "n", or "default" (uses the printer-side setting).	N/A
embednumcopies	y	Set this option to "y" to embed the number of copies to be printed in a PostScript file. Using this option will eliminate the header page normally printed between reports.	N/A
copysort	y	If you set embednumcopies to "y", you can choose to have the copies collated, by setting this option to "y".	N/A
heightScaleRate	1.0	Adjusts the space between lines.	-h

Key Name	Default Value	Description	Command Line Equivalent
widthScaleRate	1.0	Adjusts the space between characters.	-w
pagewidth	8.27	Adjusts the page width.	-pw
pageheight	11.64	Adjusts the page height.	-ph
topMargin	.25	Adjusts the top margin.	N/A
bottomMargin	.25	Adjusts the bottom margin.	N/A
rightMargin	.25	Adjusts the right margin.	N/A
leftMargin	.25	Adjusts the left margin.	N/A
Font.<Face>.<Style>	N/A	Specify the TrueType font file name. The <Face> must be either "Default" or the actual font face name (such as Helvetica). The <Style> must be either "Plain", "Bold", "Italic", or "BoldItalic".	N/A
fontsize	7.8 (for landscape) 10.0 (for portrait)	Font size in points. If this is not set, the font size is calculated automatically.	-s

Key Name	Default Value	Description	Command Line Equivalent
boldfontsize	Size in points (Default value from the fontsize parameter)	Font size for bold font in points. This option is used mostly when Font.Default.Bold is used for specifying a barcode font.	-bs
psEmbeddedFontType	type3 (default) type42	A type of font that will be embedded in PostScript.	
tabsize	8	Pasta replaces a tab with the number of spaces specified in this option.	N/A
errorlogfile	(standard error output)	Set this option to have Pasta create a log file.	-el

Arabic, Hebrew and Thai Options

Key Name	Default Value	Description	Equivalent Command Line Option
thai_space_compensation	n	In the Thai language, some characters are combined into one glyph. If this option is set to "y", Pasta will align your report by adding spaces at the end of the column that includes combined characters.	N/A

Key Name	Default Value	Description	Equivalent Command Line Option
bidi_algorithm	oracle	If you set this option to "unicode", Pasta follows the Unicode BiDi algorithm. Setting the value to "oracle" will use Oracle's algorithm.	N/A
direction	default	Options are "ltr" (left-to-right), "rtl" (right-to-left), and "default" (depends on NLS_LANGUAGE setting).	N/A
dolayout	y	To layout the text, set this option to "y". If not, set it to "n".	N/A
doshaping	y	To shape the text, set this option to "y". If not, set it to "n". This option is for Arabic only.	N/A
numerals	context	Possible values are "arabic" for Arabic numerals, "hindi" for Hindi numerals, or "context" to use Arabic or Hindi depending on the context. Required for Arabic data only.	N/A

Related Topics

Setup for Basic Printing with Pasta, page 6-10

Defining Configuration Files for Specific Printers, page 6-11

Modify/Add Printer Type to Use Pasta, page 6-12

Setting Margins, page 6-13

Printing a Report Using the noprint Option, page 6-14

Generating Other Formats Using the Preprocessing Option, page 6-15

Font Source, page 6-16

Language-Specific Font Support, page 6-17

Command Line Parameters, page 6-24

Command Line Parameters

When using the Pasta utility from the command line, you can use the options below.

```
FNDPSTAX [options]
```

-c<number>	<number> specifies the number of copies to print.
-el<logfile>	Specifies the error log file's path and name. The path is the absolute path to the error log file.
-f<filename>	<filename> specifies the input file name. Example: -fmyfile.txt
-F<cfgfile>	Specifies the configuration file's path and name.
-h<rate>	Adjusts the space between lines. The default value is 1.0. If <rate> is larger than 1, the space between lines will be larger.
-w<rate>	Adjusts the space between characters. The default value is 1.0. If <rate> is larger than 1.0, the space between characters will be larger.
-l	Print in landscape mode. (The default is portrait).
-o<filename>	<filename> specifies the output file name.
-s<size>	Overrides the font size option in pasta.cfg.
-bs<size>	Overrides the font size in -s and fontsize for bold font.
-np	No print option.
-ph<height>	<height> specifies the paper height in inches.
-pw<width>	<width> specifies the paper width in inches.
-pform<psfile><pf file>	Converts a PostScript file <ps file> to a PrintForm file <pf file>.

-pf<pf file>	<pf file> specifies a PrintForm file to be merged in a Pasta output at runtime.
-pn<printername>	<printername> specifies the printer name.
-t<banner title>	Banner option for use with the Unix lp command.
-v	Displays the version number.
-x<ps|text.[charset|auto]>	Specifies the output format. Two output formats are supported: PostScript ("ps") and text ("text").

Related Topics

Defining Configuration Files for Specific Printers, page 6-11

Modify/Add Printer Type to Use Pasta, page 6-12

Setting Margins, page 6-13

Printing a Report Using the noprint Option, page 6-14

Generating Other Formats Using the Preprocessing Option, page 6-15

Font Source, page 6-16

Language-Specific Font Support, page 6-17

Configuration File Options, page 6-18

Using PrintForms

PrintForms is a feature of the E-Business Suite that allows you to create your own background template for a report and use it at print-time either as a background image (such as your company logo) or as a standard form (such as an invoice).

The PrintForm process combines the input report file and the background template file to generate one PostScript file, which is then passed to a PostScript printer.

Because this process is executed at print-time, the need to customize report definitions is eliminated, which also simplifies maintenance. Moreover, using the PrintForm as a standard form can replace printing solutions that require expensive preprinted stationery and specialized printers.

There are two phases to using PrintForms:

1. Create the PrintForm

2. Deploy the PrintForm to Oracle E-Business Suite

Note: The following instructions are general guidelines for creating

and using PrintForms. For more detailed instructions see the "Pasta User's Guide Release 3.0" available in Knowledge Document 239196.1 on My Oracle Support.

Create the PrintForm

Creating a PrintForm is a five-step process:

1. Create an image file using the supported editor of your choice.
2. Convert the image file to a PostScript file (.ps).
3. Convert the PostScript file to a PrintForm file (.pf) to use as your template.
4. Place the PrintForm file in the appropriate directory.
5. Update printer driver information (in the Printer Drivers window) to use the PrintForm.

Create an image file

The following editors can be used to create the image file that will become your PrintForm template.

- Oracle Reports
- Microsoft Word, Excel, or Powerpoint (with Adobe PDFMaker)
- Any Windows Application
- Any Application that generates PDF

When creating your image file ensure that the paper size, margin settings, and style of the document are appropriate for the report you are fitting the template to.

> **Note:** Even if you need only one type of image (such as a company logo added to all your reports), you must create one PrintForm for each print style that you plan to use (such as portrait, landscape, and landwide) with the appropriate placement of the image for the style.

Convert the image file to a PostScript file

The PrintForm template file must be created from a PostScript file. The PostScript file can be generated in one of the following ways:

1. Generate a PostScript file from Oracle Reports.

2. Convert any Windows application file (such as Microsoft Word or PowerPoint) to PostScript using the Adobe Generic Printer Driver "save to file" option.

3. Convert any PDF file to PostScript using a third-party utility, such as Xpdf (available for Windows and Unix platforms).

   Note: For information about downloading and using Xpdf see http://www.foolabs.com/xpdf/

Convert the PostScript file to a PrintForm file

Use a Pasta command line parameter to convert the PostScript file to a PrintForm (.pf) file. During this process comments are added to the PostScript file.

The syntax for the command line is as follows:

```
FNDPSTAX -pform <PostScript template file>.ps <PrintForm file>.pf
```

For more information about Pasta command line parameters, see Command Line Parameters, page 6-24.

Deploying the PrintForm to Oracle E-Business Suite consists of the following steps:

1. Place the PrintForm in the APPL_TOP.

2. Update the printer driver information.

3. Restart the Concurrent Manager.

Place the PrintForm in the appropriate directory

You can place the PrintForm file anywhere under APPL_TOP in the Concurrent Processing node, but Oracle recommends that you put the file under the FND_TOP/resource/<lang> directory, where <lang> is the language code (such as US for English or KO for Korean).

Update the printer driver information

1. Navigate to the Printer Drivers form.

2. Query up the printer driver to which you want to add the PrintForm option.

3. Edit the Arguments field for the printer driver to use the PrintForm in the appropriate directory.

   For example, if you place a PrintForm called logo_ls.pf in the $FND_TOP/resource/US directory and you want to apply the PrintForm to the PASTA_LANDSCAPE driver, edit the Arguments field as follows:

```
-pn$PROFILES$.PRINTER -f$PROFILES$.FILENAME -c$PROFILES$.CONC_COPIES -l
-pf<FND_TOP>/resource/US/logo_ls.pf
```

where <FND_TOP> is the actual path for your $FND_TOP.

For more information about this form, see Printer Drivers Window, page 6-53.

Restart the Concurrent Manager

You must restart the concurrent manager for this to take effect. See Printer Setup Information is Cached on Demand, page 6-9.

Common UNIX Printing System (CUPS) Integration

CUPS is a third-party product that provides printing service to most PostScript printers and raster printers. It is based on the Internet Printing Protocol (IPP) technology making it open to any system that supports IPP. CUPS can run on various kinds of Unix systems, such as Solaris, HP-UX, AIX, Tru64, and major Linux distributions. CUPS is provided under the GNU GPL license.

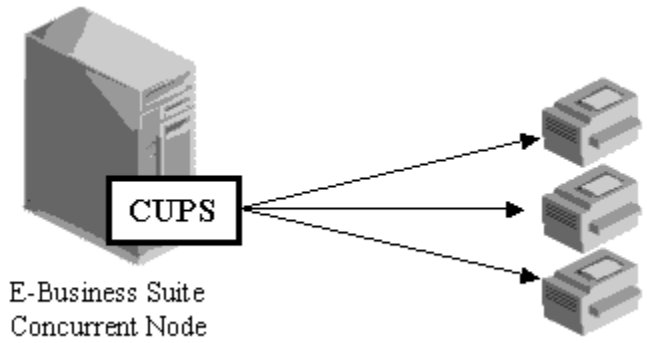
For more details about CUPS, refer to http://www.cups.org/

Two integration scenarios are described in the following sections:

- All printers connected to a single CUPS server residing on the E-Business Suite concurrent processing node
- Multiple CUPS distributed servers with printers connected to each CUPS server

Scenario 1: All printers connected to a single CUPS server residing on the E-Business Suite concurrent processing node

This scenario is graphically depicted in the following diagram:



There are no special configuration steps required if the printer system has been replaced with CUPS on the server.

If you maintain both the System V printing system and CUPS, you must define a new

printer type with the required print styles and printer drivers.

1. Create a new Pasta configuration file for CUPS in the $FND_TOP/resource directory (example: pasta_cups.cfg).

2. Ensure that the lp command that is set for the printCommand parameter in the Pasta configuration file is a CUPS lp command. You can specify the full path to the command if you maintain both the UNIX standard lp command and the CUPS lp command.

3. Create a set of printer drivers for each print style with the configuration file created in Step 1.

   Example:

   Printer Driver Name: PASTA_LANDSCAPE_CUPS

   Arguments:

   ```
   -pn$PROFILES$.PRINTER -f$PROFILES$.FILENAME -cPROFILES$.CONC_COPIES
   -l -Fpasta_cups.cfg
   ```

   You can refer to the existing printer driver definitions created by Oracle, such as PASTA_LANDSCAPE, PASTA_PORTRAIT, or PASTA_LANDWIDE.

4. Create a printer type for the CUPS printer.

   Example:

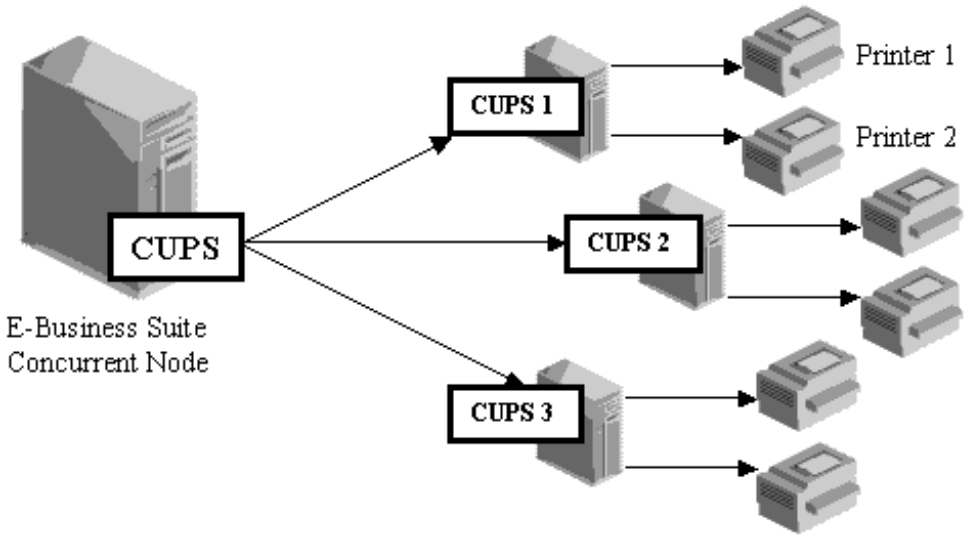
   Printer Type: Pasta printer for CUPS

Print Style	Printer Driver
LANDSCAPE	PASTA_LANDSCAPE_CUPS
PORTRAIT	PASTA_PORTRAIT_CUPS
LANDWIDE	PASTA_LANDWIDE_CUPS

5. Register your printer with the print style.

Scenario 2: Multiple CUPS distributed servers with printers connected to each CUPS server

The following figure displays multiple CUPS distributed servers with multiple printers attached to each:



In this scenario, one printer type is required for each server on which a CUPS process is running with a physical printer.

1. Create a Pasta configuration file in the $FND_TOP/resource directory for each server on which the CUPS process is running .

   For example: pasta_cups1.cfg, pasta_cups2.cfg, pasta_cups3.cfg

2. For each configuration file, make the following change:

   ```
   printCommand=lp -h<cupshost> -d{printername}
   ```

   where <cupshost> is replaced with the actual server name.

   Make sure that the lp command is a CUPS lp command. You can specify the full path to the command if you maintain both the UNIX standard lp command and the CUPS lp command.

   Leave {printername} as is. This parameter will be replaced by Pasta at runtime with the actual printer name.

3. Create a set of printer drivers for each print style with the configuration file created in Step 1.

   Example:

   Printer Driver Name: PASTA_LANDSCAPE_CUPS1

   Arguments:

   ```
   -pn$PROFILES$.PRINTER -f$PROFILES$.FILENAME -cPROFILES$.CONC_COPIES
   -l -Fpasta_cups1.cfg
   ```

   You can refer to the existing printer driver definitions created by Oracle, such as PASTA_LANDSCAPE, PASTA_PORTRAIT, or PASTA_LANDWIDE.

4. Create a printer type for each server in Oracle E-Business Suite.

   For example: Pasta printer CUPS1, Pasta printer CUPS2, Pasta printer CUPS3

   Add the required print styles associated with the printer drivers to the printer type.

   Example:

   Printer Type: Pasta printer for CUPS1

Print Style	Printer Driver
LANDSCAPE	PASTA_LANDSCAPE_CUPS1
PORTRAIT	PASTA_PORTRAIT_CUPS1
LANDWIDE	PASTA_LANDWIDE_CUPS1

1. Register your printer with the print style.

   For example, if printer1 is defined on the server CUPS1 and you have created Pasta printer CUPS1 printer type for the server, register printer1 with the Pasta printer CUPS1 printer type.

Related Topics

Printers Window, page 6-49

Printer Types Window, page 6-47

Print Styles Window, page 6-50

Printer Drivers Window, page 6-53

Defining Configuration Files for Specific Printers, page 6-11

Configuration File Options, page 6-18

Customizing Printing Support in Oracle E-Business Suite

Oracle E-Business Suite provides numerous predefined printer types with which you can identify your printers, as well as print styles that define the dimensions of Oracle Reports output files, and printer drivers that instruct the various printer types how to output the selected print style.

Use the Printer Types form to query the combinations of print style and printer driver that support each type of printer you have.

Important: Predefined printing components may have to be modified for different printer types and/or operating platforms.

For example, if a blank page is being printed after each printed page, the number of rows defined for the print style may need to be reduced, or an escape sequence that is being interpreted differently, creating a page eject command, may have to be rewritten.

Verify and, if necessary, Customize Printer Driver Definitions

Upon installation for any printer type you are using, verify your printer driver definitions, particularly the following:

- Initialization string

  Print a short report to verify the page's printing orientation. If you want to change the printer's default font for the report, you would include that information in the Initialization string.

- Reset string

  Print two short reports with different printing orientations, for example, one that is landscape and another that is portrait, to verify the printer is resetting itself properly.

- Arguments

  Print a short report to verify the arguments to the operating system's print command or a custom print program are being interpreted correctly.

If you need to define a new print style, verify the printer driver you assign to the new print style, for any printer type you use.

Verify and, if necessary, Customize Oracle Reports SRW Drivers

If you have a printer type that does not properly interpret the control characters in the SRW driver files that set page breaks, bold on/off and underline on/off attributes in your Oracle Reports files, you can copy the SRW driver file and modify it.

Creating Custom Printer Drivers

If necessary, edit the Initialization string and the Reset string for the printer type you are using. Refer to your printer's user guide for instructions. The Initialization and Reset fields appear on the Printer Drivers form.

Edit your Initialization string or Reset string if:

- Your printer type requires different control characters.

- The control characters have a different meaning due to your operating system and platform.
- Language translation changes the meaning of the control characters. The printer needs special control characters to select different character sets.
- You want to change the printer's default font for the report (Initialization string only).

Printer Driver Methods

There are three methods to invoke a printer driver:

Command

The concurrent manager can issue an operating system print command and its arguments.

An operating system print command, along with all its arguments, is entered in the Arguments field of the Printer Drivers form.

Program

The concurrent manager can call a custom print program and pass arguments to the program.

The name of a custom print program is entered in the Program Name field and any arguments to be passed to the program are entered in the Arguments field of the Printer Drivers form.

Subroutine

The concurrent manager can call a predefined Oracle E-Business Suite subroutine that passes a print command and arguments to the printer via the operating system.

A subroutine is predefined by Oracle E-Business Suite, and the name is entered in the Program Name field of the Printer Drivers form.

The arguments field is disregarded when the driver method is *Subroutine*. However, the concurrent manager reads the Initialization and Reset escape sequences.

On UNIX systems, the subroutine method, unlike the command method, does not start an operating system shell along with the print command.

Example - Using the Program Driver Method

The Program driver method allows customers to define their own custom print programs. For example, your company might want to write a custom program that opens a file, allows the file to be edited and saved under a second filename, then sends the second (edited) file on to the printer by issuing the print command. This method of

issuing print commands is called a *filter*.

Location for Custom Print Programs

To call a custom print program using the Printer Drivers form, the program name, including the full pathto the program, should be entered in the Program Name field.

The path to the program name is not necessary if the program's location can be identified by the operating system's PATH environment variable (i.e., is in the $PATH variable name).

For platforms where the equivalent of a $PATH variable doesn't exist, then use the full path name. A path can be up to 255 characters.

Custom print programs are not registered as concurrent programs with Oracle Application Object Library, but are called after the concurrent process has completed.

Using Operating System Shell Scripts

For operating system shell scripts, the printer driver method can be either command or program, as long as you populate the argument field correctly.

The script for a command shell procedure should reside in:

```
$FND_TOP/$APPLBIN.
```

Arguments That a Concurrent Manager Can Supply Values For

The concurrent manager can supply four different values as arguments to the operating system print command it issues, or to a custom print program that it calls. An example of using these values as arguments follows.

Example - Entering a Print Command and Arguments

In this example, the UNIX print command "lp" is entered along with the arguments that a concurrent manager can supply values for. While print commands vary, the tokens for which values are retrieved are always the same.

Because print commands are operating system dependent, please refer to *Oracle E-Business Suite Installation Guide: Using Rapid Install* .

Example - Printer Drivers Form's Arguments field:

```
lp -d$PROFILES$.PRINTER -n$PROFILES$.CONC_COPIES -t"$PROFILES$.TITLE"
$PROFILES$.FILENAME
```

The following table lists arguments and their contents for the UNIX lp print command:

Argument Syntax	Token and Value Retrieved
-d$PROFILES$.PRINTER **-d** calls out the destination printer.	**$PROFILES$.PRINTER** retrieves the operating system name of the printer associated with the request.
-n$PROFILES$.CONC_COPIES **-n** calls out the number of copies to print.	**$PROFILES$.CONC_COPIES** retrieves the value of the profile option *Concurrent:Report Copies*, unless this value is updated at runtime.
-t"$PROFILES$.TITLE" **-t** calls out the report title to print on a banner or header page.	"**$PROFILES$.TITLE**" retrieves the title of the output file, typically titled as *Application username.Request ID*. For example, if user John Smith ran a report whose concurrent request ID was 64225, the title would be JSMITH.64225. This is operating system dependent.
$PROFILES$.FILENAME	**$PROFILES$.FILENAME** calls out the filename of the report to be printed. The value retrieved is the output file name, including the path to the file. Note that this file is a temporary file created from information from the Printer Driver definition (from the Printer Drivers window) and the actual output for the report.

In addition, the following table lists arguments and their contents in the cases of an original print request and reprint requests:

Argument Syntax	Original Print Request Value Retrieved	Reprint Value Retrieved
$PROFILES$.ORIGREQID	Request ID.	Original request ID.
$PROFILES$.ORIGUSERNAME	User name.	Original user name.
$PROFILES$.REPREQID	0 (zero)	Reprint request ID.
$PROFILES$.REPUSERNAME	NULL	Reprint request user name.

Argument Syntax	Original Print Request Value Retrieved	Reprint Value Retrieved
$PROFILES$.OUTFILENAME	Output file path and filename.	Output file path and filename. Note that the file here contains only the report output.
$PROFILES$.OUTFILEHOST	Host name.	Host name.

Using Standard Input

When Standard Input is set to Yes, the printer driver accepts standard input, so you can feed a report's output directly to the printer from standard input. Two examples of using standard input are:

- when you run a pipe in UNIX such as "cat myfile | lpr" rather than "lpr myfile", the output file is sent to the stdin (standard input).
- the UNIX command lpr, which accepts standard input when a filename is not specified.

The Standard Input field should be set to No when the Driver Method is set to Program or Subroutine. Unless the program accepts standard input, the Standard Input field should always be set to No.

> **Important:** When Standard Input is set to No, the print command issued by the concurrent manager runs asynchronously. That is, the concurrent manager issues the command, and does not wait for an operating system response.

Using Initialization and Reset Strings

Use the initialization and reset strings to set and reset the orientation, character set and line density for your printer.

Initialization and reset strings consist of control characters and escape sequences.

- A control character can be represented by " ^ " followed by another character.
- An escape sequence can be identified by either " /e " or " \e ".

> **Important:** You see "/e" for escape sequences defined using the

Printer Drivers form (because you cannot enter the backslash (\ ) character into a form when your terminal definition uses backslash as the [Menu] key). You see "\e" for escape sequences originally defined in .pdf files that were later upgraded in Oracle E-Business Suite printer drivers.

For nonprintable characters, you may represent their value in octal mode. For example, 0x26 is represented as " /046 ". As an example, if you need to represent the escape sequence:

^ [^ L ^ [l 6 D (0 x 26)

you can represent it as:

/ e ^ L / e l 6 D / 0 4 6

Using a Spool File

When Spool File is set to No, then a temporary file is created where the initialization and reset strings are inserted, and the file is sent to the print command or program.

Set the Spool File to Yes only if the print program creates its own temporary file. This option is recommended when using the Program driver method and the print program creates its own temporary file.

This option helps to reduce the creation of temporary files, since the concurrent manager will not create a temporary file when Spool File is set to Yes.

When Spool File is set to Yes, it is recommended that the:

- Standard Input be set to No
- Initialization and reset fields are null.

This option does not apply to the Subroutine driver method.

Creating Custom SRW Drivers

SRW drivers are read by Oracle Reports when a report is generated, and insert control characters that tell the destination printer where to set page breaks, and which characters to format as bold or underlined.

SRW drivers only pertain to Oracle Reports output files. An SRW driver is used during the generation of a report. A printer driver is used when the completed output file is sent to the printer.

SRW drivers are designed for the DEC LN03 printer, and all printers that understand the same control characters as the LN03.

Location and Content of SRW Driver Files

SRW driver files reside in $FND_TOP/$APPLREP, and have the file extension ".prt". The predefined SRW file names are:

- A.prt
- P.prt
- L.prt
- PD.prt
- W.prt

Creating a Custom SRW Driver

You can customize any of the SRW driver files to support a printer type that is not correctly interpreting the control characters used to set page breaks and format text as bold or underlined in Oracle Reports files.

For example, you may need to change the control characters that instruct the printer to set a page break.

```
on an LN03 on an XYZ LaserInk
new page ... ^L  ^[E
```

If you need to change formatting control characters for page breaks, underlined text, or bold text in Oracle Reports:

- Copy the .prt file (SRW driver) and rename the copy.
- Modify the new file with new control characters.
- Place the modified copy of the SRW driver file in $FND_TOP/$APPLREP.
- Associate the new driver with a print style and/or printer driver definition.

> **Important:** Copy the SRW driver (.prt file) and rename it before starting any text editing.

SRW Drivers - Print Styles and Printer Drivers

When the concurrent manager calls Oracle Reports to run a report, the SRW driver name is passed as a parameter to Oracle Reports.

The SRW driver is not required because some customers might be using styles or printer drivers for non-Oracle Reports programs.

The SRW driver name you enter in the Print Styles and Printer Drivers forms is used in slightly different ways depending on whether you are printing or simply viewing the report.

If you run an Oracle Reports program without printing the output file, the SRW driver associated with the report's print style is used.

If you run an Oracle Reports program and print the output file, the SRW driver that is correct for the destination printer type is chosen by selecting the SRW driver associated with the printer driver.

Related Topics

Overview of Printers and Printing, page 6-1

Creating Custom Printer Drivers, page 6-31

Print Styles field help, page 6-50

Printer Drivers field help, page 6-53

Hierarchy of Printer and Print Style Assignments

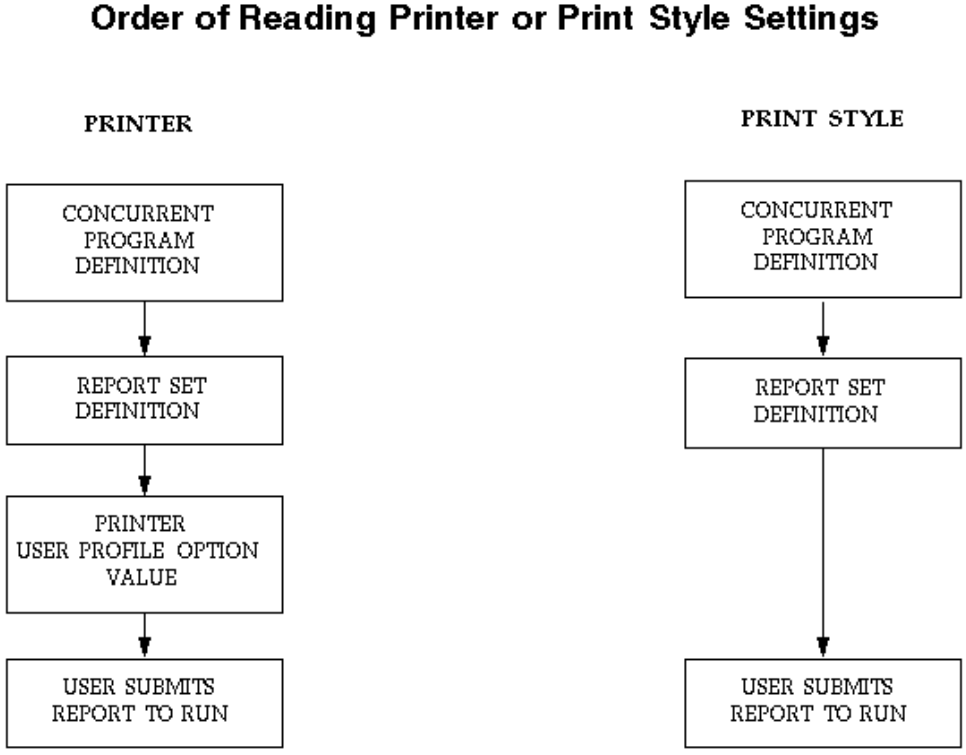
A printer and a print style can be chosen and their identities can be included in a concurrent program's definition. When a concurrent program is defined to send its output to a specific printer, or is required to generate its output in a specific print style, those values cannot be overridden by users, or by report set default settings, or by user profile default settings.

Often, a default value can be set in more than one way. This leads to a hierarchical relationship among the various default settings, where one default takes precedence over another. The diagram below illustrates the order of how printer or print style values are read by the concurrent manager when submitting a report program to run.

> **Important:** Defining a concurrent program with a default print style, or requiring a concurrent program to output a specific print style, does not make that style available at a printer. You must assign the print style, and its corresponding printer driver, to each printer type you wish to print from.

The concurrent manager reads the printer value using the following hierarchy: concurrent program definition, report set definition, printer user profile option value, and value specified by user during report submission.

The concurrent manager reads the print style value using the following hierarchy: concurrent program definition, report set definition, and value specified by user during report submission.



Hierarchy of Printer Assignments

As System Administrator, you can restrict concurrent programs and reports to direct their output to a specific printer. Restricting a program or report's output to a specific printer overrides user profile option settings and prevents report set or user runtime printer choices.

If a printer is not included as part of a concurrent program's definition, then default printer settings may be entered, as indicated in the table below. Users can override any default setting at runtime.

The following table describes the printer assignment hierarchy:

Form	Explanation
Concurrent Programs *System Administrator*	As System Administrator, you can define a concurrent program to always direct its output to only one specific printer. This setting cannot be overridden at runtime or when defining a report in a report set.

Form	Explanation
Request Set *System Administrator*	As System Administrator, you can assign a default printer to a report within a report set.
Request Set *Application Users*	Users can assign a default printer to a report within a report set, when they own the report set. This default setting can be changed by the System Administrator.
Personal Profile Values *Application Users*	Users can assign a default printer for all their reports using their Personal Profile Values form. This assignment overrides the default Printer profile option set by the System Administrator.
System Profile Values *System Administrator*	As System Administrator, you can assign a default printer to an installation site, Oracle application, responsibility, or user. Users can override this setting at runtime.

Hierarchy of Print Style Assignments

As System Administrator, you can require concurrent programs and reports to generate their output in a specific print style. Requiring a program's or a report's output to be in a specific print style prevents report set or user runtime print style choices.

Requirements for alternate print styles

All concurrent programs whose execution method is "Oracle Reports" require a print style to be selected when the program is defined. When the print style is not designated as a *required* print style, then other print styles may be selected, either as a default for a report in a report set, or at runtime when submitting the report, if two conditions are satisfied:

- The print style complies with the concurrent program's minimum values for columns and rows (entered on the Concurrent Programs form).
- The print style has been assigned to the destination printer's printer type (entered on the Printer Types form).

The following table describes the print style assignment hierarchy:

Form	Explanation
Concurrent Programs *System Administrator*	As System Administrator, you can require a concurrent program to generate its output in a specific print style. This setting cannot be overridden at runtime or when defining a report in a report set. If a Print Style is entered in a program definition, but is not required, it serves as the first default setting to be read.
Request Set *System Administrator*	As System Administrator, you can assign a default print style to a report within a report set.
Request Set *Application Users*	Users can assign a default print style to a report within a report set, when they own the report set. This default setting can be changed by the System Administrator.

System Administrator Printer and Print Style Settings

Program Definitions, Printers and Print Styles

As System Administrator you can restrict programs to send their output files only to a specified printer, for example, a printer in a secure office, using the Concurrent Programs form. You can also require a report to generate its output in a specific print style.

Assigning Default Printers and Print Styles to Reports in a Set

As System Administrator you can identify a default printer for each report within a report set, and assign a default print style for each report, using the Request Set form.

Assigning Default Printers Using Profile Options

As System Administrator you can identify a printer as a *default printer* for your installation site, a specific Oracle Application, a specific responsibility, or any of your end users, by setting the "Printer" user profile option in the System Profile Values window.

Users can override a default profile option value by:

- Setting their own personal "Printer" profile option using their Personal Profile Values form.
- Selecting another (available) printer at runtime when submitting a report.

End User Printer and Print Style Settings

End users may:

- Set default print styles for reports in their report sets, using their Request Set form.
- Identify a default printer of their own by using the Personal Profile Values form.

  Users may override the default profile option setting their System Administrator defines.
- Choose any available printer and print style when running reports, when using the Run Reports form.

  If a default printer or print style displays, users may override the default if other printers or print styles are available.

Related Topics

Overview of Printers and Printing, page 6-1

Printer Types, Print Styles, and Printer Drivers, page 6-2

Setting up Your Printers, page 6-8

Printer Setup with Pasta: , page 6-10

Printers, page 6-49

Print Styles, page 6-50

Printer Drivers, page 6-53

Postscript Printing in UNIX

You can convert your report output files into postscript format when printing in some UNIX environments by using the *enscript* UNIX utility.

> **Important:** Refer to your UNIX documentation before using enscript. Usage and the arguments employed by enscript may be specific to your platform.

Concurrent Manager Arguments

The concurrent manager can supply four different values as arguments to an operating system print command or custom print program. See the example of using all four values provided by the concurrent manager. See: Example - Entering a Print Command and Arguments, page 6-34

See the example of using the enscript UNIX utility and two of the values the concurrent manager supplies as arguments. See: Example - Using the UNIX Enscript Command, page 6-44.

Enscript Arguments and Print Styles

The following table lists some sample enscript arguments, using the Courier font, for converting a report's output into postscript for the portrait, landscape, landwide, and A4 print styles.

Print Style	Enscript Arguments	Explanation	Result
Portrait	-fCourier10	Font is Courier 10 point.	80 characters portrait
Landscape	-r -fCourier8	**-r** rotates the printer's output 90 degrees to print in landscape mode. Font is Courier 8 point.	132 characters landscape
Landwide	-r -fCourier6	**-r** rotates the printer's output 90 degrees to print in landscape mode. Font is Courier 6 point.	180 characters landscape
A4	-fCourier10	Font is Courier 10 point.	132 characters landscape (A4 paper)

Example - Using Enscript to Print Postscript

In this example, the enscript command, followed by its arguments, is entered in the Arguments field of the Printer Drivers window, and the Driver Method is set to *Command*.

Printer Drivers window Arguments field:

```
enscript -r -fCourier8 -B -P$PROFILES$.PRINTER $PROFILES$.FILENAME
```

The following table explains the syntax for the enscript command.

Syntax	Explanation
-r	Enscript argument. Rotates the printer's output 90 degrees to print in landscape mode.
-fCourier8	Enscript argument. **-f** selects the font, in this example the font is Courier with a point size of 8.
-B	Enscript argument. Omits page headings.
-P$PROFILES$.PRINTER	Enscript argument. **-P** precedes the name of the printer which the output is sent to. Concurrent manager token. **$PROFILES$.PRINTER** retrieves the operating system name of the printer associated with the request.
$PROFILES$.FILENAME	Concurrent manager token. **$PROFILES$.FILENAME** calls out the filename of the report to be printed. The value retrieved is the output file name, including the path to the file. Note that this file is a temporary file created from information from the Printer Driver definition (from the Printer Drivers window) and the actual output for the report.

In this example, the UNIX enscript command is entered along with two arguments that a concurrent manager can supply values for.

- Since the argument "$PROFILES$.CONC_COPIES" is not used, the number of copies to be printed is set by the enscript default (which is usually one).
- Since the argument "$PROFILES$.TITLE" is not used, the concurrent manager does not provide a value for printing the report title on a banner or header page.

The following table lists additional arguments and their contents in the cases of an original print request and reprint requests:

Argument Syntax	Original Print Request Value Retrieved	Reprint Value Retrieved
$PROFILES$.ORIGREQID	Request ID.	Original request ID.
$PROFILES$.ORIGUSERNAME	User name.	Original user name.
$PROFILES$.REPREQID	0 (zero)	Reprint request ID.
$PROFILES$.REPUSERNAME	NULL	Reprint request user name.
$PROFILES$.OUTFILENAME	Output file path and filename.	Output file path and filename. Note that the file here contains only the report output.
$PROFILES$.OUTFILEHOST	Host name.	Host name.

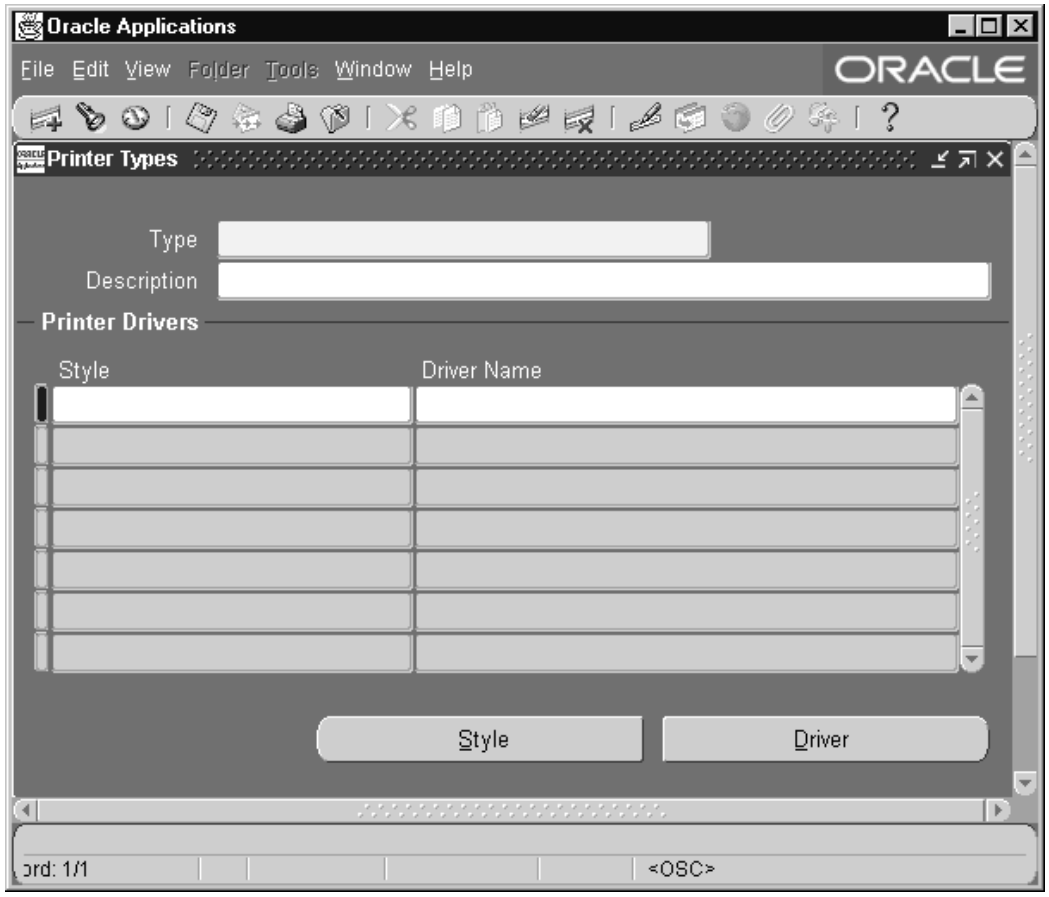
Related Topics

Overview of Printers and Printing, page 6-1

Setting up Your Printers, page 6-8

Hierarchy of Printer and Print Style Assignments, page 6-39

Printer Types Window



Use this window to define a printer type and to assign print styles and their corresponding printer drivers to the printer type.

Defining printer types allows you to assign print style and printer driver definitions to any number of printers by registering the printers as a specific "type".

When users choose a printer to send a report to, the available print styles are normally determined by the printer type.

Concurrent programs, however, can be defined to require their report output in a specific print style. For example, some Oracle Reports programs may require a specific print style in order to print correctly.

> **Important:** You should issue a *Restart concurrent manager* command for all currently active managers whenever you edit an existing Printer Type, Print Style, or Printer Driver.

See: Controlling Concurrent Managers, page 4-134

Printer Types Block

Type

Enter a name for a printer type. Example printer types might be "LINE" for a line printer or "LN03" for an LN03 model printer.

You select this printer type when you register a printer using the Printers window.

Printer Drivers Block

Use this block to assign print styles and printer drivers to your printer types.

The Style button opens the Printer Styles window.

The Driver button opens the Printer Drivers window.

Related Topics

Overview of Printers and Printing, page 6-1

Printer Types, Print Styles, and Printer Drivers , page 6-2

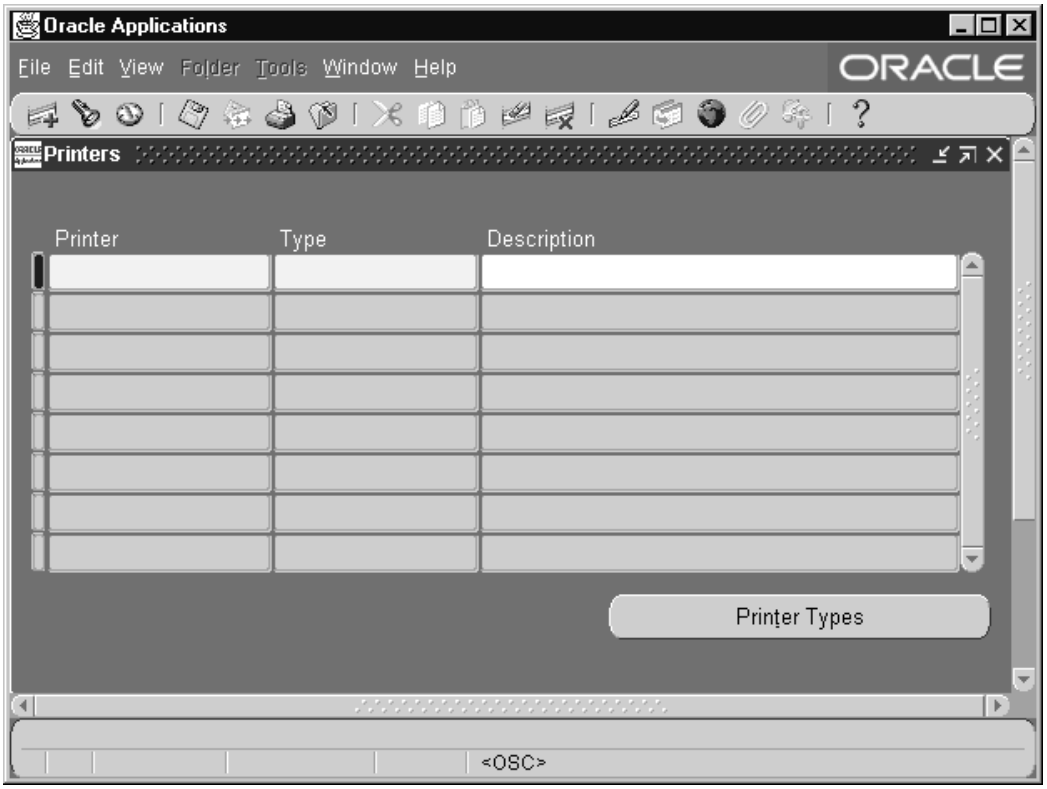
Setting up Your Printers, page 6-8

Printers, page 6-49

Print Styles, page 6-50

Printer Drivers, page 6-53

Printers Window



- Register printers with Oracle E-Business Suite by entering the operating system's name for the printer and assigning it a printer type.
- You must register a printer before you can print reports from it, using Oracle E-Business Suite.
- You can only register a printer with a previously defined printer type. Use the Printer Types window to define printer types.
- You can specify the default printer to which a user submits reports by setting the " Printer" user profile option.

Printers Block

Printer

Enter the name your operating system specifies for the printer.

Type

Select your printer type (i.e., manufacturer and model). Some reports require a printer of a specific type in order to print correctly.

You can only select a previously defined printer type. Use the Printer Types button to open a window to define a printer type.

Related Topics

Overview of Printers and Printing, page 6-1

Printer Types, Print Styles, and Printer Drivers, page 6-2

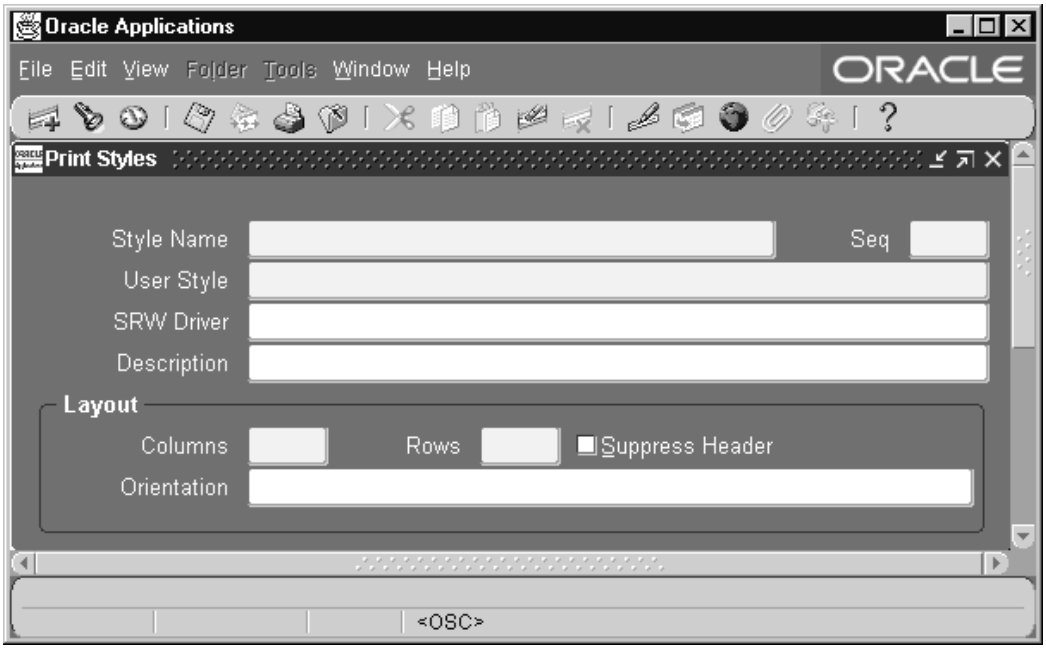
Setting up Your Printers, page 6-8

Printer Types field help, page 6-47

Print Styles field help, page 6-50

Printer Drivers field help, page 6-53

Print Styles Window



Use this window to define print styles. A print style describes how your report should be printed. For example, print style determines the:

- Number of lines per page
- Width of each line

- Page orientation (e.g., portrait or landscape)

Oracle E-Business Suite reports are designed to work with standard, shipped print styles. The following print styles are predefined:

- Portrait
- Landscape
- Landwide
- A4
- Dynamic Portrait

Not all reports work with all print styles. You may define additional print styles to customize your reports.

Once defined, a print style cannot be deleted.

Print Styles Block

Define a print style. The combination of Name and User Name uniquely identifies a print style.

> **Important:** You should issue a *Restart concurrent manager* command for all currently active managers whenever you edit an existing Printer Type, Print Style, or Printer Driver.

See: Controlling Concurrent Managers, page 4-134.

Style Name

Multiple print styles display alphabetically in a list window.

You cannot update a print style's name.

Sequence

Enter a number that determines the display sequence for your print style when performing a query in this window. A negative sequence number appears before zero, and zero appears before a positive sequence number.

User Style

This user name does not appear anywhere except this window.

SRW Driver

Enter the name of the Oracle Reports (SRW) driver to be called when printing an applications report generated by Oracle Reports. This field is used only by applications reports generated by Oracle Reports.

Layout Block

Columns

Enter the number of columns your print style defines.

Rows

Enter the number of rows your print style defines.

Suppress Header

Reports may print with a header page that indicates who requested the report and when. Check the Supress Header check box to define a print style that suppresses printing of this header page.

For example, suppressing the header page when printing checks prevents a check from being overwritten and maintains the orderly sequence of check numbers.

Orientation

Enter the orientation of your printed page, for example, portrait or landscape.

Related Topics

Overview of Printers and Printing, page 6-1

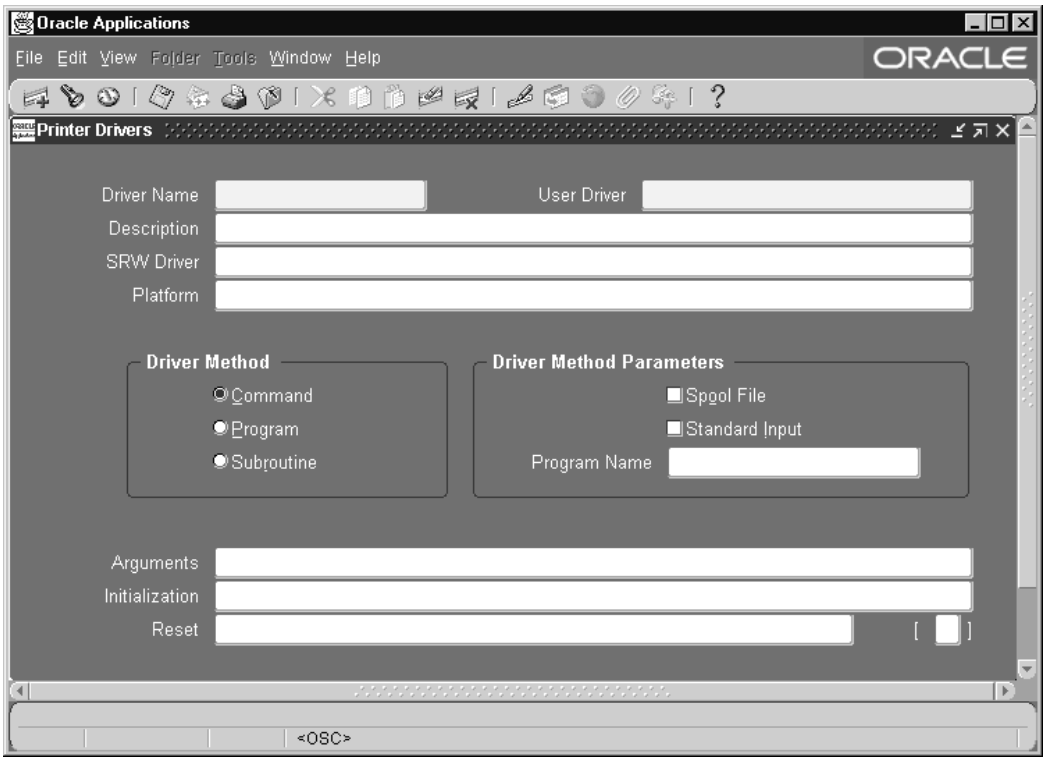
Printer Types, Print Styles, and Printer Drivers , page 6-2

Printer Types field help, page 6-47

Printers field help, page 6-49

Printer Drivers field help, page 6-53

Printer Drivers Window



Use this window to define your printer driver and printer commands.

> **Important:** You should issue a Restart concurrent manager command for all currently active managers whenever you edit an existing Printer Type, Print Style, or Printer Driver.

Oracle E-Business Suite ships printer drivers for the following print styles:

- Portrait
- Landscape
- Landwide
- A4
- Dynamic Portrait

Oracle E-Business Suite also ships printer drivers for specific printer types, including the following:

- Apple

- DEC LN03
- HP Laserjet II, HP Laserjet III, HP Laserjet 4
- HP line printer, HP 256X line printer
- EPOCH
- EPSON FX1050 and DMTX1
- QMS PS 825/925

Define additional printer drivers if you have different types of printers, or define additional print styles.

Printer Drivers Window Fields

Driver Name

The printer driver name must be unique for a given platform.

User Driver

This user name is referenced by Oracle E-Business Suite and must be unique for a given platform.

SRW Driver

Enter the name of the Oracle Reports (SRW) printer driver, if any, that will be invoked by your printer driver. Only Oracle Reports programs require this information.

Enter the entire path to the file, or just the file name. If you enter only the file name, Oracle E-Business Suite assumes the file is located in the $FND_TOP/$APPLREP directory.

Platform

Select the platform for which the printer driver is defined. Do not assign platform codes to printer drivers unless you have multiple drivers of the same name. If it cannot find a specific platform code associated with a driver, the concurrent manager will default to the driver with a null platform code.

Driver Method Region

Command The printer driver executes within an operating system shell. An example is the lpr command in UNIX.

Program The printer driver executes directly as a program, not

through an operating system shell.

- An example is a C standalone program for printing.
- This method executes faster than the Command method, but cannot access shell commands like PRINT on MS-DOS.

Subroutine

The printer driver executes a predefined Oracle E-Business Suite routine.

Subroutines are specific to operating platforms and are invoked directly by a system call from the concurrent manager

Driver Method Parameters Region

Spool File

Select whether the printer driver creates its own copy of a file for printing. If this check box is checked when the Driver Method is set to Program, the print program creates its own spool file.

- An example of spool files is the UNIX lpr command, which creates its own copy of a file if you do not specify the -s option.

Standard Input

Select whether the printer driver accepts standard input. Uncheck this check box when the Driver Method is set to Program. Unless the program accepts standard input, this check box should always be unchecked.

- An example is the UNIX command lpr, which accepts standard input when a filename is not specified.

Program Name

If the driver method is *Program*, enter the full path to the program that the driver invokes. The path is not necessary if the program's location can be identified by the operating system's PATH environment variable. See Location for Custom Print Programs, page 6-34.

If the driver method is *Subroutine*, enter the subroutine name that the driver invokes.

Arguments

When the Driver Method is set to Program, enter any generic arguments that must be supplied to the print program.

When the Driver Method is set to Command, enter the full command and its

arguments.

Initialization

Enter the initialization string that must be sent to the printer before the printer driver can begin printing.

Reset

Enter the reset string that returns the printer to its ready state when printing is complete.

[ ]

The double brackets ([ ]) identify a descriptive flexfield that you can use to add data fields to this form without programming.

This descriptive flexfield allows you to define special commands specific to your printer driver and/or the platform it runs on.

Related Topics

Overview of Printers and Printing, page 6-1

Printer Types, Print Styles, and Printer Drivers , page 6-2

Creating Custom Printer Drivers, page 6-31

7

Setting Up Oracle E-Business Suite Help

Setting Oracle E-Business Suite Help Profile Options

Oracle E-Business Suite Help uses several profile options (in the profile category Help System):

- Applications Help Web Agent - set this profile option if you want to have online help launched on a web server other than that specified by the Apps Servlet Agent profile.
- Help Utility Upload Path - used in uploading help files. See: Downloading and Uploading Help Files, page 7-2.
- Help Utility Download Path - used in downloading help files. See: Downloading and Uploading Help Files, page 7-2.
- Help System Root - determines which Contents navigation tree appears when the online help first launches. At the site level, this should be set to 'fnd:library'. To have an application's navigation tree appear in context-sensitive help, set this profile at the application level to '<application short name>:<root node key>'. For example, set it to '<ap:contents>' for the Oracle Payables main tree.
- Help Localization Code - used to group and differentiate help documentation within a language.

Customizing Oracle E-Business Suite Help

This section describes how you can create and upload custom Oracle E-Business Suite help files for the online help system.

Oracle E-Business Suite help files are formatted as HTML allowing easy modification using commercial HTML text editors. You can also add customized files of your own to the help system.

If you have licensed Oracle Tutor, you can use it to edit your Oracle E-Business Suite help files. See: Using Oracle Tutor, page 7-17.

> **Caution:** With each new release of Oracle E-Business Suite and each patch you accept, you may need to reapply your changes to any updated help files you have modified, if you want access to the latest information.

Customizing Oracle E-Business Suite Help includes the following topics:

- Downloading and Uploading Help Files (Help System Utility), page 7-2
- Linking Help Files, page 7-5
- Updating the Search Index, page 7-8
- Customizing Help Navigation Trees, page 7-8
- Customizing Help in a Global Environment, page 7-16

Customizing help files involves utilities available under "Help Administration" from the seeded System Administration responsibility. These utilities include the Help Builder and the Help Upload and Help Download functions.

Downloading and Uploading Help Files

The Generic File Manager Access Utility (FNDGFU) is used for downloading and uploading help files. For an overview of this utility, see: Generic File Manager Access Utility (FNDGFU), page 13-25

Oracle E-Business Suite help files are stored in the database. The Oracle E-Business Suite Help Administration pages of the Help System Utility are provided for retrieving and replacing them in the course of customization.

Setting Help System Utility Profile Options

Before using the Help Download or Help Upload pages you should define the upload and download directory paths. Oracle E-Business Suite provides profile options for you to set these paths.

Use the profile option Help Utility Download Path to define the directory location to which the Help System Utility will download files. Use Help Utility Upload Path to define the directory location from which your customized files will be transferred back into the Oracle E-Business Suite Help System.

Identifying Help Files for Customization

Help files are downloaded by file name. To identify the specific file that you want to customize, open the document in the Oracle E-Business Suite Help System. Use the

view source function of your browser to view the HTML source code. The source information will include the file name.

To identify the language and product of the help file, use the source document URL. The final three nodes of the source document URL are the language, the product name, and the anchor or target name.

Using this document again as an example, you will see the final three nodes of the URL are /US/FND/@ht_updown#ht_updown. This identifies the language as US, the product group as FND (Oracle Application Object Library), and the target name as ht_updown.

> **Note:** The syntax in the URL, @ht_updown#ht_updown, is an example of the Oracle E-Business Suite special syntax used to link documents by anchorname. For more information about this syntax see Linking Help Files, page 7-5.

The Oracle E-Business Suite Help System also provides reports to cross-reference target names and help file names. See Creating Reports, page 7-4.

Downloading Help Files

You download help files by language and by product. That is, you select the language (for example, US for U.S. English) and you select the product (for example, AR Oracle Receivables). It is important to note the two-letter code for the product (in this example, the two-letter code is AR) because the product code determines the download directory.

To download files, navigate to the Help Download page. This page is listed under Help Administration available under the seeded System Administration responsibility.

Follow these steps to download a single help file:

1. Select Single File Download.
2. Enter search criteria for the file you want.
3. Select the file you want from the search results, and select Download.

Follow these steps to download multiple help files:

1. Select Bulk Files Download.
2. Enter in the Target Directory and Filter Options as appropriate.
3. Select Download.

Uploading Help Files

Once you have customized the help files, use the Help System Utility to upload the documents into the help system. Your files are uploaded from the upload directory specified in the profile option Help Utility Upload Path.

You can upload different types of files, including:

- HTML files (all HTML files must have a .htm extension)
- GIF graphics files (must have a .gif extension)
- Adobe® Acrobat files (must have a .pdf extension)
- Cascading Style Sheets (must have a .css extension)

Follow these steps to upload a single help file.

1. Copy the customized file to a directory accessible from your desktop. .
2. Navigate to the Help Upload page. This page is listed under Help Administration available under the seeded System Administration responsibility.
3. Select Single File Upload.
4. Specify the application under which this help file belongs.
5. Specify the customization level (100 and above indicates a custom file; below 100 indicates a file from Oracle).
6. Select Upload.

Follow these steps to upload multiple help files:

1. Copy the customized files to the appropriate product folder in the upload directory.
2. Specify the application under which these help files belong.
3. Specify the customization level (100 and above indicates a custom file; below 100 indicates a file from Oracle).
4. Select Upload.

Creating Reports

The Help System Utility provides two reports for you to cross-reference help targets and file names.

Help Target to File Names Report This report lists by target, each file that contains the target, the document title of the file, and the product.

File Name to Help Targets Report This report lists every file name and document title by language and product and all the targets found within each file.

Follow these steps to run these reports:

1. Navigate to the Help Reports page. This page is listed under Help Administration

available under the seeded System Administration responsibility.

2. Select the report you want to run.

3. Select the desired application, language, and output format.

4. Click Go.

Linking Help Files

The Oracle E-Business Suite help system supports a special syntax for hypertext links that keeps them working even when files are renamed or split into parts. The special syntax, which is explained in detail below, looks like this:

```
For more about widgets, see  <A HREF="@widgets#widgets">All About
Widgets</A>.
```

Oracle E-Business Suite help files use this syntax, and you can use it too in your custom help files. Or if you prefer, you can always use conventional hypertext links based on filename.

Linking Help Files includes the following topics:

- Special Link Syntax, page 7-5
- Cross-Application Links, page 7-6
- Related-Topics Links, page 7-7
- Context-Sensitive Help, page 7-7

Special Link Syntax

Links in Oracle E-Business Suite help files point, not at a particular filename, but rather at one of the named anchors contained in the file. The Oracle E-Business Suite help system resolves anchorname to file dynamically, every time a link is negotiated.

Information on which files contain which anchornames is put into the help system automatically on upload. Authors must ensure that anchornames are unique across an application's help files to prevent duplicate links. In return, they need never worry about a change in filename breaking their links.

Named Anchors in Conventional HTML

By **named anchor** is meant the following kind of HTML tag:

```
<A NAME="anchorname"></A>
```

Named anchors can be placed anywhere in the body of an HTML file, and are typically used for links internal to the file in question. A pound sign (#) is placed before the anchorname in the link that points at it.

For example, you would use HTML like the following to allow users to jump forward to a section with the anchorname of "widgets":

```
For more about widgets, see  <A HREF="#widgets">All About Widgets</A>
below.

<A NAME="widgets"></A> <H2>All About Widgets</H2>
```

Extended to Support Interdocument Links

Oracle E-Business Suite help files extend this conventional HTML syntax to create links not only within, but also **between** help files. To link to a file that contains a particular named anchor, you simply place an at sign (@) before the anchorname. To link to the precise spot within the file where this anchor appears, you append a pound sign followed by the anchorname, just as you would in conventional HTML. This results in the following special syntax:

```
<A HREF="@anchorname#anchorname">link text</A>.
```

For example, to link to the file that contains the "widgets" anchor illustrated above, at the point in the file that this anchor occurs, you would use HTML like the following:

```
For more about widgets, see  <A HREF="@widgets#widgets">All About
Widgets</A>.
```

If you simply want to link to the top of the file that contains this anchor, you can omit the pound-sign segment "#widgets."

Links in Oracle E-Business Suite help files rarely omit the pound-sign segment. This means that however topics are rearranged within or among files, links to these topics from other files always go to the proper file, and to the precise spot within the file where the topic occurs.

> **Caution:** Do not use case to make distinctions between anchornames. Unlike most web browsers, the Oracle E-Business Suite help system treats anchornames in a case-insensitive fashion.

Cross-Application Links

In the Oracle E-Business Suite help system, all help files associated with a particular application exist in the same directory, as far as their URL is concerned. Help files associated with other applications exist in directories named after the application's short name. All these application directories are at the same level in the help system.

To create a link that goes to a help file associated with a different application, you create a relative link that goes up a level to the parent of all help application directories, and then back down through the other application's directory, before concluding with Oracle E-Business Suite's special link syntax. This results in the following cross-application link syntax:

```
<A HREF="../shortname/@anchorname#anchorname"> link text</A>
```

For example, if the "All About Widgets" topic illustrated above were an Oracle Payables help topic, and you wanted to link to it from an Oracle General Ledger help file, you

would use a link like the following, where AP is Oracle Payables' short name:

```
For more about widgets in Oracle Payables, see  <A
HREF="../AP/@widgets#widgets">All About Widgets</A>.
```

When used in this fashion, application short names are case insensitive.

> **Note:** These application help directories are merely "virtual" directories recognized by the Oracle E-Business Suite help system when used in URLs. All files are actually stored in the database, with application short name being one attribute among many associated with them.
>
> Oracle Payables' official short name is SQLAP. This has been shortened to AP for the virtual directory used in the Oracle E-Business Suite help system. Similarly, Oracle General Ledger's official short name of SQLGL has been shortened to GL, and Oracle Assets short name of OFA has been shortened to FA.

Related Topics Links

Links are not limited to a single target in the Oracle E-Business Suite help system. You can point your links at multiple topics and files by using the following syntax:

```
<A HREF = "@anchorname1,anchorname2,anchorname3"> Related Topics</A>
```

When a user negotiates the link, a page headed "Related Topics" appears, containing a list of the page titles corresponding to these anchornames, with each title linked to the file in question.

To include cross-application links, simply prefix the application short name and a colon to the anchorname:

```
<A HREF = "@anchorname1,shortname:anchorname2,anchorname3"> Related
Topics</A>
```

Context-Sensitive Help

When you ask for help in Oracle E-Business Suite, the topic for your current window opens. If you ask for help from a report parameters window, your help file opens to a discussion of that report.

Oracle E-Business Suite help files contain special anchornames to enable these context-sensitive links. When calling help from an Oracle E-Business Suite Forms-based window, Oracle E-Business Suite looks for an anchorname based on the form name and the window name combined as follows:

```
<A NAME="form_name_window_name"></A>
```

You can override the form_name portion of the anchorname by specifying a HELP_TARGET parameter in the parameter field of the Form Functions window. Use the syntax HELP_TARGET = "alternative_form_name". See: Form Functions, *Oracle Security Guide*.

When calling help from an Oracle Application Framework (HTML-based) window from the global Help button, the anchorname follows this syntax:

```
<A NAME="appShortName_packageFunctionalComponent_pageName"></A>
```

For more information on coding for the global Help button, refer to the documentation listed in the *Oracle Application Framework Developer's Guide*, available from My Oracle Support Knowledge Document 1315485.1.

When calling help from a report parameter window, Oracle E-Business Suite looks for an anchorname constructed as follows:

```
<A NAME="SRS_concurrent_program_shortname"></A>
```

Updating the Search Index

Oracle interMedia Text enables the search feature provided by the Oracle E-Business Suite help system. Run the concurrent program Rebuild Help Search Index to rebuild the search index after uploading customized documents. This ensures that they will be included in any searches your users perform.

Customizing Help Navigation Trees

You use the Help Builder applet to customize the help navigation trees that appear in your browser window's navigation frame when help is invoked.

> **Caution:** With each new release of Oracle E-Business Suite and each patch you accept, you will need to reapply your changes to any updated help navigation trees you have modified, if you want access to the latest information. In addition, Oracle does not provide any mechanism for identifying changes between releases of Oracle E-Business Suite help navigation trees.

You can use the Help Builder to perform the following tasks:

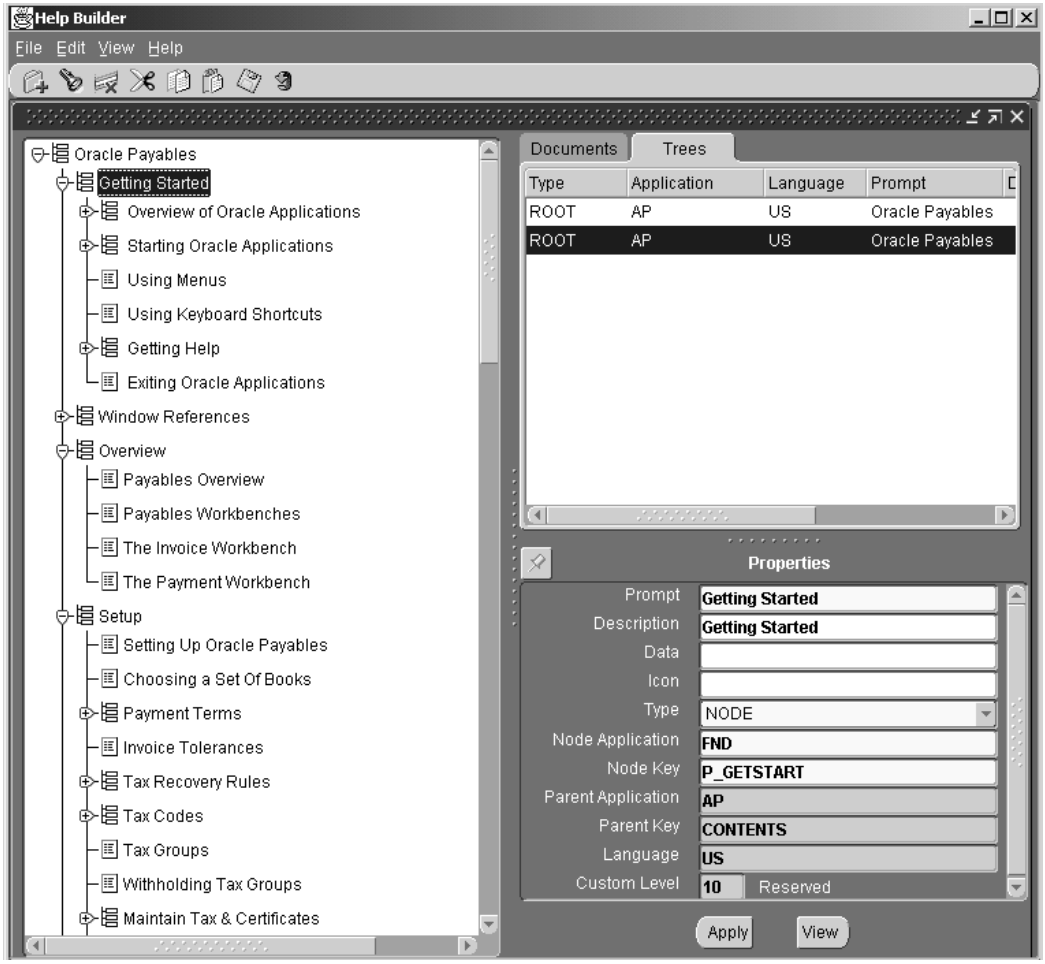
- Open a tree for editing, page 7-11
- Add new help files to a tree, page 7-12
- Add new nodes to a tree, page 7-13
- Add nodes to one tree from another, page 7-14
- Change the organization of a tree, page 7-14
- Create a new navigation tree, page 7-15

For help understanding the information associated with each of the Help Builder's fields, see Help Builder Window Reference, page 7-15.

Accessing the Help Builder

To access the Help Builder, use the seeded responsibility System Administration and navigate to the Help Builder under Help Administration.

The Help Builder User Interface



The Help Builder window default view consists of three panes. The left pane displays the tree that is currently selected. Use this area to manipulate your tree by adding nodes, deleting nodes, and dragging nodes into the positions desired. The top right pane displays items matching the searches you have performed using the Find Documents or Find Trees functions. Click on the Trees or Documents Tab as appropriate. The bottom right pane is the Properties Pane. This area displays the properties of the item (root, node, or document) currently selected. Those properties with enabled fields can be updated.

Help Builder Menus

Help Builder has the following menus:

File Menu

The File Menu provides the following functions:

New Creates a new root node. Selecting this option will open the Root Node Properties window for you to enter the appropriate values for your new root node.

Open Opens a tree. Selecting this option will open the Find Trees window for you to enter selection criteria to find the appropriate tree.

Save Saves the current changes.

Reload Reloads the current tree to apply all changes throughout the tree hierarchy.

Print Prints the current tree pane.

Exit Exits the Help Builder.

Edit Menu

The Edit Menu provides the following functions:

New Node Creates a new node beneath the selected node on the tree. If the selected node is a document, a document node is created. If the selected node is a branch node, a branch node is created.

Cut Cuts the selected item (document or branch node).

Copy Copies the selected item (document or branch node).

Paste Pastes an item beneath the currently selected node on the tree. If the node selected is a document, the item will be pasted beneath it on the tree. If the node selected is a branch that has been expanded, the item will be pasted as a child of the selected node. If the node selected is a branch that has not been expanded, the item will be pasted as a sibling beneath the selected node.

Delete Deletes the selected item.

Properties... Allows you to update the properties of the selected item via the Properties window. The Properties window is identical to the Properties Pane.

See Help Builder Window Reference, page 7-15 for descriptions of the Properties fields.

Preferences... Allows you to set interface preferences via the Preferences window. The preferences you can set are:

Background Color

Line and Box Color

Default Node Color

Default Node Text Color

Font Name

Font Size

View Menu

The View Menu provides the following functions

Node Properties Enable the check box to display the Node Properties pane (enabled is the default).

Toolbar Enable the check box to display the Toolbar (enabled is the default).

Statusbar Enable the check box to display the Status Bar (enabled is the default).

Find Documents Opens the Find Documents window.

Find Trees Opens the Find Trees window.

Help Menu

The View Menu provides the following functions:

About... Displays information about the Help Builder.

Library Opens the Oracle E-Business Suite Help Library.

Help Builder Tasks

Opening a tree for editing

To open a tree for editing:

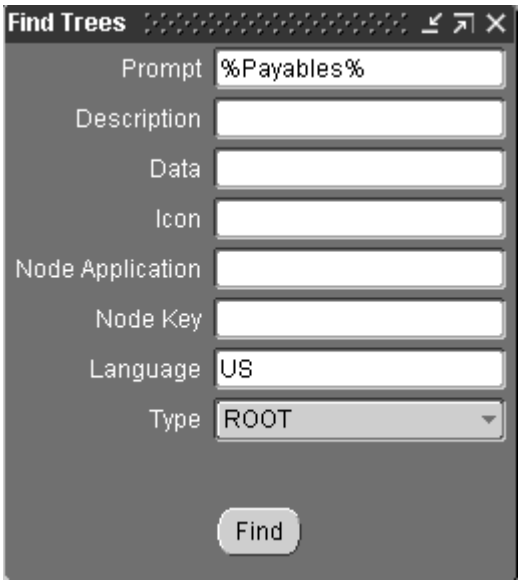
1. Open the Find Trees window using one of the following menu options:

   (M) File > Open

   (M) View > Find Trees

2. Enter your search criteria in the **Find Trees** window, and click **Find**.

   For example, enter %Payables% in the **Prompt** field to find the Oracle Payables tree.



See Help Builder Window Reference, page 7-15 for descriptions of the Find Trees search fields.

3. Trees matching your criteria will be displayed in the upper right pane of the Help Builder window on the Trees tab. Select a tree in the list to see its properties displayed in the Properties pane below. Double-click a tree to open it.

   The tree's top-level node will appear in the left pane. Expand and contract nodes to display the part of the tree you want to edit.

Adding new help files to a tree

To add new help files to a tree:

1. Upload the help files to the database, page 7-2.

2. Open the tree for editing, page 7-11.

3. Click the **Find Documents** icon on the toolbar, or select Find Documents from the View Menu.

   The **Find Documents** window appears.

See Help Builder Window Reference, page 7-15 for descriptions of the Find Documents search fields.

4. Enter your search criteria, select **Exclude documents already on a tree**, and click **Find**.

   Files corresponding to the information you entered appear on the **Documents** tab of the upper right pane of the Help Builder window. Select a document to view its properties in the Properties Pane.

5. To add the file to the tree, drag it from the **Documents** tab and drop it on the tree in the the position desired.

   Note: Files containing multiple named anchors appear multiple times. Each Target will have its own listing. Choose the Target that corresponds to the topic you want to add.

   If the topic you want to add is not the header target of the file, but a target within the document, you must supply the special link syntax in the **Data** field of the Properties Pane.

   For example, the target name might appear in the Data field as @ht_updown. To link directly to this anchor you would add #ht_updown to the end of the anchorname. The resulting entry in the **Data** field will be @ht_updown#ht_updown.

   For more information about this syntax see Linking Help Files, page 7-5.

Adding new nodes to a tree

To add new nodes to a tree:

1. Open the tree for editing, page 7-11.

2. Select the node beneath which you want to add a new node, and click **New Node** on the toolbar or select New Node from the Edit Menu.

   Note: The New Node feature will add a node that is like the node selected. For example, if a branch node is selected a branch node will be added beneath it. It a leaf node is selected, a leaf node will be added beneath it.

   When adding a branch node as a sibling to an existing branch node, be sure that the selected branch node is not expanded. If the existing branch node is expanded, the new node will be added as a child to the selected branch.

3. Enter information for the new node in the Properties Pane, and click **Apply**.

4. Click **Save** to save your changes.

Adding nodes from one tree to another

To add nodes from one tree to another:

1. Open the tree for editing, page 7-11.

2. From the View menu select **Find Trees**.

3. In the Find Trees window select **Node** from the **Type** poplist and enter search criteria for the nodes you want to add. Click **Find**.

   Nodes corresponding to the information you enter appear on the **Trees** tab of the main Help Builder window.

4. Drag nodes from the **Trees** tab and drop them on the tree.

5. Click **Save** to save your changes.

Changing the organization of a tree

To change the organization of a tree:

Caution: Changes made to nodes added from another tree are reflected in the original tree and all other trees that include them.

1. Open the tree for editing, page 7-11.

2. To move a node, drag the node from its current location and drop it at its new location in the tree.

3. To delete a node, select it and click **Delete** on the toolbar or select

Delete from the Edit Menu.

4. To change a prompt, select the node, enter the new prompt in the **Prompt** field of the **Properties** pane, and click **Apply**.

   Other node properties can be changed in a similar fashion.

Note: If the same node appears elsewhere in the tree, your changes will not appear there until you click the **Reload** button on the toolbar. For one node to be the same as another, the Node Key and Node Application of their parent nodes must be the same, as well as all their own properties. Their grandparent nodes and above can be different.

Creating a new navigation tree

To create a new navigation tree:

1. Choose **File** >**New**.

   The **Root Node Properties** window appears.

2. Enter information for the tree's root node, and click **Apply**.

3. Add new nodes to the tree, page 7-13.

4. Add new help files to the tree, page 7-12.

5. Add nodes from other trees to the tree, page 7-14.

6. To view the new tree with context-sensitive help, enter its root as the Help Tree Root for some application, responsibility, or user.

   To view it stand alone, substitute its root in the root= parameter at the end of your site's Oracle E-Business Suite Help URL. Include the application short name. For example, for a root named ROOT_INV belonging to the Oracle Inventory product, you would use root=INV:ROOT_INV in the URL.

Help Builder Window Reference

Field names and descriptions for the Help Builder window are given below.

Application Application shortname of application that owns the help file.

Custom Level Customization level of the node. 100 is the default for customer use. Levels under 100 are reserved for system use.

Data	If the node links to a help file, the file name or a target name preceded by an "@" symbol, or an absolute URL.
Description	Longer description of the node, if the Prompt is terse. Otherwise may simply repeat the Prompt.
Filename	Pre-upload filename of the help file.
Icon	Not used.
Language	Language code of help files covered by the node.
Node Application	Application shortname of the application that owns this node. If different from the value given for the ROOT, this node and all the nodes it branches into have been grafted into the tree from another application.
Node Key	String that uniquely identifies the node in this Node Application. The node key can be generated automatically or typed into the field (for a new node). However, once references to the node exist, the node key cannot be changed.
Prompt	The text that appears on the tree for this node.
Target	Anchorname contained in the help file. Do not precede with an @ sign in the Find Documents window.
Title	Title of the help file.
Type	ROOT: Top-most node of a navigation tree. NODE: Node that branches into other nodes, but is not the ROOT. If **Data** is not null, it links to a help file as well. DOCUMENT: Node that does not branch into other nodes, but simply links to a help file.
Version	Version identifier of the help file.

Customizing Help in a Global Environment

The Oracle E-Business Suite help system contains files translated into many different languages, and localized for diverse countries and regions. If your enterprise crosses linguistic and cultural boundaries, or if you use Oracle Human Resources, the following information may apply when customizing your help files.

Linking Between Different Languages

One level up the virtual directory hierarchy used in Oracle E-Business Suite help URLs are the application directories used to construct cross-application links. Two levels up are the language directories, which you can use to construct cross-language links.

To create a link that goes to a help file in a different language, use the following link syntax:

```
<A HREF = "../../language_code/shortname/@anchorname#anchorname"> link
text</A>
```

For example, to link to "All About Widgets" in the French version of Oracle Payables help, you would use the following link, where AP is Oracle Payables' short name and F is the French language code:

```
For more about French widgets, see  <A
HREF="../../F/AP/@widgets#widgets">Qu'est-ce qu'un widget?</A>.
```

When used in this fashion, language codes are case insensitive.

> **Note:** After following a link to a different language, users stay in that language until they follow a link back out to their original language. This can be either a link they encounter within a help file, or a link from the navigation tree, which remains in their original language throughout.

Using Oracle Tutor

Oracle Tutor is a documentation tool designed to help companies create and maintain their end user process documentation. Oracle Tutor defines end user process documentation as policies and procedures, instructions, and support documents. Oracle Tutor also provides a system for

- distributing process documentation to employees by job role
- customizing Oracle E-Business Suite online Help
- integrating process documentation with Oracle E-Business Suite courseware
- building customized Oracle E-Business Suite training materials based on job title or topic
- auditing process documentation to ensure accuracy and effectiveness

Oracle Tutor is comprised of three components:

- the documentation methodology
- model documents

- software

The Tutor **method** defines all aspects of the documentation process: the content and format of the different document types, as well as the relationship between documents; the process by which documents are created and maintained; and the methods by which documents are distributed and used for training and reference purposes.

Tutor **model documents** are complete process documents and courseware that provide a starting point -- document "owners" edit the models instead of creating documents from scratch.

Tutor **software** includes tools for keeping documents up to date and distributing them on a need to know basis.

- Author is the tool used by individual document owners to create and edit MS Word documents quickly and easily.

- Publisher is the tool used to

  1. build desk manuals by job title,

  2. build student and instructor guides by job title or topic, and

  3. ensure document integrity through the generation of a wide range of cross-reference reports.

Together, these three components help companies overcome the operational challenges of documenting and maintaining business practices.

For more information, please see the *Tutor Author User Manual*, the *Tutor Publisher User Manual,* and the *Tutor Implementation Guide.*

8

Time Zone Support

User-Preferred Time Zones

Overview

Release 12.2 of Oracle E-Business Suite includes as standard a feature called *User-Preferred Time Zone Support*. In most existing E-Business Suite implementations, all users interact with the system in the corporate time zone, which will normally be the time zone of the headquarters of the implementing company, and the time zone in which the database runs. This means that remote users have to be aware of the time difference between their location and that of the corporate headquarters.

Employing the user-preferred time zone feature enables users to specify their local time zone for both display and entry of date-with-time fields. Key consequences of this are:

- Users see date-with-time fields in their preferred (local) time zone, and can enter dates with time in this time zone.
- Date fields without a time component are not affected by this feature.
- The data in the database continues to be stored in the standard corporate time zone.

This chapter discusses the capabilities, limitations, and implementation details of the user-preferred time zone feature.

Definitions

Server Time Zone/Corporate Time Zone - Server time zone or corporate time zone are used to describe the same concept throughout this chapter. They reflect the primary time zone set at your site, and usually correspond to the time zone of your headquarters or primary operation (the primary legal entity time zone). You set this value in the 'Server Timezone' site level profile. It must match the database level setting for server time zone.

Important: Oracle strongly recommends that the database time zone

should not be changed once it has been set and data has been entered.

Client Time Zone/User Time Zone - Client time zone or user time zone are used as synonyms to describe the same concept throughout this chapter. They reflect the time zone in which the user is operating, which is normally also the time zone in which he is physically located.

Location-Based Time Zone - Location-based time zone refers to the time zone of the contextual location. Uptake of location-based time zone is product-specific, and is not covered in this chapter.

Time Zone Concepts

Conceptually, there are two types of date fields:

- Dates *with* a time component – used to indicate a specific point in time within a particular day
- Dates *without* a time component – used to denote a particular day, but not a specific point in time within that day

Date fields with a time component can be represented in any time zone, and thus displayed in whichever time zone is most meaningful to the end user. Generally, users prefer to view dates in their own (local) time zone. With the user-preferred time zone feature enabled, date with time fields will be converted to the user's preferred time zone for display.

Date fields without a time component cannot be represented in different time zones, because no meaningful conversion is possible for a date that does not include a specific time. Such a date is entered with respect to one time zone, and in general must be viewed as a day in that time zone, regardless of the location (and possibly different time zone) in which it is being viewed. Oracle E-Business Suite typically uses the corporate time zone for these day definitions: dates without a time component represent the day with respect to the corporate headquarters (corporate days).

There are some exceptions to this rule. For example, dates without a time component may be held as *ANSI dates*, to represent dates independently of the time zone in which they are being viewed. In such a case, a benefit that starts on 1st January will start on that date wherever in the world it is made available; that is, it will apply to anyone who is in a time zone where it is 1st January.

Many dates without a time component represent pointers to a financial period. These dates are not meant to indicate the exact hour and minute that a transaction occurred, but rather the financial period into which the transaction is accounted. This is a financial bucketing from the perspective of the implementing company. For example, the invoice dates on Payables or Receivables invoices never change based on who is looking at them: they are classified as invoices for that day (and thus that period), regardless of the viewer. These financial accounting dates are often defaulted or derived

from actual transaction dates, and it should be understood that although the transaction dates will be converted to the user's preferred time zone for display, the financial accounting dates will not.

For a discussion of time zones in the context of other globalization-related topics, see Globalization Support in *Oracle E-Business Suite Concepts*.

User-Preferred Time Zones and Products

This section examines the considerations that apply when employing user-preferred time zones with a range of Oracle E-Business Suite components and products.

First, the effect of user-preferred time zones on interactive and non-interactive user interface components will be examined. This is followed by a comprehensive survey of points that should be considered when employing user-preferred time zones with various Oracle E-Business Suite products.

User-Preferred Time Zone Support for Interactive UI Components

Oracle Forms-based UIs derive the user-preferred time zone based on the user profile 'Client Timezone'. This profile option can be set on a per-user basis, to suit individual requirements.

HTML-based UIs derive the user-preferred time zone based on the preference 'Timezone'. This profile option can be set on a per-user basis, to suit individual requirements.

When automatic user-preferred time zone conversion is enabled, Forms-based and HTML-based user interfaces will behave as described in the following subsections.

Oracle Forms-based UI Components

The behavior of Oracle Forms components is described below.

Oracle Forms: Using the Calendar List of Values

The calendar list of values shows the name of the time zone associated with that date field (this will be the user's preferred time zone if it has been set up, or the corporate time zone if the user has not yet set his time zone preference).

A Convert Time Zone button enables you to view the date with the time value converted to any time zone of interest. This built-in time zone calculator may be useful for users who occasionally need to see a particular date with time in a time zone other than their own. Additionally, you can enter a date with time in a specific time zone of your choosing, and when you close the calendar list of values, the value will be converted to the time zone associated with the field for display.

Oracle Forms: Find Windows

When you enter search criteria in a date field of a find window, you must be aware of whether you are searching on a date that has a time component or not. Dates without time components will be searched using the standard corporate time zone. Dates with time components will be searched using the user's preferred time zone.

Oracle Forms: Query By Example

When querying on a date with time field, you can enter a date with a specific time, and retrieve transactions for that specific date and time in your preferred time zone. If you enter only a partial value, however, the query will run as though there is no time component, and return records that match the partial value in the corporate time zone.

Queries on date fields without a time will continue to return records matching the date in the corporate time zone, as in previous releases.

Oracle Forms: Concurrent Report Submissions

The concurrent request windows or Standard Report Submission form will operate in the user-preferred time zone. The job queues will show the requested dates and times according to the user's local time zone, including jobs that were submitted by users in other time zones. If a job is scheduled to run at 12:30 pm in New York time (GMT-5), but if a user in Los Angeles (GMT-8) looks at the queue, he will see that the job as scheduled for 9:30 am Los Angeles time (GMT-8).

Concurrent reports output (reports, log files and out files), will, however, always use the standard corporate reporting time zone.

Oracle Forms: WHO Columns

The Forms Record History values (WHO columns) are dates with time but will always be displayed in the corporate time zone.

Oracle HTML-based UI Components

The behavior of Oracle HTML-based UI components is described below.

HTML UIs: Using the Date Picker

The date picker will allow you to pick a specific date, but not a time component. Once a date with time is selected, the time component will generally default to the current time unless otherwise specified. The date with time is displayed in relation to the client time zone.

HTML UIs: Search

Unless specified otherwise, date with time based searches are based on the client time zone, and always require a time component to be specified. When left blank, a default time (usually 00:00:00) will be appended.

Unless specified otherwise, date-only based searches will be based on the corporate time zone.

Flexfield Dates or Dates with Time

Key and Descriptive Flexfields where the Value Set is defined with the DateTime datatype will have the user-preferred time zone conversion applied. You should review the custom Flexfield Value Set definitions before enabling time zone support. If the user-preferred time zone conversion is considered inappropriate for a particular Flexfield segment. then consider changing the ValueSet used, or the ValueSet datatype.

Note: Oracle-provided DateTime ValueSets and Flexfield segments

cannot be changed.

Concurrent Program Parameter definitions are based on Flexfields. On the Process/Report Submission form, all DateTime parameters will honor the user-preferred time zone.

User-Preferred Time Zone Support for Non-Interactive UI Components

When automatic user-preferred time zone conversion is enabled, non-interactive user interfaces will behave as described in the following subsections.

Interface Tables and Programs

All dates or dates with times loaded into an Oracle E-Business Suite interface table should be converted to the standard corporate time zone prior to loading.

Public APIs

All dates and date with times passed into or out of Oracle E-Business Suite public PL/SQL or Java APIs are assumed to be in the standard corporate time zone, unless specified otherwise. HCM (HRMS) APIs are a notable exception, and HCM (HRMS) dates are in the local time zone of the transaction.

Ad Hoc Reporting Tools

All dates and date with times viewed from the Oracle E-Business Suite schema through any ad hoc reporting tools should be assumed to be in the standard corporate reporting time zone. The exception is HCM (HRMS) dates, which are in the local time zone of the transaction.

Unless specified otherwise, Oracle Reports and Oracle XML Publisher reports display Date and Date with time in corporate time zone.

Database Import/Export

Whenever data is moved between Oracle E-Business Suite implementations (or from another system into an Oracle system), the time zone of the source system should be compared to the time zone of the Oracle E-Business Suite schema (the standard corporate time zone). If the time zones are different, then the appropriate conversions should be made to all date with time fields before import into Oracle E-Business Suite.

Oracle Web Applications Desktop Integrator (Oracle WebADI)

All date and datetime values which are downloaded using WebADI are in the corporate time zone.

EDI Gateway

EDI Gateway is aware of the standard corporate time zone for Oracle E-Business Suite. For selected transactions, the time zone identifier will be bundled with the transaction

data itself, so that the destination system can properly interpret the date. Likewise, for inbound transactions, if a time zone is specified with a date field, EDI Gateway will convert the date to the standard corporate time zone before inserting the data into Oracle E-Business Suite. See the EDI Gateway documentation for specific transaction details.

Implementation Considerations for Oracle Service Products

The Service products use a combination of transaction-specific time zones and user-preferred time zones for displaying dates and times. With the user-preferred time zone feature enabled, service agents are able to see dates in their preferred time zone. In addition, they can see the current time in the time zone for the primary contact of the service request. This information can also be used to relate to customers in remote time zones.

Field service technicians are also able to select their preferred time zone and interact with the system in that time zone when requesting material for a repair at a customer site, or requesting availability of material to be picked up or shipped to the work site.

As part of Service setup, consider the following implementation tasks:

- Define coverage templates
- Define customers
- Define calendar, shift, and shift assignments
- Define resources and assign resources to calendars

Each of these wiill be described further below.

Define Coverage Templates

Coverage templates allow coverage times to be specified in multiple time zones. Use the time zones of your call center locations as the time zones for your coverage templates.

For example, coverage can be offered from 09:00 to 17:00 Eastern Time and 08:00 to 16:00 Central European time for each time zone. Different service levels can then be specified in terms of reaction and resolution times. This allows modeling of multiple call centers, each working in a different time zone.

A particular time zone coverage can also be defined as the default time zone coverage, and applied to all other time zones. This allows modeling of a single call center that works in a specific time zone, and which takes calls at hours specified in that time zone. If you offer 24x7 coverage, you can define a single coverage using any time zone.

Define Customers

The time zone of the location should always be specified when defining addresses for customers and contacts.

Define Calendar, Shift, and Shift Assignments

Use the Define Shift form to define calendars, shifts, and shift assignments. Shifts are independent of time zones, and can be reused across time zones. For example, if technicians in New York work from 09:00 to 17:00 Eastern Time, and technicians in San Francisco work from 09:00 to 17:00 Pacific Time, only one calendar and one set of shifts should be defined.

Define Resources and Assign Resources to Calendars

When defining a field service technician resource, his primary time zone must be specified in the Service tab, and the resource assigned to the corresponding shift: the shift pattern will then be applied in the technician's time zone.

Implementation Considerations in Oracle Common Application Components

Defining Resource Work Shifts in the Forms-based Calendar

A shift is defined outside the context of any time zone; only when it is applied to the resource that will use it is the appropriate time zone context applied. Consequently, shift definitions are not affected by the availability of the user-preferred time zone feature.

Standalone Task Manager in Forms or HTML

If a time zone is not selected for a task, then all times will be in the corporate time zone. If a time zone is selected for a task, and user-preferred time zone support is enabled, then times appear in the selected time zone.

Implementation Considerations in Supply Chain Management

Manufacturing Calendar

Although Bills of Material (BOM) calendars can be shared across facilities, the shift definitions are always defined in terms of the corporate time zone. Consequently, shift definitions are not affected by the availability of the user-preferred time zone feature.

There must be one calendar for each time zone in which manufacturing facilities are present. Facilities in the same time zone can share the same shift patterns. Facilities in different time zones will need their own shift definitions.

Complex Maintenance and Repair Operations

Complex Maintenance and Repair Operations (CMRO) users are advised to operate in the corporate time zone.

Cost Management

Accounting Periods

Accounting period start and end dates are stored and displayed in the corporate time zone. Users will be able to close an accounting period when the corporate time is past

the scheduled close date indicated on the Inventory Accounting Periods form.

Periodic Cost Processors

The Periodic Processors (Cost Processor, Acquisition Cost Processor, Acquisition Cost Adjustment Processor, Absorption Cost Processor, and Iterative Periodic Average Cost Processor) all require date fields to be input in corporate time.

Inventory Management

Inventory Positions

The inventory position is built and displayed in the corporate time zone.

Consigned Inventory From Supplier

For creating consumption advice documents, the 'Create Consumption Advice' concurrent program uses the corporate time zone to aggregate consumption transactions based on transaction date.

Work In Progress

Repetitive Production Lines

Users have to enter and view Repetitive Production lines Start and Stop time fields (time only fields) in the corporate time zone.

Flow Manufacturing

WIP Lines

In the WIP Lines form, users have to enter and view Flow Production lines Start and Stop time fields in the corporate time zone.

Planning Integration

In the Kanban Planning form and the Graphical Kanban Workbench form, users have to enter and view all date times in the corporate time zone. Planning outputs (Forecast, MDS, and MPS) are bucketed in the corporate time zone. Kanban size and number of cards are calculated on the basis of demands from planning. Consequently, in order to maintain consistency with planning, Kanban planning calculates and displays in the corporate time zone.

In Mixed Model Map Workbench form, users have to enter and view all date times in the corporate time zone. Planning outputs (Forecast, MDS, and MPS) are bucketed in the corporate time zone. Resource and in-process-Kanban size are calculated on the basis of demands from planning. Consequently, in order to maintain consistency with planning, Mixed Model Map Workbench calculates and displays in the corporate time zone.

Oracle Shipping Execution

Release Rules

When defining release rules, users can define pick release rolling windows using the Schedule Ship Dates and Requested Dates. user-preferred Timezone, when enable for the user, applies to Release rules. When a user with a user-preferred time zone employs a release rule defined in the corporate time zone, the Scheduled Ship Dates and

Requested Dates will be converted to that user's preferred time zone.

For example, if one user defines a Release Rule with a Scheduled Ship Date time of 9:00 AM in the corporate time zone, the same rule employed by another user will have the Scheduled Ship Date time converted to 9:00 AM in the second user's preferred time zone. Possible time zone implications should therefore be considered carefully when using the same release rule across multiple users in different time zones.

Quality

Charts

The Histogram, Control Chart, Pareto Chart, and Time Series Chart always use the corporate time zone to display the date and time quality collection elements on the axis labels.

User Defined Formula

If the user defined formula uses the date and time element from the server, the value will be in the corporate time zone, but if the formula uses the date and time element from the form field itself, the value will be in the user-preferred time zone (if set up).

Supply Chain Planning

The products in the Supply Chain Planning suite include Advanced Supply Chain Planning, Demand Planning, Global Order Promising, Inventory Optimization, Collaborative Planning, Transportation Planning, Manufacturing Scheduling, and Material Requirements Planning. Of these products, only Manufacturing Scheduling will display date/time fields in the user-preferred time zone. All other products will display date/time fields in the corporate time zone only.

The planning products display business documents such as sales orders, purchase orders, and WIP jobs, which are all collected from execution modules such as Order Management, Purchasing, and Work in Process. Although these execution modules can display the business documents in a user-preferred time zone, within the planning user interface the date/time fields on these documents will be shown in the corporate time zone only. There is no Date/Time conversion when data is collected from ERP server to planning server or released from planning server to ERP server.

Some of the Advanced Planning products show views of supply, demand and resource capacity data bucketed into daily, weekly and period time buckets. These views of data are always shown in the corporate time zone.

The Global Order Promising product is integrated with Oracle Order Management. The user can enter a request date for a sales order within Order Management. This date, entered in the context of the user-preferred time zone within Order Management, is converted to the corporate time zone when performing an availability check against Order Promising (promise date on the sales order). The results of Order Promising are converted back to the user-preferred time zone when displaying within the Order Management screens.

Material Requirement Planning (MRP)

Availability of the user-preferred time zone feature is not relevant to MRP since MRP date fields do not have a time stamp. MRP suggests completion dates to be the end of the day in the corporate time zone. If midnight in the corporate time zone happens to be during a working shift for the manufacturing plant, a job might be completed earlier or later than the expected date. This is not an issue for shop floor management (OSFM), as OSFM does not integrate with MRP.

Transportation Planning

Transportation Planning module supports location specific time zone. Review the Transportation Planning User Guide for details of the transaction-specific time zone support provided.

Field Service

Advanced Scheduler UI

The Advanced Scheduler UI can work in User-Preferred time zone mode, but this is only suggested if all incident sites are located in the same time zone. If scheduling will be done to incident sites in different time zones, Incident-Based time zone support should be set up.

Oracle Release Management

The horizon start date is pegged to the beginning of the day (in the corporate time zone), and the horizon end date is pegged to the end of the day (in the corporate time zone).

Workbench UI

All dates are in a user-preferred time zone on this workbench, except for the dates in the Horizontal Demand and Horizontal Schedule windows: these are always shown in terms of the corporate time zone.

CUM Workbench UI

The CUM start date on CUM Workbench is always shown in the corporate time zone.

Demand Processor

The horizon start date is pegged to the beginning of the day (in the corporate time zone), and the horizon end date is pegged to the end of the day (in the corporate time zone).

Oracle Process Manufacturing

Process Manufacturing Financials

In OPM Financials, the following processes are scheduled in the user-preferred time zone:

- Cost Rollup Process
- OPM Accounting Pre-Processor

- Actual Cost Process
- Cost Update Process

The start and end dates of the OPM Accounting Pre-Processor are stored and displayed in corporate time zone.

Oracle Supply Chain Trading Connector

Time Zone for Messaging

Oracle Supply Chain Trading Connector provides time zone support for messaging. RosettaNet specifies the time for messaging in GMT, using the ISO 8601 format, and requires all PIPs to be in GMT. However, XML Gateway does not convert this time to the time specified in the Trading Partner system. In addition, the window time and the database time can differ because of differences between user location and instance location. As a result, CLN supports conversion of time received in the system to the displayed window time. CLN also provides a default time for messages.

Other standards, such as OAG, specify the message time as an offset of GMT. For example, if GMT is 12:00 pm, then OAG presents the time as GMT-9. If a time zone is not dictated by the standard, then a default time is recommended.

Implementation Considerations in Oracle E-Records

Oracle E-Records operates in the corporate time zone, and all transactions are recorded and displayed in this time zone. The profile option 'EDR: Server Time Zone' must be set to the corporate time zone. If this value is not set, no events can be raised, and an error indicates that all e-records have a null value for the time zone.

Implementation Considerations in Financials

Accounting Dates on Financial Documents

In Oracle Financials, most dates simply bucket transactions into financial periods. Although these dates are often derived from actual transaction dates and times, these dates merely act as pointers to a financial period, rather than to an actual point in time. Conversions do not occur on such date fields. Consequently, use of user-preferred time zones has no effect on how financial documents and GL transactions are viewed and processed.

Currency Conversion Rates

As with accounting dates, the effective dates for currency conversion rates simply point to a day (or set of days) within an accounting period, and define the conversion rate that is used for transactions bucketed into that day (or set of days). Date conversions do not occur on conversion rate effective dates. Consequently, use of user-preferred time zones has no effect.

Subledger Transactions

When a transaction date with time is captured on a subledger transaction, such as a shipping date or purchasing receipt date, it is viewed in the user's preferred time zone. When these transactions flow into Financials, the transaction dates, in reference to the corporate time zone, are often used to derive the accounting dates. The accounting dates are also shown in the corporate time zone rather than the user's preferred time zone. These dates indicate the period into which the transaction will be posted, rather than the actual date on which the transaction occurred.

When drilling down from a financial document into the subledger transactions, users should be aware that the accounting dates are relative to the corporate time zone and will be shown as such, whereas some subledger transaction dates will be shown in the user's time zone

Implementation Considerations in Human Resources

In Oracle Human Resources, all objects that have the DateTrack feature (such as benefit plans or element types), use dates globally, without a time component. This means that, for example, the date associated with a benefit plan is relative to the time zone in which the benefit plan is used. Therefore, if a benefit plan is set up to start on 1st January, it will start on this date wherever it is used in the world.

Where entering dates and associated times in separate fields in Oracle Human Resources products, these dates and times are not affected if user-preferred time zone support has been enabled. User-preferred time zone support only affects dates that have a time component within the same field. These dates and times apply either to the local time zone in which they were entered, or to a context-specific location time zone. For example, the start and end times of an applicant interview are relative to the location of the interview.

The following sections describe the time and date behavior specific to Human Resources.

Human Resources Features Affected by User-Preferred Time Zone Support

This section describes the features specific to Human Resources that are affected if user-preferred time zone support has been implemented.

SSHR Employee Directory

The Employee Details page of the Self-Service Employee Directory displays the local time at the worker's location.

The Server Timezone profile option should be set to the time zone corresponding to the server time zone. The Employee Details page will then display the correct local time for the worker's location.

To use this feature without allowing user-preferred time zone support to be enabled for other windows and pages, carry out the following steps:

- Prevent the Client Timezone profile option being set at the user level. For more

information on profile options, see *Oracle E-Business Suite Developer's Guide* and Oracle E-Business Suite Security Guide.

- Use the Personalization Framework functionality to remove the Time Zone field from the Preferences page.

Time Zone Implementation Considerations in iRecruitment

When changing the status of an application in iRecruitment, the system records the exact date and time of the change as of the user-preferred time zone. However, if a retrospective status change is made, the system sets the date and time as the start of the day (midnight) of the standard corporate time zone.

When viewing the Application Status History, iRecruitment converts the date and time recorded for each status change to the user-preferred time zone.

Date Only Collection Parameters

The start and end collection parameters that use date only will display the date in the standard corporate time zone. These processes collect data from the start of the specified Start date to the start of the specified End date, as per the corporate time zone.

These parameters are unaffected by user-preferred time zone support.

Date and Time Collection Parameters

The time displayed in the parameters that display date and time is in the user-preferred time zone. The application converts the time to the standard corporate time zone before storing and running the program on the server.

Use of a user-preferred time zone can lead to unexpected results in some circumstances. For example, if the server is in the Pacific time zone, and a user enters a Collect To value of 03-JAN-2013 0700 in London (8 hours ahead), this represents 02-JAN-2013 at 2300 in the Pacific time zone. This discrepancy leads to the concurrent program collecting data up to the start of 2nd January, instead of up to the start of 3rd January as expected.

Human Resources Features Not Affected by User-Preferred Time Zone Support

The following features are unaffected if you implement user-preferred time zone Support. This section clarifies how dates and times in Human Resources products behave, whether or not user-preferred time zone support is enabled.

Effective Date feature

If the user profile DateTrack:Login Date (YYYY/MM/DD) has not been set, the Effective Date defaults to the date of the standard corporate time zone, as of the time the user logs into the HRMS application.

A user with security privileges can use the Alter Effective Date window to change the Effective Date. The Effective Date may need to be changed if the application is used in a time zone that is sometimes in a different date from the standard corporate time zone.

DateTrack Effective Start and End Dates

All DateTrack effective start and end dates have no time component. Therefore, no time zone adjustment is possible.

Timecard Entry and Timekeeper (OTL)

When an employee enters start and end times in the Timecard Entry and Timekeeper windows, these times are for the local time zone in which the employee actually worked. Therefore, a manager reviewing a Timecard can assume that the time shown on a Timecard is for the local time at the working location . This may be different from the worker's assignment location: for example, a worker may normally work in Los Angeles, but be called away to New York . For this reason, Timecard Entry and Timekeeper do not use the assignment's location time zone, since this does not account for people working away from an assignment location. This behavior is contrary to the behavior if user-preferred time zone Support is enabled, which displays times in the user's preferred time zone.

For daylight savings adjustments to be calculated accurately in the Self-Service Timecard Entry screen, the application server time zone should be set to a time zone that includes daylight savings adjustment similar to the user time zone.

Shift Patterns

When creating a shift pattern, start and end dates are not supplied. This enables a single shift pattern to be applied to a date in any location. A date is associated with a shift when a working shift pattern is assigned to an employee. The start and end times of the shift then apply to the time zone in which the employee works the shift.

Locations Time Zone Attributes

A location can optionally be associated with a specific time zone. If a time zone is set for a location, some windows display the time zone: the times displayed in such windows are in the location time zone.

Windows That Display Time Information

The following windows do not display dates and times in the user-preferred time zone, but in a context specific time zone, as described below for each window.

Employee Review Window

The Employee Review window has a Time Zone field. When selecting a location for the employee's review, the Time Zone field displays the time zone associated with that location. The review start and end times are in the location time zone, regardless of your location or the user's preferred time zone.

Applicant Interview Window

The start and end times are in the time zone of the interview location.

Book Event Window and Event Bookings Window

The windows display the event's start time and end time, along with the event's location. The event times are in the time zone of the event location.

Absence Window

This window displays the Project start and end dates and times, and the Actual start and end dates and times. These dates and times are in the time zone in which the employee works.

Work Incidents Window

This window displays the Incident and Reporting dates and times in the time zone of

the incident and reporting locations.

Normal Working Start and End Time

A number of windows display Normal Working Start and End Times. These include:

- Application
- Position Description
- Organization, as Work Day information
- Employee Assignment, on Standard Conditions

As these are time-only values, the time zone depends on the context in which the times are used. For example, an employee in New York could have the same position as an employee in Los Angeles . If the position start time is 0900, this is 9 am New York time for the worker in New York , and 9 am Los Angeles time for the worker in Los Angeles. These windows will not display times in the user's preferred time zone, even if user-preferred time zone support is enabled.

Approvals Management

All dates with a time component that display in the Approvals Management user interface are as of the standard corporate time zone. They are not affected by how the Time Zone profiles are set.

Whenever a future-dated rule is created or updated, it will be effective from the start of the day (midnight) as of the standard corporate time zone. Therefore, future-dated rules come into effect at the same point in time for all time zones, irrespective of when midnight occurs in each time zone.

When creating or updating a rule that is not future-dated, the rule comes into effect immediately, regardless of time zone. All future transactions (worldwide) are affected from the moment the rule is created or updated, regardless of the time zone of the transaction.

Learning Management

Learning Management organizes classes and events for specific dates and times. The following Learning Management windows and pages display date and time information based on the event location, rather than the user-preferred time zone:

- Development Event
- One Time Event
- Resource Booking (Other Details tab)
- Sessions window (PUI)

- Sessions page (OAF)

The following Learning Management windows also display date and time information:

- Self-Paced Class
- Scheduled Class

The date and time in these windows is based on the time zone of the class. The Learning administrator enters the class time zone when creating the class.

Implementation Considerations for Oracle iStore

Oracle iStore displays date/time in the corporate time zone only, and iStore UIs display the corporate time zone context when the user-preferred Time Zone feature is enabled at your site. For more information, refer to My Oracle Support Knowledge Document 403414.1, Oracle iStore Documentation Resources, Release 12.

Upgrade Considerations

There are no upgrade considerations for E-Business Suite customers upgrading from 11.5.10 CU2 or higher who have already enabled user-preferred time zone support. The upgrade will be transparent and the relevant functionality will not change.

For E-Business Suite customers who are upgrading to Release 12.2 from a release in which the user-preferred time zone feature was not enabled or not used, existing time zone practices must be taken into account when enabling user-preferred time zone support.

Prior to Release 11.5.10 (with CU2), users could deal with time zone differences in one of two ways:

- By entering data in the standard corporate time zone (as recommended by Oracle).

  The choices in this case are between:

  1. Enabling the user-preferred time zone feature and allowing users to continue working in the standard corporate time zone (as before the upgrade)
  2. Enabling the user-preferred time zone feature, setting a preferred time zone, and starting to interact with the system in this time zone. Failing to set a user-level time zone preference at user level will result in the standard corporate time zone continuing to be used.

- By entering data in the local time zone, and using a custom solution to resolve any issues that arise when comparing data across transactions initiated in different time zones.

  In this case, enabling the user-preferred time zone feature could result in unwanted date-with-time conversions, whereby local dates with time are converted to the

user-preferred time zone: this is because the system assumes that all dates with time are stored in the corporate time zone.

For further details, refer to *Oracle E-Business Suite Upgrade Guide: Release 11i to Release 12*.

Implementation Details

This section provides the technical considerations involved in implementing the user-preferred time zone feature.

Technology Stack Requirements

For user-preferred time zone support to operate correctly, all of the following must be true:

- The operating system time zone setting for the database server must be set to the standard corporate time zone.
- The database must be configured to use the time zone file `timezlrg.dat` rather than the `timezone.dat` file.
- The database must be started in the standard corporate time zone.
- Every application tier JVM (Java Virtual Machine) must be started in the standard corporate time zone. This can be achieved by setting the application tier operating system time zone to match the standard corporate time zone.
- Profile 'Server Timezone' (SERVER_TIMEZONE_ID) must be set at site level, and must be set to the same standard corporate time zone as the database.
- Profile 'Client Timezone' (CLIENT_TIMEZONE_ID) must be set at user level. This is applicable to Oracle Forms based UIs.
- Preference 'Timezone' must be set at user level. This is applicable to HTML based UIs.
- Profile 'Enable Timezone Conversions' (ENABLE_TIMEZONE_CONVERSIONS) must be set to 'Yes' (or 'Y') at site level.
- Profile 'Concurrent: Multiple Time Zones' (CONC_MULTI_TZ) must be set to 'No' (or 'N') at the site level.
- Environment variable FORMS_APPSLIBS must be set in the Forms tier
- Forms must be launched through the Personal Home Page or Navigator portlet.

These requirements are discussed in more detail below.

Time Zone File

The database must be started using the `timezlrg.dat` file, which contains the time zone definitions that are used within Oracle E-Business Suite. To do this in a UNIX environment, issue a command such as:

```
setenv ORA_TZFILE $ORACLE_HOME/oracore/zoneinfo/timezlrg.dat
```

before starting the database.

The database must also be started in the standard corporate time zone. To set this in a UNIX environment, issue a command such as:

```
setenv TZ <Timezone Code>  [For example, 'America/Los_Angeles']
```

You can verify your setup by running the following command in SQL*Plus:

```
select to_char(SYSDATE, 'DD-MON-RRRR HH24:MI:SS')
from dual;
```

to ensure the date with time returned are correct for the corporate time zone.

Applications Profiles and Preferences

The profile option Server Timezone (SERVER_TIMEZONE_ID) should be set at site level to the standard corporate time zone (the time zone in which the server has been set to run).

> **Caution:** This profile option should not be changed once set, as existing data will not be updated.

Users may specify their preferred time zone at user level. This is done by setting the 'Timezone' preference in HTML-based applications, or by setting the 'Client Timezone' profile option in Forms-based applications. As with most profile options, the user will need to log out and log back in for the change to take effect. The preferred time zone may be changed as often as needed.

The profile option 'Enable Timezone Conversions' (ENABLE_TIMEZONE_CONVERSIONS) has a default value of 'No' at site level. This will cause the applications to continue showing all dates in the corporate time zone. Setting this value to 'Yes' will enable the automatic conversion of all date with time fields to the user-preferred time zone.

> **Important:** Unless users are notified of this change, they may think that they are still operating in the corporate time rather than local time (or vice versa), and consequently enter or interpret data erroneously.

Note the behavior of the existing profile 'Concurrent: Multiple Time Zones' (CONC_MULTI_TZ). This was an older feature to handle batch processing. Setting this profile to 'Yes' alters the default value that appears for the Scheduled Start Date in the Submit Requests screen to SYSDATE-1. With the new user preferred time zone feature

enabled, this profile is no longer needed, and should have a value of 'No'.

Environment Variable FORMS_APPSLIBS

This environment variable controls multiple aspects of Oracle Forms in the Oracle E-Business Suite environment, and must be left unchanged from the installed setting.

Launching Forms-based Applications

The time zone feature is only available in Oracle Forms based user interfaces within Oracle E-Business Suite when the user logs in through the Personal Home Page or the Navigator portlet. Direct launching of Forms, for example by typing a URL into the browser address line, is supported only for bootstrap purposes, and will not enable the time zone feature, or other features such as language settings and date formats.

9

Managing Globalization

Overview of Globalization Support

This chapter describes some of the features of globalization (formerly internationalization) support in Oracle E-Business Suite. Topics covered in this chapter include language values used in user sessions, behavior of month name abbreviations, and multilingual external documents. In addition, the Oracle E-Business Suite windows for Languages, Natural Languages, and Territories are described.

For additional information on globalization concepts and features in Oracle E-Business Suite, see My Oracle Support Knowledge Documents 393861.1, "Oracle E-Business Suite Release 12 Globalization Guide", and 393320.1, "Oracle E-Business Suite Release 12 Internationalization Update Notes". Also see *Oracle E-Business Suite Concepts Guide*.

Language Values for User Sessions

Language Values for User Sessions using AppsLocalLogin.jsp

This section describes how the language is determined for a user's session and other interactions with the system.

Login Page Language

If the language is a licensed Oracle E-Business Suite language, the initial login page language is determined from one of the following sources, in the order shown:

- A language value passed on the AppsLocalLogin.jsp URL
- The language used in an earlier login session
- The browser language

If the language is not a licensed Oracle E-Business Suite language, the initial login page

is in the Oracle E-Business Suite base language.

> **Note:** The default for Single Sign-On (SSO) SSO logins will fall back to browser language preferences, in order. The default for non-SSO logins will fall back to the base language if the first language preference is not available.

The user can change the login page language selection by choosing one of the other language name icons on the login page. The language name icons on the login page represent the active languages in the Oracle E-Business Suite installation. If the user selects an alternative language, the login page is refreshed in that alternative language. Logging in and logging back out will display a URL with the correct language value. The URL can be bookmarked.

> **Important:** In Oracle E-Business Suite Release 11.5.10, the language name icons could be hidden by setting a Local Login Mask profile option. In Release 12, this profile option is no longer available. To hide the language name icons, use the Oracle Application Framework Personalization feature.

Login Page Language and Runtime Session Language

Users must ensure that the language used for the login page is the language they want to use for their Oracle E-Business Suite runtime session. The runtime session will start in the language used in the login page. After logging in, users can, if they wish, change the runtime session language from the Preferences screen. By default, the ICX:Language profile value is not used to determine the runtime session language when logging in through the login page. However, this behavior can be modified by setting the "Applications Override SSO Server Language" profile option.

ICX:Language Profile Option

The ICX:Language profile option is used for the following:

- Default language to use for offline with other users on the system. For example, the language to use for an e-mail notification to be sent to a user on the system, and which may be read offline from the system.

- Default language to use for other Oracle E-Business Suite login methods besides the login page. For example, the language to use for the session language when logged in through JDeveloper.

- Session language when the "Applications Override SSO Server Language" profile option is set to "Override SSO Server Language".

The ICX:Language profile is normally not set at user level when a given user initially

starts using Oracle E-Business Suite. Regardless of whether a value exists for the profile at user level, the language chosen on the login page (the runtime session value) is not automatically saved to the ICX:Language profile value for the user. The user can either specifically set a value for the ICX:Language profile at user level after logging in, or continue to inherit the value from the higher profile value levels. A value always exists for the ICX:Language profile at site level.

Preferences Page Language Support

The current values for the runtime session language and the saved ICX:Language profile value for the user are displayed on the Preferences page.

The following updates are allowed:

- Change only the runtime session language
- Change only the saved ICX:Language profile value for the user
- Change both the runtime session and the ICX:Language profile values to either the same values or different values

Language Value from Login External to Oracle E-Business Suite

This section describes language value behavior when logging into Oracle E-Business Suite from non-Oracle E-Business Suite-based login systems.

For Oracle E-Business Suite, the language used in such a non-Oracle E-Business Suite system cannot be relied on to determine the runtime session language for the Oracle E-Business Suite session. That language may not be an Oracle E-Business Suite-supported active language, and it may not be the preferred language of the user for the runtime Oracle E-Business Suite session. Therefore, the ICX:Language profile value for the user determines the Oracle E-Business Suite runtime session for that user.

Once the Oracle E-Business Suite runtime session is initiated, the user can use the Preferences page to change the runtime session language for the session. The user can also choose to change his saved ICX:Language profile value.

For an Oracle E-Business Suite environment integrated with Oracle WebCenter or Oracle Portal, the Oracle E-Business Suite session will inherit the portal language if the session is launched from that portal. If the portal language is not supported or enabled in Oracle E-Business Suite, the Oracle E-Business Suite session language will fall back to the Oracle E-Business Suite base language.

Language Values for Oracle Workflow Notifications

Oracle Workflow email notifications use the user-level language set in the Preferences page and also stored in the ICX:Language profile option.

Date Formats in NLS Implementations

The storage space for month name abbreviations is limited to 3 bytes. For some language and character set combinations, the month name abbreviation requires more than 3 bytes. For this reason, when a language is used in which any month name abbreviation exceeds 3 bytes in the current character set, all month name abbreviations for that language are automatically replaced by numeric representations that conform to the 3-byte space limit.

The languages affected in a UTF8 implementation are: Albanian, Bulgarian, Canadian French, Croatian, Czech, German, Greek, Hungarian, Icelandic, Polish, Russian, Simplified Chinese, Slovak, Thai, Traditional Chinese, Turkish, and Vietnamese.

The languages affected in their local character sets are: Simplified Chinese, Thai, Traditional Chinese, and Vietnamese.

Support for Calendars

By default, the calendar in the Date Picker in an application shows the Gregorian calendar. However, users can change their calendar display by setting profile options as described below.

The Hijrah and Thai Buddhist calendars are supported for Oracle Forms-based and Oracle Application Framework applications, as well as Oracle Workflow notifications, Gantt charts and other graphs, and Oracle XML Publisher reports. The support is added as an output (display) format only. Internal date object representation stays as for the Gregorian calendar. Concurrent processing and descriptive flexfields also support displaying dates in these calendars based on user preferences.

The following features are supported:

- Users can choose a preferred calendar by setting the profile option FND: Forms User Calendar (FND_FORMS_USER_CALENDAR).
- For the Hijrah calendar, users can display both Hijrah and Gregorian calendars by clicking on a button in the calendar window.
    - With the two calendars displayed, a user can pick a date from either the Hijrah or Gregorian calendar in Oracle Forms-based products.
    - The displayed Hijrah and Gregorian calendars are synchronized.
    - Initially, only the preferred calendar is displayed in the Date Picker.
- Users can choose the first day of week in a Date Picker by setting the profile option FND: Calendar Week Start Day (FND_CALENDAR_WEEK_START_DAY).
- Dates in a secondary calendar can be displayed as tooltip text in Oracle

Forms-based products if the profile option FND: Tooltip Calendar (FND_TOOLTIP_CALENDAR).

FND: Forms User Calendar Profile Option

Users can set the FND: Forms User Calendar profile option to their preferred calendar. Valid values are:

- Arabic Hijrah
- English Hijrah
- Gregorian
- Thai Buddha

By default, the user calendar displays the Gregorian calendar, but if this profile option is set to another value, that calendar is used.

FND: Calendar Week Start Day Profile Option

Users in countries can set the FND: Calendar Week Start Day profile option to their preferred first day of the week (Sunday is the default first day of the week).

FND: Tooltip Calendar Profile Option

Users can set the FND: Tooltip Calendar profile option to a calendar other than the preferred calendar. Within the Date Picker, a given date will be displayed in this calendar's format as a tooltip.

Note: This feature applies to Oracle Forms-based products only.

Numeric Format Support

In addition to the decimal period (dot) and comma, the space ( ) and single quote (') characters are supported as number group separators. Users can choose these number group separators from the "Number Format" LOV in the Preferences Page.

Examples of formats that are supported are listed in the table below:

Group Separator	Decimal Separator	Example
Comma	Period	1, 000.00

Group Separator	Decimal Separator	Example
Period	Comma	1.000,00
Space	Comma	1 000,00
Single Quote	Period	1'000.00
Space	Period	1 000.00
Single Quote	Comma	1'000,00

Note: The JTF applications support only the decimal period and comma as number group separators.

Lightweight MLS

The Lightweight MLS feature allows additional languages to be enabled without the need to apply the corresponding NLS patches. In releases prior to Release 12.1.3, it was necessary to apply the relevant NLS patches before additional languages could be used.

Lightweight MLS enables Oracle E-Business Suite to run multiple languages in the same system, with the various languages being employed by different users as desired. Such a system can simultaneously support a mixture of languages with full translation (NLS patches applied) and languages without translation (Lightweight MLS, with no NLS patches applied).

The Lightweight MLS feature provides the capability to enter and maintain language data through an English user interface; for example, setup data or customer data such as item or item description. In addition, users can set up custom reports or templates and then generate language reports with the language data. You can also move to a full translation at any time by applying the NLS patch for the relevant language.

For more information, refer to My Oracle Support Knowledge Document 1077709.1, *Using Lightweight MLS With Oracle E-Business Suite*.

Multilingual External Documents

Oracle E-Business Suite ships with a set of external documents, or those documents directed toward your customers and trading partners, for which we model the data multilingually. Any document for which the data model is multilingual can be submitted, through a single request, to run in one language or in any subset of the installed languages. Your Italian customer, for example, can receive invoices printed in

Italian, while your Korean customer receives packing slips printed in Korean.

Oracle Shipping

- Bill of Lading
- Commercial Invoice
- Pack Slip

Oracle Order Management

- Price List
- Sales Order Acknowledgment

Oracle Receivables

- Dunning Letter Print
- Print Statements
- Transaction Print

Oracle Purchasing

- Printed Change Order Report (Landscape)
- Printed Change Order Report (Portrait)
- Printed Purchase Order Report (Landscape)
- Printed Purchase Order Report (Portrait)
- Printed RFQ Report (Landscape)
- Printed RFQ Report (Portrait)
- Dispatch Purchase Order

Oracle Sourcing

- Negotiation PDF

Oracle Payables

- Invalid PO Supplier Notice
- Prepayment Remittance Notice
- Print Invoice Notice
- Supplier Open Balance Letter

Oracle Human Capital Management (Human Resources)

- Full Person Details
- Full Applicant Details
- Full Assignment Details
- Full Work Details

Oracle Payroll

- Check Writer
- Deposit Advice
- Third Party Checks

Translations Window

In windows with an enabled translation icon, users can click on the icon to bring up a Translations window to enter or update translated values for specific records in the database. See: Creating Translations for a Record, *Oracle E-Business Suite User's Guide*.

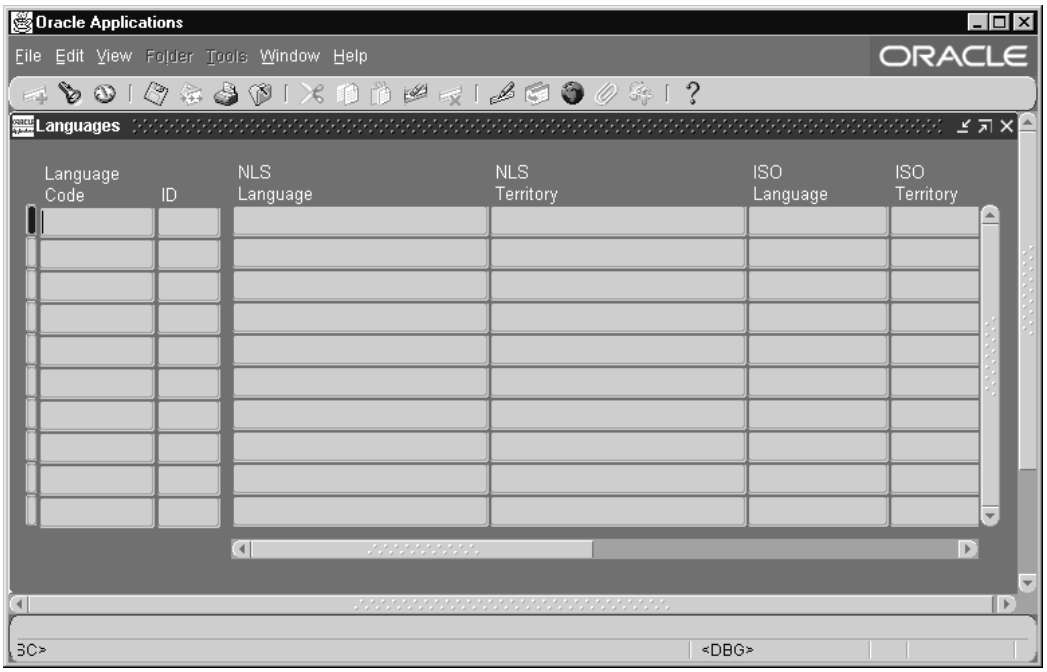
Users can enter translations for multiple languages at once in the Translations window. Translated values should be entered using the Translations window.

Users should be aware of which fields are allowed to be translated. Users can click on the translation icon to identify if a field is translatable. If a user types into a field that was not meant to be translated, the value in the source language will be overwritten, and the translated value will be the only entry for that field. All users will see the data in that language, and the value in the original language will be lost.

Currencies Window

For information on how to define currencies using this window, see: Defining Currencies, *Oracle General Ledger User's Guide*.

Languages Window



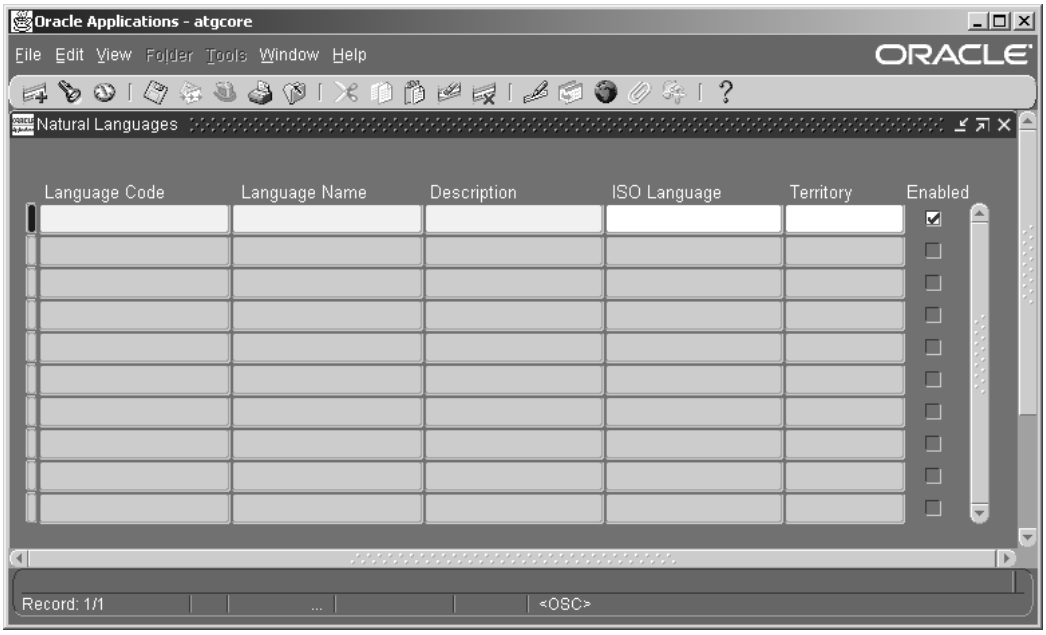
Use the Languages window to review information about the languages available for use in Oracle E-Business Suite.

Languages Record

Each record includes the Language Code, ID, NLS Language, NLS Territory, ISO Language, ISO Territory, Default Code Set, Installed flag, Local Date Language, UTF8 Date Language, and Description.

The columns Local Date Language and UTF8 Date Language are used in date rendering. For some languages, the length in bytes of the month short name is longer than the prepared buffer in a UTF8 database; for these languages, the UTF8 Date Language is used as the date language in UTF8 environments. In databases using other character sets, the Local Date Language is used.

Natural Languages Window



The Natural Languages window allows you to review information about the languages available for use in Oracle E-Business Suite.

The Natural Languages table allows you to associate the name of a language with another entity, such as a person or company.

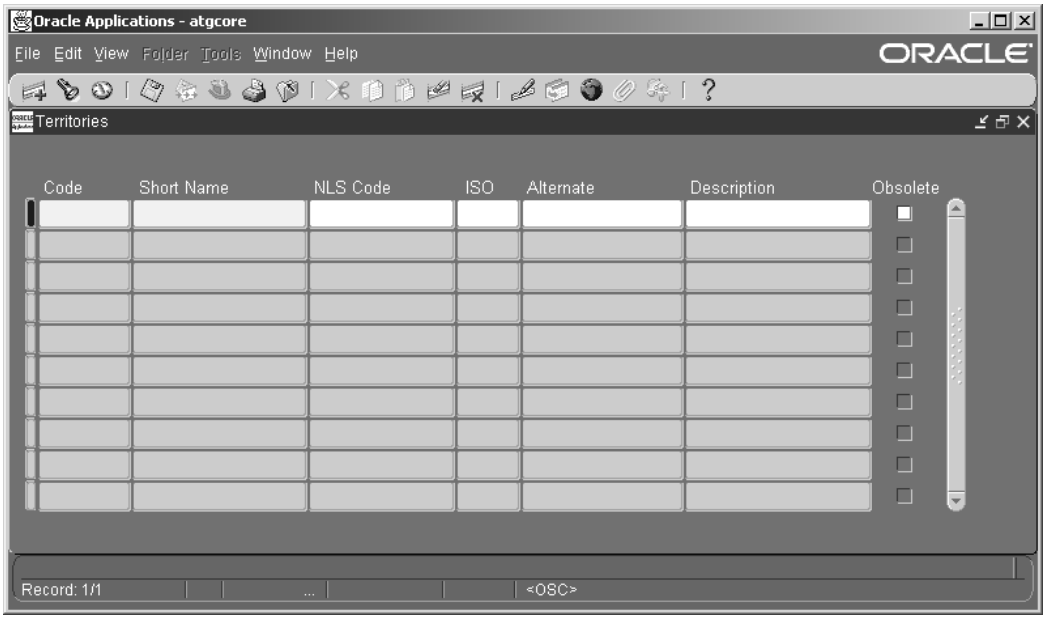
- You can add new languages in this window. Prefix the language code for new languages with an 'x'.
- You can update the enabled status of seeded languages.
- You cannot delete saved records.

Natural Languages Record

Each record includes the Language Code, Language Name, ISO Language, and Territory for each language.

You can use Oracle Applications Manager (OAM) License Manager to enable a licensed language..

Territories Window



Use the Territories window to review and modify information for the country values used in Oracle E-Business Suite.

Territories Block

Each record includes the two-letter uppercase territory Code such as "US", the Short Name for the territory such as "United States", the NLS Code, the ISO numeric entity code, an Alternate territory code, and a longer description (Description), such as "United States of America".

Normally you would not want to update the seeded data that comes with your products, but you may wish to modify the way the country is represented in List of Values through out your applications.

You can update the description of the Territory to change the territory value displayed in List of Values used in Oracle E-Business Suite products.

10

Defining Document Sequences

What is a Document Sequence?

A document sequence uniquely numbers documents generated by an Oracle E-Business Suite product. Using Oracle E-Business Suite, you initiate a transaction by entering data through a form and generating a document, for example, an invoice. A document sequence generates an audit trail that identifies the application that created the transaction, for example, Oracle Receivables, and the original document that was generated, for example, invoice number 1234.

Document sequences can provide proof of completeness. For example, document sequences can be used to account for every transaction, even transactions that fail.

Document sequences can also provide an audit trail. For example, a document sequence can provide an audit trail from the general ledger into the subsidiary ledger, and to the document that originally affected the account balance.

Document sequences generate audit data, so even if documents are deleted, their audit records remain.

Related Topics

Defining a Document Sequence, page 10-1

Defining Document Categories, page 10-4

Assigning a Document Sequence, page 10-5

Document Numbering vs. Document Entry, page 10-7

Defining a Document Sequence

To define a sequence, you select a sequence name and an application to "own" the sequence.

- A sequence can number documents stored in database tables belonging to its

owning application.

- Audit records for a sequence are stored in the application's audit table, titled *Application Short Name*_DOC_SEQUENCE_AUDIT. For example, the audit table for a sequence owned by Oracle Payables is AP_DOC_SEQUENCE_AUDIT.

  Important: Your database administrator must grant access to an application's audit table for all ORACLE usernames associated with responsibilities that will use the sequence (responsibilities that access forms using the sequence).

You can set start and end dates for when the sequence is available. The start date defaults to the current date. By default, there is no end date, so the sequence definition does not expire.

You can choose whether a sequence numbers documents automatically, or accepts numbers manually entered by a user.

Automatic, Gapless, and Manual Numbering

Automatic numbering assigns a unique number to each document as it is generated. Automatic numbering is sequential by date and time of creation.

Gapless numbering also automatically generates a unique number for each document, but ensures that the document was successfully generated before assigning the number. With Gapless numbering, no sequence numbers are lost due to incomplete or failed document creation.

Important: We recommend that you choose this type only when gapless numbering is essential, as it may affect the performance of your system.

Manual numbering requires a user to assign a unique number to each document before it is generated. With manual numbering, numerical ordering and completeness is not enforced. Users can skip or omit numbers when entering the sequence value.

Automatic Numbering - Initial Value and Message Display

If you define a sequence to automatically number documents, you can:

- Enter an initial value for your sequence. The default is "1".

- Choose whether you want to display a message when a document is generated, telling the user the name of the sequence, and the sequence value (document number).

Two examples of sequence definitions, one with automatic numbering and the other with manual numbering, are represented in the table below.

Field in Document Sequences form	EXAMPLE 1 Sequence with Automatic Numbering	EXAMPLE 2 Sequence with Manual Numbering
(Sequence) NAME	AUTOPAY	ADJUSTMENTS
(Owning) APPLICATION	ORACLE PAYABLES - Sequence can number documents stored in an Oracle Payables database table.	ORACLE RECEIVABLES - Sequence can number documents stored in an Oracle Receivables database table.
EFFECTIVE DATE - START	CURRENT DATE & TIME (Default value)	OCT-01-94 User defines sequence "Adjustments" not to be available until Oct 1, 1994.
EFFECTIVE DATE - END	Field left blank. Sequence does not expire.	DEC-31-94 User defines sequence "Adjustments" to no longer be available after Dec 31, 1994.
(Numbering) TYPE	AUTOMATIC - Unique numbers are automatically generated in sequence. GAPLESS No omissions or gaps in numbers are possible, due to a rollback if the document creation is unsuccessful.	MANUAL - User must enter a unique number before transaction can be completed, and document is generated. User may skip or omit numbers.
INITIAL VALUE	1 (Default value) User could enter their own initial value, for example, 5700.	**Not Available when numbering type is Manual.**
MESSAGE	YES - When a document that is automatically numbered is created, a message displays the sequence name and the sequence value (document number).	**Not Available when numbering type is Manual.**

Related Topics

What is a Document Sequence?, page 10-1

Defining Document Categories, page 10-4

Defining Document Categories

Document categories organize documents into logical groups.

- A document category (also called a document type) is one of the rules you use to define which documents a sequence assigns numbers to.
- You can separately number each document category by assigning a different sequence to each category.

A document category identifies the database table that stores documents resulting from transactions your users enter.

- When you assign a sequence to a category, the sequence numbers the documents that are stored in a particular table.

Use categories to more precisely classify your documents. For example, you can categorize accounts receivable invoices into several different categories, such as:

- Chargebacks
- Deposits
- Guarantees
- Debit Memos
- Credit Memos
- Sales Invoices
- Customer Service Invoices

Similarly, you can categorize accounts payable or purchase invoices into several different categories, such as:

- Standard
- Expense Report
- Prepayment
- Interest
- Credit Memo
- Debit Memo

Related Topics

What is a Document Sequence?, page 10-1

Defining a Document Sequence, page 10-1

Assigning a Document Sequence, page 10-5

Assigning a Document Sequence

Before you can assign a sequence to number documents, you must define which documents are to be numbered.

Sequences versus Assignments

Defining a sequence is different from assigning a sequence to a series of documents.

- A sequence's definition determines whether a document's number is automatically generated or manually entered by the user.
- A sequence's assignment, that is, the documents a sequence is assigned to, is defined in the Sequence Assignments form.

Defining Documents for numbering by Assigned Sequences

You specify a combination of four rules that define any given document for assignment to a specific sequence name.

You can then assign a different (numbering) sequence to each document definition.

The four rules, that when combined, define what documents a selected sequence assigns numbers to are:

Application

You select the application that generates the documents you wish to number.

For example, to number sales invoices, you select Oracle Receivables.

Category

You select a document category to identify a logical subset of documents.

For example, if you do not want to number all invoices in Oracle Receivables, you can choose to number only the category of sales invoices.

A category identifies a table that stores transactions entered (documents generated) using an Oracle E-Business Suite application.

The Category values you can choose from to define a

document are dependent upon the application you select.

Set of Books

You select the chart of accounts for your business that is affected by the documents you wish to number. You may optionally enable this rule through the Document Flexfield.

Method

You select the method that your documents are entered, automatic or manual. You may optionally enable this rule through the Document Flexfield.

Automatic is when a concurrent process, such as an external program, is set up to enter transaction data into an Oracle E-Business Suite product.

Manual is when a document is manually entered using a form in an application.

Assignment of Sequences to Document Definitions

For each unique document definition there can only be one active sequence assignment. A document definition consists of the Application, Category, and the optional Document Flexfield segments Set of Books and Method

> **Important:** When assigning sequences to a document definition, each active sequence can be assigned to only one unique combination of application and category (i.e., application table).

Active Assignments and Active Sequences

An active sequence assignment does not have a post dated end date. That is, the assignment's end date is not before the current date.

- An active sequence *assignment* either has no end date, or an end date that is not before the current date.
- A sequence assignment and its dates of effectivity are defined on the Sequence Assignments form.

A sequence *definition* must be active as well. That is, the sequence definition's end date (as opposed to its assignment's end date) must not be before the current date.

- A sequence definition and its dates of effectivity are defined on the Document Sequences form.

When you define a document sequence, you give the sequence a name, and define how the sequence numbers each document by:

- Choosing whether numbers are automatically generated in sequence, or entered

manually by the user.

- Entering the initial value or first number in the sequence.

Related Topics

What is a Document Sequence?, page 10-1

Defining a Document Sequence, page 10-1

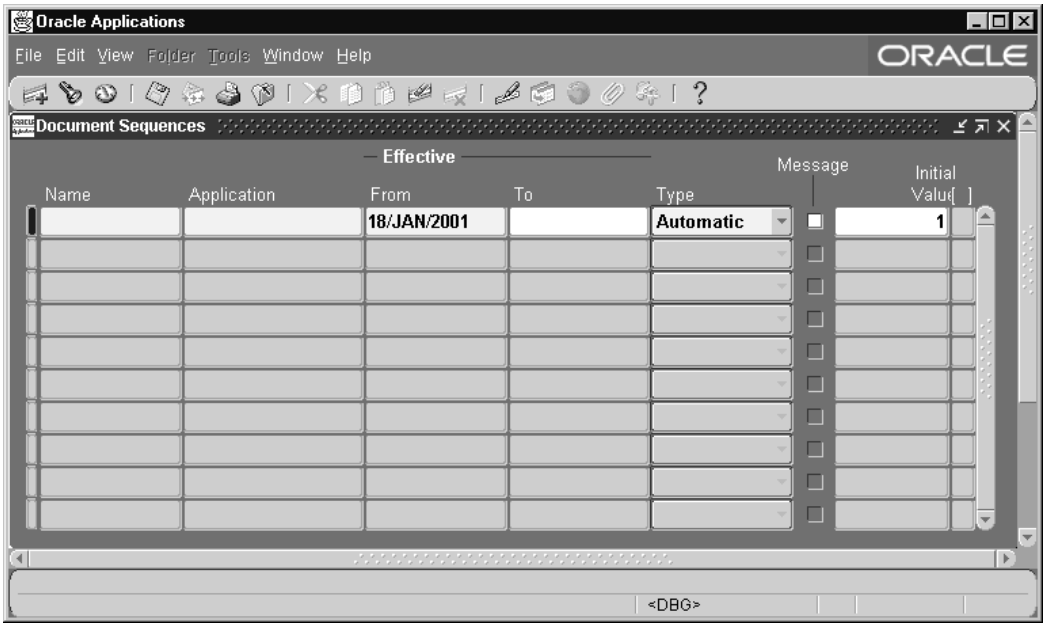
Defining Document Categories, page 10-4

Document Numbering vs. Document Entry, page 10-7

Document Numbering vs. Document Entry

Do not confuse the type of document numbering a sequence employs, which can be automatic or manual, with the method of entering documents, which is also defined as either automatic or manual.

Document Sequences Window



Name a new document sequence and define how the sequence numbers each document.

A document sequence uniquely numbers documents generated by an Oracle E-Business Suite product (for example, invoices generated by Oracle Receivables). Using the Sequence Assignments window, you assign your sequence to number only those documents that satisfy rules you define.

A document sequence numbers documents generated by an Oracle application, for example, journal entries generated by Oracle General Ledger.

Document sequences ensure that every document your users create can be accounted for. See: Sequences Assignments, page 10-12.

Document Sequences Block

Define the name, type of numbering scheme, effective dates, and initial value for your document sequence.

Name

Once entered, sequence names cannot be changed.

Application

Once selected, the application associated with your sequence cannot be changed.

Audit records for your sequence are stored in the application's audit table, titled *Application Short Name*_DOC_SEQUENCE_AUDIT. For example, the audit table for a sequence owned by Oracle Payables is AP_DOC_SEQUENCE_AUDIT.

Effective From/To

Enter the dates on which your document sequence takes effect/is no longer enabled. The Start on field automatically defaults to the current date, and once a sequence is defined, the start date cannot be changed. If you leave the End on field blank, your document sequence does not expire; and if you enter an end date and define your sequence, the end date cannot be modified later. If there is no end date defined and there are no active assignments for a sequence, you can disable the sequence by entering the current date as the end date. Once disabled, a sequence cannot be reactivated.

Type

Once defined, you cannot change the type of document numbering sequence.

Automatic Sequentially assigns, by date and time of creation, a unique number to each document as it is generated.

Manual Manual numbering requires a user to assign a number to each document before it is generated. You must enter unique values. However, please note that numerical ordering and completeness is not enforced.

Important: The Automatic-By-User type is currently not supported, and is reserved for future use.

Warning: The Gapless Numbering type is valid only in the context of certain localizations. We recommend that you choose this type only after consulting with Worldwide Support, as it may affect the performance of your system.

Message

Check the Message check box if you want each document to display a message (in the message line near the bottom of the screen) informing the user of the sequence name and value (number).

This check box only applies to sequences with the automatic type of numbering. Messages appear only on form displays, and are not written to a request's log file.

Once a sequence is defined, the message choice cannot be changed.

Initial Value

Enter a value for the first document in your sequence. This field only applies to sequences with automatic or gapless numbering type. Sequence numbers should not be greater than eight (8) digits.

If you leave this field blank, the first document is automatically assigned a value of "1".

Once a sequence is defined, this initial value cannot be changed.

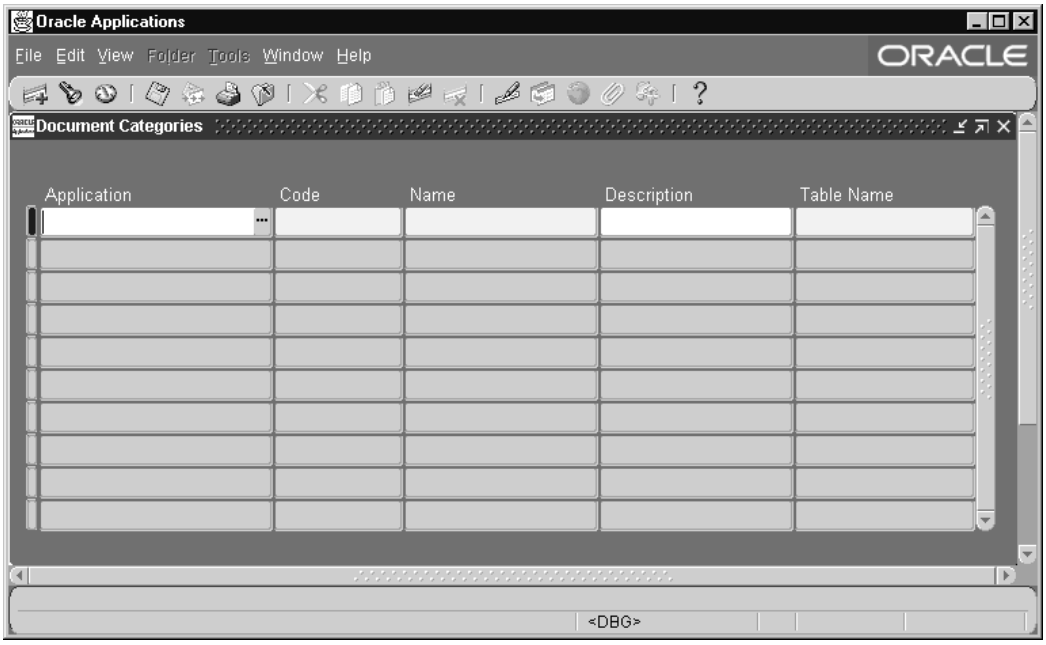
Related Topics

What is a Document Sequence?, page 10-1

Document Categories, page 10-10

Sequence Assignments, page 10-12

Document Categories Window



Define categories for your documents in order to divide your documents into logical groups, which you can number separately by assigning different sequences.

A document sequence uniquely numbers each document the sequence is assigned to.

- Using the Sequence Assignments form, you assign your sequence to number only documents that satisfy rules you define.
- Document category, or type, as it may be titled on some forms, is one of the rules that define which documents a sequence assigns numbers to.

Each category identifies a table that stores documents resulting from transactions your users generate.

- When you assign a sequence to a category, the sequence numbers the documents that are stored in the table.

Document Categories Block

Name a document category and associate a table with the category.

When you enter this block, Oracle automatically queries for any existing document categories.

Application

Once a category is defined, you cannot change the choice of application. Only tables belonging to the selected application can be assigned to a category.

Code

Category code must be unique within an application. Once a category is defined, you cannot update its code.

Name

You can update the name, if you wish. For example, if the category name is predefined, you can change the name to a more familiar value.

Description

You can update the description, if you wish. For example, if the category description is predefined, you can change the description to a more familiar value.

Table Name

Select the name of the table that stores the documents you want to identify by your category.

- When the sequential numbering feature checks for completeness or generates a report, it locates the category's documents in the table.
- Only tables belonging to the application associated with the category can be chosen.
- Once a category is defined, you cannot change the choice of table.

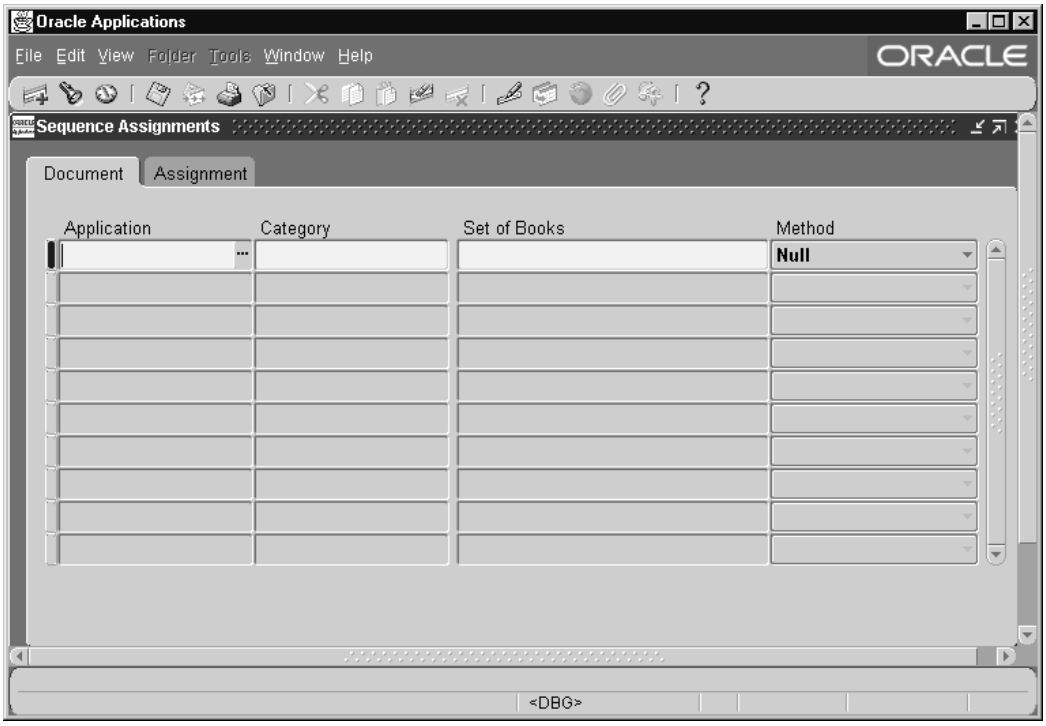
Related Topics

What is a Document Sequence?, page 10-1

Document Sequences, page 10-7

Sequence Assignments, page 10-12

Sequence Assignments Window



Define which documents a document sequence can number, and then assign the document sequence to your definition.

A document sequence numbers documents generated by an Oracle E-Business Suite product (for example, invoices generated by Oracle Receivables).

Documents can be defined by the application that generates them and their category (the table in which they are stored). Additional fields appear when the optional rules for defining documents (Set of Books and Method of document entry) are enabled.

Besides entering a document definition and assigning a sequence to it, you can, if you wish, enter effective dates for the assignment.

Prerequisites

Define the document sequence using the Document Sequences window. See: Document Sequences, page 10-7.

Sequence Assignments Block

Specify documents by the application that generates them and the category of the document (table where the documents are stored). You can also include in your document definition the set of books they affect, and the method by which the

document is entered.

Once a document definition is entered, you select a sequence to assign it to, and if you wish, enter effective dates for the assignment.

There can only be one active sequence assigned to each unique combination of Application, Category, Set of Books, and Method. The last two criterion are optional, and are set in the Document Flexfield.

However, the same sequence, the same numbering scheme, and initial value can be assigned to more than one combination of Application, Category, Set of Books, and Method as long as the Application and Category remain the same.

Application Select the application that generates the documents you wish to number.

For example, to number sales invoices, you select Oracle Receivables.

Category Select a document category to identify a logical subset of documents.

For example, if you do not want to number all invoices in Oracle Receivables, you can choose to number only the category of sales invoices.

Assignment Region

Since the effective dates for an assignment must fall within the sequence's start and end dates, the list of available sequences depends on the start and end dates specified for the assignment.

Start Date/End Date

Enter the dates on which the sequence assignment to your document definition takes effect/is no longer enabled. The Start Date field automatically defaults to the current date, and once a sequence assignment is defined, the start date cannot be changed.

If you leave the End Date field blank, your sequence assignment does not expire; and if you enter an end date and define your sequence assignment, the end date cannot be modified later.

If there is no end date defined and there are no active assignments for a sequence, you can disable the sequence assignment by entering the current date as the end date. Once disabled, a sequence assignment cannot be reactivated.

Sequence

Select a sequence to assign to your document definition. The sequence's application and the document's application must be the same.

Once you define a sequence assignment, the sequence name cannot be updated later.

If you want to disable the sequence assignment and assign a new sequence to the document definition (Document Flexfield combination), you must first, enter an End Date to disable the current sequence assignment, then, second, create a new record (row) for the new assignment.

Document Flexfield

The Document Flexfield may consist of none, one or two segments.

Set of Books Select the chart of accounts for your business that is affected by the documents you wish to number.

Method Select the method that your documents are entered, automatic or manual.

Automatic is when a concurrent process (e.g., an external program) enters transaction data into an Oracle E-Business Suite product, which generates documents.

Manual is when a document is manually entered using a form in an application.

Once defined, a Document Flexfield definition cannot be updated. You may not define additional segments for the Document Flexfield.

> **Important:** To enable this descriptive flexfield, use the Descriptive Flexfield Segments window. Select the application *Application Object Library*, and the title "Document Flexfield". Be sure to unfreeze the flexfield; then, navigate to the Segments window and enable the segments. Freeze your flexfield after you set it up, and save and compile the new definition.

Related Topics

Descriptive Flexfields Concepts, *Oracle E-Business Suite Flexfields Guide*

What is a Document Sequence?, page 10-1

Document Sequences, page 10-7

Document Categories, page 10-10

11

Applications DBA System Configuration Tools

About System Configuration

During a new installation or an existing system upgrade, you define the configuration of your system by supplying information such as:

- Database type
- Database SID
- Domain name
- Top-level directories
- Licensing type
- Country-specific functionalities
- Internationalization settings
- Applications node services

These values are propagated to the individual system configuration files, and also stored in a central repository file called a *context*.

Your system configuration can be changed as needed by using one of several utilities designed to report on and manage the configuration information. Some of these utilities are accessed from the command line and some are Web-based.

Web-Based Configuration Utilities

The following utilities are Web-based. You access them through Oracle Applications

Manager (OAM).

AutoConfig

System configuration parameters are stored and managed by AutoConfig. It is the main configuration engine used by Rapid Install, which supplies configuration information to AutoConfig, which, in turn, stores the configuration for each system node in a node-specific configuration file call a context file.

You can also use AutoConfig independently of a Rapid Install operation to view and edit the individual configuration parameters that define your system. At any time after the initial installation, you use the Configuration Editor in Oracle Applications Manager to update various system settings.

> **Note:** For more information, see AutoConfig in *Oracle E-Business Suite Concepts*.

License Manager

Products, country-specific functionalities (localized products), and languages that you license or begin to use after the initial installation must be registered as active in order to be included in various system maintenance tasks. Using License Manager, you can create reports about currently registered products and register additional products, country-specific functionalities (localized products).

> **Note:** For more information, see License Manager in *Oracle E-Business Suite Maintenance Guide*.

Command Line Configuration Utilities

These AD utilities are run from the command line. They are more fully described later in this chapter.

AD Splicer

Splicing refers to the process of adding a product that was not included in a base release to the products in an existing system. AD Splicer modifies the APPL_TOP and database so that AutoPatch and AD Administration recognize the product as valid.

> **Note:** For more information, see AD Splicer, page 11-3 in this chapter.

File Character Set Converter

This utility converts the character set of individual files (those not included in processing performed by AD Administration, AutoPatch, or Rapid Install) to the

character set used in your system.

AD Splicer

AD Splicer performs the same product registration function as License Manager. However, it registers *off-cycle products* (those that are released between release update packs) as active in your system. This process of "splicing" modifies the APPL_TOP and database so that AutoPatch and AD Administration recognize the off-cycle product as a valid product for a specific release.

Patches that contain off-cycle products also contain the control files that AD Splicer needs to register the product. The patch also contains a readme file that describes how to install the new product(s).

AD Splicer Control Files:

There are two kinds of AD Splicer control files: product definition and product configuration. You must customize the product configuration file, then copy it and the product definition file to APPL_TOP/admin before you run AD Splicer.

Product Definition Files:

There are two product definition files per spliced product: <prod>prod.txt and <prod>terr.txt. These files define the product and the associated language information and must not be edited. For example, the product definition files for Oracle Sales Analyzer (zsa) are zsaprod.txt and zsaterr.txt.

Product Configuration:

The newprods.txt file acts as a template to define necessary parameters for a spliced product.

The following is an example of the product configuration file for Oracle Sales Analyzer (zsa):

```
product=zsa
base_product_top=*APPL_TOP*
oracle_schema=zsa
sizing_factor=100
main_tspace=*Product_Name*D
index_tspace=*Product_Name*X
temp_tspace=*Temporary_Tablespace*
default_tspace=*Product_Name*D
```

You may need to edit some of the values for the parameters in this file. Refer to the following table for more information. Do not change the order of the entries in the product configuration file: they must appear exactly as shown in the example.

Product Configuration Parameters

Parameter	Description
product=	Do not edit this entry. The product abbreviation <prod> is already set, and must match the <prod>prod.txt and <prod>terr.txt control files for this product. Most internal references use <prod>.
base_product_top=	Identifies the base directory that contains the product's files. The default value, APPL_TOP, means the product's files are written in the directory your APPL_TOP environment is set to. If you want to write the product files to another directory, replace the APPL_TOP value with the full directory path.
oracle_schema=	Identifies the Oracle schema where database objects for the product are created. The default Oracle schema is the same as the product abbreviation. You can change this if you want to put the product's database objects in a different schema. Moving a product's objects from one schema to another involves export/import and updates to internal Oracle E-Business Suite tables, so choose your initial schema carefully.
sizing_factor=	Identifies the sizing factor Oracle E-Business Suite uses when creating tables and indexes for this product. The default value of 100 means 100%. The product's tables and indexes are created with the default sizes determined by Oracle. We recommend you accept the default sizing factor.

Tablespaces:

Release 12 uses the Oracle Applications Tablespace Model (OATM), so you do not need to supply AD Splicer with parameters for identifying tablespaces.

> **Note:** For more information, see Tablespace Management in *Oracle E-Business Suite Concepts*.

The AD Splicer Interface:

AD Splicer is a command line utility. It does not use menus or input screens.

Running AD Splicer Interactively:

An example of running AD Splicer interactively is when adding a new off-cycle product, as described in *Oracle E-Business Suite Maintenance Guide*. The procedure is as follows:

1. Set the environment.

   You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment, *Oracle E-Business Suite Maintenance Guide* for the basic steps.

2. Start AD Splicer.

   Start AD Splicer with the appropriate command for your operating system:

UNIX:

```
$ cd $APPL_TOP/admin
$ adsplice
```

Windows:

```
C:\> cd %APPL_TOP%\admin
C:\> adsplice
```

You must run AD Splicer for each APPL_TOP and database combination so that the Applications utilities recognize the product as being spliced properly into the database.

Running AD Splicer Non-Interactively:

Several options exist to support running AD Splicer non-interactively. Added to the command line as needed, they are:

- **-defaultsfile** - Specifies the location of the defaults file <$APPL_TOP/admin/SID/defaultsfile>

- **-interactive** - Indicates whether AD Splice should run interactively or not. Valid values are y or n.

- **-restart** - Tells AD Splice to restart from where it left off. Valid values are y or n.

- **-abandon** - Used to abandon a previous failed session or continue with the session. Valid values are y or n (abandon or continue, respectively).

- **-stdin** - If this parameter is specified, AD Splice prompts for passwords from the standard input. The default is for passwords to be supplied without prompting.

File Character Set Converter

The File Character Set Converter converts individual files, one at a time, from one character set to another. You may need to perform this task to convert text files you receive from Oracle to the character set used by your system. For example, you might need to convert SQL*Plus scripts, PL/SQL scripts, loader files, driver files, ODF files, header files, or HTML files.

> **Tip:** In general, you do not need to run File Character Set Converter manually. AD Administration, AutoPatch, and Rapid Install will normally perform any required character set conversion automatically.

Required Parameters:

The following parameters are used when running the converter.

File Character Set Converter Parameters

Parameter	Definition
source_file	Path and file name for the (source) file to be converted.
source_char_set	Character set for the file to be converted (source).
destination_file	Path and file name for the (destination) file after it is converted.
dest_char_set	Character set for the converted (destination) file.

The File Character Set Converter Interface:

The File Character Set Converter is a command line utility. It does not use menus or input screens.

Running the File Character Set Converter:

To run this utility, perform these steps:

1. Set the environment.

   You must set the environment in order to apply the environment variables that

define your system. This task is common to many AD utilities. See Setting the Environment, *Oracle E-Business Suite Maintenance Guide* for the basic steps.

2. Start the utility.

   Start the File Character Set Converter with this command:

   ```
   $ adncnv <source_file> <source_char_set> <destination_file>
   <dest_char_set>
   ```

   The path and file name for the source and the destination files can be the same if the source file's directory and the APPLTMP directory are on the same file system. In general, it is simpler and safer to use different source and destination file names.

   > **Tip:** If you cannot convert to the same file name, convert to a different file name, or change APPLTMP to a directory on the same file system as the source file directory.

   For example, to convert the file afdict.ldt from the WE8ISO8859P1 character set to the UTF8 character set, you would enter commands as shown in this example:

   UNIX

   ```
   $ cd $FND_TOP/patch/115/import/<language>
   $ cp afdict.ldt afdict.old
   $ adncnv afdict.old we8iso8859p1 afdict.ldt utf8
   ```

12

Developer Tools

Developer Tools

Oracle E-Business Suite provides several tools to help developers create and debug custom forms and programs. Described in this chapter are the Forms Personalization, Work Directory, and Portlet Generator features.

In addition, the CUSTOM library allows extension of Oracle E-Business Suite forms without modification of Oracle E-Business Suite code. For information on the CUSTOM library, see the *Oracle E-Business Suite Developer's Guide*.

Form Personalization

The Form Personalization feature allows you to declaratively alter the behavior of Forms-based screens, including changing properties, executing builtins, displaying messages, and adding menu entries.

For each function (a form running in a particular context based on parameters passed to it), you can specify one or more Rules. Each Rule consists of an Event, an optional Condition, the Scope for which it applies, and one or more Actions to perform.

An Event is a trigger point within a form, such as startup (WHEN-NEW-FORM-INSTANCE), or when focus moves to a new record (WHEN-NEW-RECORD-INSTANCE). There are standard events that almost every form sends, and certain forms send additional product-specific events.

The Scope is evaluated based on the current runtime context to determine if a Rule should be processed or not. The Scope can be at the Site, Responsibility, User, or Industry level. Each Rule can have one or more Scopes associated with it.

The Condition is an optional SQL code fragment that is evaluated when the Event occurs; if it evaluates to TRUE then the Actions are processed.

Each Action consists of one of the following:

- setting a Property, such as making a field Required or hiding a Tab page
- executing a Builtin, such as GO_BLOCK, DO_KEY or EXECUTE_FUNCTION
- displaying a Message
- enabling a Special menu entry

Once Rules are defined, when the target function is run then the Rules are automatically applied as events occur within that form.

Although the Form Personalization feature is declarative, the intended audience is a person familiar with Oracle Forms including the PL/SQL programming language, and the *Oracle E-Business Suite Developer's Guide*. Additionally, any change made could interfere with the base code of a form (the code that Oracle ships).

Please refer to "Form Personalizations in Oracle E-Business Suite", Knowledge Document 395117.1, on My Oracle Support for more information.

Work Directory

The Work Directory feature enables a developer, support consultant, or other technical specialist to test modifications to forms and concurrent programs in Oracle E-Business Suite without affecting users of the same code tree.

Using the Work Directory, a user can be logged into an Oracle E-Business Suite system but access a version of a form or concurrent program that is not within the standard $PROD_TOP directory. For example, an on-site developer can test out a new version of a custom form without affecting other testing on the system.

You can use the Work Directory feature for alternate files of forms and concurrent programs only.

Implementation

To implement this feature, set up a directory to hold the alternate files for your forms or concurrent programs.

To use an alternate file, set the profile option FND:Override Directory with the path for the directory containing the alternate file.

> **Important:** This profile option should usually be set at the User level only. If you set FND:Override Directory at the Site level, for example, you will affect all users at that site using the particular forms.

Using the Work Directory

After you have created the alternate directory and set the profile option FND: Override

Directory with the appropriate value, you can use files in that alternate directory.

In searching for the appropriate file path for a form or concurrent program, Oracle E-Business Suite will first check to see if the profile option FND: Override Directory is set and if a given file exists in the specified directory. If the above two conditions are true then the alternate file is used. If the profile option is not set or if the necessary file does not exist in that directory, then the default (usual) file path is used.

> **Note:** The Oracle E-Business Suite Navigator caches the paths to files that have been successfully opened. If the standard form has been opened, then that form will be used for the remainder of the session. To switch to a different file path, you must exit and restart Oracle E-Business Suite.

To provide a visual indication that an alternate form is in use, the developer of the form should specify a different version number for the form in the PRE-FORM trigger. This version number appears during runtime using Help > About Oracle Applications. For more information, see the *Oracle E-Business Suite Developer's Guide.*

Web Enabled PL/SQL Window

> **Note:** This window and the information below are provided for backward compatibility only.

Use this form to maintain the FND_ENABLED_PLSQL runtime registry.

PL/SQL Object Block

Enter the following fields.

Name

The name of the PL/SQL object.

Type

The type of the PL/SQL object. The object may be a package, package procedure, or procedure.

Enabled

Check this box to enable the PL/SQL object., and the Logging Service

Portlet Generator

The Portlet Generator tool lets you convert existing standalone Oracle Application Framework regions into portlets. After you convert, or *portletize* regions, these portlets are available to be integrated into any compatible portal product, such as Oracle WebCenter, Oracle Portal, or any WSRP-compliant portal.

To be available for portletization, a region must have the following properties.

- Any region to portletize must have an Application Module (AM) defined and must have its standalone property set to true.
- An inline region must have an AM defined and have its standalone property set to true.
- A content region must have an AM defined.

> **Note:** Content regions do not have a standalone property, so there is no requirement for such a property setting for content regions.

The following region types can be portletized.

- advancedTable
- bulletedList
- cellFormat
- contentContainer
- defaultDoubleColumn
- defaultFormStack
- defaultSingleColumn
- defaultStack
- flowLayout
- footer
- gantt
- graphTable
- hgrid

- header
- hideShow
- hideShowHeader
- labeledFieldLayout
- messageComponentLayout
- query
- rowLayout
- shuttle
- stackLayout
- subTabLayout
- table
- tableLayout
- tree

Prerequisites

The Portlet Generator uses the FWK_PORTLET_GEN table to determine which regions are available for portletization. This table must be refreshed periodically to include any new regions. Oracle E-Business Suite provides the Refresh Portlet Generator Table concurrent program (FWKPRTTB) to refresh the table. Run the Refresh Portlet Generator Table program regularly to ensure that all eligible regions appear when you search for regions to portletize. This program does not require any parameters.

> **Important:** You must run the Refresh Portlet Generator Table concurrent program prior to using the Portlet Generator tool for the first time, otherwise no rows are returned when you search for regions to portletize.

Using the Portlet Generator

You can access the Portlet Generator by two different methods.

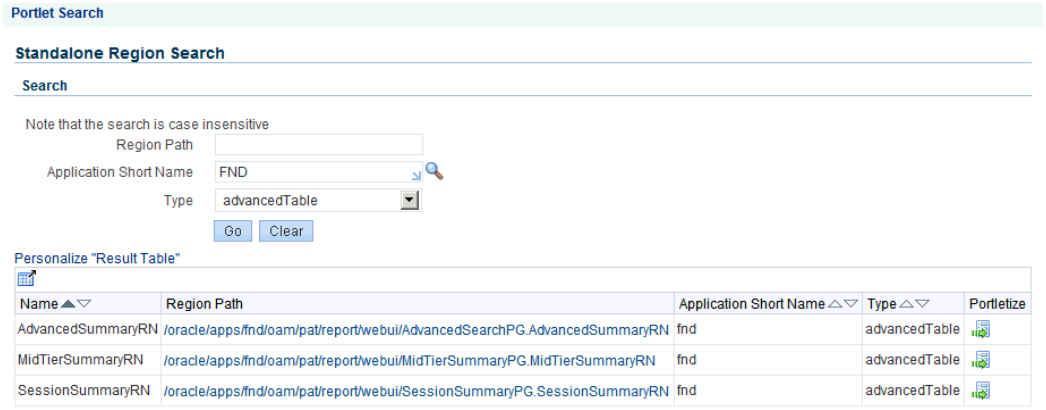
- From the Oracle E-Business Suite Navigator, choose the Functional Administrator responsibility and select Home. Then select the Portletization tab. In this case the

Portlet Generator initially displays the Portlet Search page to let you search for the region you want to portletize.

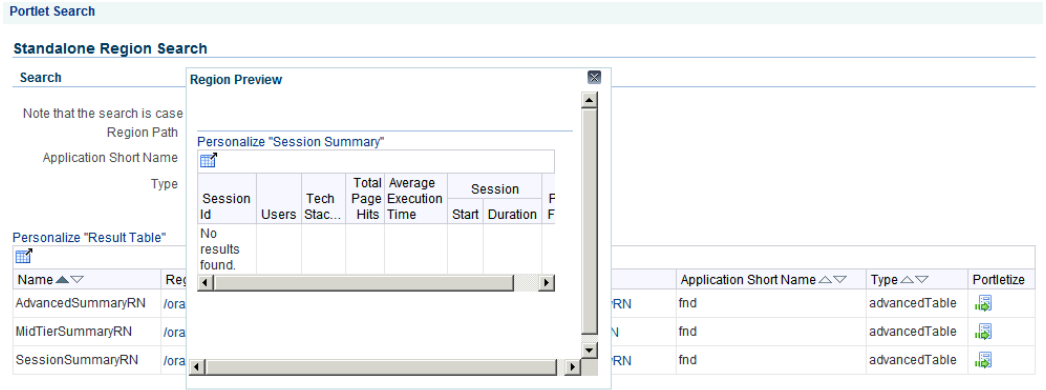
- Select a Portletize link within the page where the region appears. Portletize links appear in the same area where Personalize links appear within a page.

  Note: Portletization and personalization are mutually exclusive. You can either portletize or personalize a region, but not both.

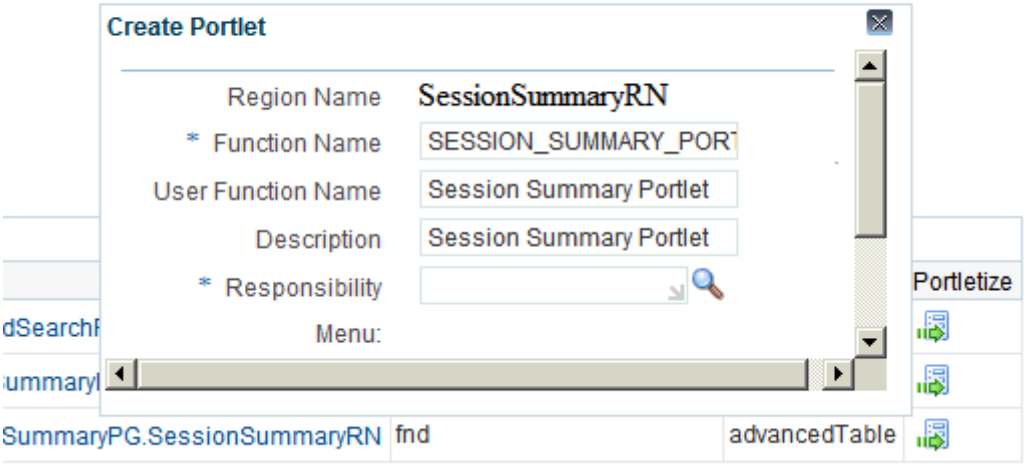
In the Portlet Search page, search for the region you want using the full region path, the application short name, or the region type as search criteria. The search criteria fields are case-insensitive.



In the search results list, you can click a region link in the Region Path column to display a pop-up preview of the region.



To create a portlet from a region, choose the portletization icon for that region in the search results list, or select the Portletize link for the region in the page where the region appears.



The Create Portlet pop-up window automatically generates an internal function name, a user function name for display in user interfaces, and a description for the portlet based on the region name. You can optionally enter different values for the function name, user function name, and description if you want. The Portlet Generator also validates whether the automatically generated function name already exists for another portlet. If so, you must enter a unique function name before you can save the portlet definition.

You must select the responsibility through which users can access the portlet. After you select the responsibility, the Create Portlet window displays the top-level menu associated with that responsibility. Users must have access to this responsibility and menu to use the portlet.

Integration with Application Development Framework

You can integrate an Application Development Framework (ADF) application from the Oracle E-Business Suite home page, regardless of how or if the ADF application integrates with Oracle E-Business Suite data on the back end.

For more information on using ADF applications with Oracle E-Business Suite, see My Oracle Support Knowledge Document 563047.1, *Using Oracle Application Framework (OAF) and Application Development Framework (ADF)*.

The ADF application must be located on a separate application tier server machine from the Oracle E-Business Suite application server.

The ADF application must be deployed using Oracle WebLogicServer 11*g* or another appropriate application server.

1. Set the value for the profile External ADF Application URL (internal name FND_EXTERNAL_ADF_URL) to the context root of the URL for the application,

such as 'https://<host>:<port>/<context-root>'.

For example, set it to:

```
http://www.myhost.com:7001/my-context-root
```

The profile External ADF Application URL can be viewed and updated on all four access levels by a system administrator. Users cannot see or update this profile.

2. Create a new Oracle E-Business Suite function using the 'ADFX' function type. Set the Web-HTML call as appropriate.

   - Function Type: 'External ADF Application' (ADFX)
   - Web-HTML Call: 'GWY.jsp?targetPage=<an ADF page, such as /faces/ModelerHome.jsp>'

     For example, set the HTML call to:

     ```
     GWY.jsp?targetPage=faces/MyADFShoppingAppDashboard
     ```

3. Incorporate the new function into Oracle E-Business Suite security by adding it to a menu and creating appropriate roles/responsibilities and grants. For example, add the function to a menu, add the menu to a responsibility, and assign the responsibility to a user.

4. Log in as that user and access a responsibility from the step above. The ADF application link should appear. Select the link to go to the ADF application.

Related Topics

Oracle E-Business Suite Security Guide

13

Using Loaders

Generic Loader

The Generic Loader (FNDLOAD) is a concurrent program that can move Oracle E-Business Suite data between database and text file representations. The loader reads a configuration file to determine what data to access. For information on specific configuration files consult the *Open Interfaces Guide* for your product group. The following sections describe the operation of the Generic Loader.

> **Warning:** Use only the loader files provided by Oracle E-Business Suite. If you use files not provided by Oracle E-Business Suite or modify the provided scripts you risk corrupting your database. Oracle does not support the use of custom loader files or modified Oracle E-Business Suite loader files.

Overview

The Generic Loader can download data from an application entity into a portable, editable text file. This file can then be uploaded into any other database to copy the data. Conversion between database store and file format is specified by a configuration file that is read by the loader.

The Generic Loader downloads data from a database according to a configuration (.lct) file, and converts the data into a data file (.ldt file). The Generic Loader can then upload this data to another database using a configuration file.

The loader operates in one of two modes: download or upload. In the download mode, data is downloaded from the database to a text file; in the upload mode, data is uploaded from a text file to the database.

Data structures supported by the loader include master-detail relationships and foreign key reference relationships.

In both downloading and uploading, the structure of the data involved is described by

a configuration file. The configuration file describes the structure of the data and also the access methods to use to copy the data into or out of the database. The same configuration file may be used for both uploading and downloading.

When downloading, the Generic Loader creates a second file, called the data file, that contains the structured data selected for downloading. The data file has a standard syntax for representing the data that has been downloaded. When uploading, the Generic Loader reads a data file to get the data that it is to upload. In most cases, the data file was produced by a previous download, but may have come from another source. The data file cannot be interpreted without the corresponding configuration file available.

Download database information to a text file

The text file is human-readable and portable, and can be examined and modified with any editor. Generally, a "developer key" is used to identify records written out to text files. For example, the PROFILE_OPTION_NAME, not the PROFILE_OPTION_ID, is used to identify records in the Profiles configuration file.

Upload (merge) the information in a text file to the database

In uploading, if a row exists, but has different attributes, the row is updated. If a row does not exist, a new row is inserted. Deletion of a row is not supported

These download and upload capabilities allow profile value information that is defined in one database to be easily propagated to other databases. This is useful for delivering Oracle E-Business Suite seed data to customers, as well as for copying customer profile definitions from a primary site to other sites.

The text file version of profile value data is also useful for bulk editing operations, which can be accomplished more efficiently with a text editor than with a form.

Preservation of data

FNDLOAD uses the OWNER and LAST_UPDATE_DATE attributes to determine whether to overwrite pre-existing data. The rules applied are:

1. If the entity being uploaded is not present in the database, a new entity is always inserted.

2. Entities uploaded from a file with OWNER=SEED never overwrite entities with OWNER=CUSTOM in the database.

3. Entities with OWNER=CUSTOM uploaded from a file always update entities with OWNER=SEED in the database.

4. If the owner of the entity is the same in the file and database, the entity is updated only if the LAST_UPDATE_DATE in the file is later than the LAST_UPDATE_DATE in the database.

FNDLOAD and Online Patching

Each seed data table is modified to support editioned storage of seed data. A new edition name column is added so that Run Edition data and Patch Edition data can be stored in the same table. A maintenance trigger is added to populate the column, as well as a VPD filter policy to select the correct data set per edition. During online patching, seed data changes from the Run Edition will be copied to the Patch Edition using a forward cross edition trigger. After patch cutover, obsolete data editions and code will be deleted.

Online patching of seed data tables requires that all seed data tables to have already been upgraded to support editioned storage. The Seed Data Manager will provide all the centralized logic required to implement online seed data patching. The Oracle E-Business Suite Release 12.2 upgrade will implement editioned storage on all the known seed data tables using the Seed Data Manager "UPGRADE" API. In addition, during online patching, all the seed data loaders will call Seed Data Manager "PREPARE" API with the affected table names, before loading any data.

FNDLOAD Integration

Each Seed Data Loader calls the Seed Data Manager "Prepare" API to signal that an upload is about to take place and specify which tables need to have patch edition data copies. MOst loaders are implemented as loader configuration files (LCT files) for the Generic Loader. FNDLOAD is extended to support a new PREPARE statement with the following syntax:

```
PREPARED <ENTITY>
   TABLE <TABLE_NAME>
   TABLE <TABLE_NAME>
   ...
```

Other types of loaders must add a direct call to the Prepare API before loading any data. For example,

```
"begin
   FND_ZD_SEED.PREPARE('OKL_CONTEXT_GROUPS_B');
   FND_ZD_SEED.PREPARE('OKL_CONTEXT_GROUPS_TL');
   OKL_CGR_PVT.LOAD_SEED_ROW(
           p_upload_mode      => :UPLOAD_MODE,
           p_name             => :NAME,
           p_description      => :DESCRIPTION,
           p_owner            => :OWNER,
           p_last_update_date => :LAST_UPDATE_DATE);
end; "
```

FNDLOAD Executable

The Generic Loader is a concurrent program named FNDLOAD. The concurrent executable takes the following parameters:

```
FNDLOAD <APPS username>/password 0 Y mode configfile datafile entity [
param ... ]
```

where

<APPS username>/*password*	The APPS schema and password in the form username/password[@connect_string]. If connect_string is omitted, it is taken in a platform-specific manner from the environment using the name TWO_TASK.
< 0 Y >	Concurrent program flags
mode	UPLOAD or DOWNLOAD. UPLOAD causes the datafile to be uploaded to the database. DOWNLOAD causes the loader to fetch rows and write them to the datafile.
<configfile>	The configuration file to use (usually with a suffix of .lct, but not enforced or supplied by the loader).
<datafile>	The data file to write (usually with a suffix of .ldt, but not enforced or supplied by the loader). If the data file already exists, it will be overwritten.
<entity>	The entity(ies) to upload or download. When uploading, you should always upload all entities, so specify a "-" to upload all entities.
< [param] >	Zero or more additional parameters are used to provide bind values in the access SQL (both UPLOAD and DOWNLOAD). Each parameter is in the form NAME=VALUE. NAME should not conflict with an attribute name for the entities being loaded.

File Specifications

The configuration file and data file parameters are specified in one of two ways:

```
@<application_short_name>:[<dir>/.../]file.ext
```

For example,

```
@fnd/120/loader/fndapp.lct
@po:install/data/poreq.ldt
```

Alternatively, the parameters can be specified as such:

```
<native path>
```

For example,

```
mydata.ldt
c:\loader\config\cfg102.lct
```

Examples

An example of downloading is:

```
FNDLOAD apps/password@devdb 0 Y
     DOWNLOAD testcfg.lct out.ldt FND_APPLICATION_TL APPSNAME=FND
```

This command does the following:

- connects to apps/*password*@devdb
- downloads data using the configuration file testcfg.lct
- writes data to data file out.ldt
- downloads the FND_APPLICATION_TL entity with APPSNAME parameter defined as value 'FND'

An example of uploading is:

```
FNDLOAD apps/password@custdb 0 Y
     UPLOAD fndapp.lct fnd1234.ldt -
```

This command does the following:

- connects to apps/*password*@custdb
- uploads data using the configuration file in fndapp.lct from data file in fnd1234.ldt
- The contents of the entire data file is uploaded.

Configuration File

Operation of the Generic Loader is controlled by the specified configuration file. The configuration file contains the following:

- DEFINE block
- DOWNLOAD block
- UPLOAD block

The contents of the configuration file specify the structure of the data and the access methods to use to move the data between the data file and a database.

DEFINE Block

The DEFINE block specifies the structure of the datafile records. The define block format is identical to that already generated by existing Oracle Application Object Library loaders. The structure of this section is

```
DEFINE <entity> KEY <key_attribute_name> <datatype> ...
 (BASE|TRANS|CTX) <attribute_name> <datatype> ...
 [DEFINE <child_entity> ...]
END <entity>
```

Example

```
DEFINE MENU
  KEY MENU_ID            NUMBER
  TRANS USER_MENU_NAME   VARCHAR2(80)
  TRANS DESCRIPTION      CLOB
  DEFINE ENTRY
    BASE  SUBMENU                REFERENCES MENU
    TRANS USER_SUB_MENU_NAME     VARCHAR2(60)
    BASE  DESCRIPTION_DOCUMENT   CLOB
  END ENTRY
END MENU
```

One or more KEY attributes defines the primary key of each entity. BASE and CTX attributes are those that do not require translation. TRANS attributes do. Note that BASE and CTX attributes are treated identically. That is, CTX is just a synonym for BASE. The CTX attribute type is provided merely to allow users to optionally differentiate between BASE attributes. For example, translators may wish to simplify their .ldt files by stripping out the BASE attributes. However, they may also want to keep some BASE attributes for context. By denoting some attributes as BASE and some as CTX, they can control which attributes to remove.

Data types can be standard Oracle scalar types, except that only VARCHAR2(size), NUMBER, and CLOB are currently supported. An attribute can also be defined as a foreign key reference to another entity in your configuration file. The foreign key entity must be a "top-level" entity and its download statement must include filter parameters in its WHERE clause for each of its key attributes. Also, the parameter names must match the key attribute names exactly.

Note that entity definitions can be nested to indicate master-detail relationships. Nested entity definitions inherit the key attributes of their parent entities and should not redefine them.

DOWNLOAD Statement

The DOWNLOAD statement is a SQL statement that selects rows to download. The statement can join to other tables to resolve sequence generated ID numbers into developer keys where possible. The DOWNLOAD statement may also contain bind values of the form ':NAME' which are substituted with parameter values from the command line. DOWNLOAD statements have the form

```
DOWNLOAD <entity> "select <attribute expressions> from ..."
```

Example

```
DOWNLOAD FND_LOOKUP_TYPE
  "select VA.APPLICATION_SHORT_NAME VIEW_APPSNAME,
   LT.LOOKUP_TYPE,
   OA.APPLICATION_SHORT_NAME,
   LT.CUSTOMIZATION_LEVEL,
   decode(LT.LAST_UPDATED_BY, 1, 'SEED', 'CUSTOM') OWNER,
   LT.MEANING,
   LT.DESCRIPTION
   from   FND_LOOKUP_TYPES_VL LT,
   FND_APPLICATION VA,
   FND_APPLICATION OA,
   FND_SECURITY_GROUPS SG
   where  VA.APPLICATION_ID = LT.VIEW_APPLICATION_ID
   and    OA.APPLICATION_ID = LT.APPLICATION_ID
   and    (:VIEW_APPSNAME is null or
     (:VIEW_APPSNAME is not null
   and VA.APPLICATION_SHORT_NAME like :VIEW_APPSNAME))
   and    (:LOOKUP_TYPE is null or
      (LOOKUP_TYPE is not null and LT.LOOKUP_TYPE like :LOOKUP_TYPE))
   and    SG.SECURITY_GROUP_ID = LT.SECURITY_GROUP_ID
   and    ((:SECURITY_GROUP is null and SG.SECURITY_GROUP_KEY =
'STANDARD') or
     (:SECURITY_GROUP is not null
   and SG.SECURITY_GROUP_KEY = :SECURITY_GROUP))
   order by 1, 2 "
```

Download statements for child entities may reference any key attribute of the parent entity, or any command line parameter.

UPLOAD Statement

The UPLOAD statement is a SQL statement or PL/SQL anonymous block which accepts file data and applies it to the database. The statement is executed once for each record read from the data file. Bind values in the statement are satisfied by attributes from the file data or command line parameters.

Example

```
UPLOAD FND_LOOKUP_TYPE
 BEGIN
 " begin
   if (:UPLOAD_MODE = 'NLS') then
     fnd_lookup_types_pkg.TRANSLATE_ROW(
  x_lookup_type  => :LOOKUP_TYPE,
  x_security_group      => :SECURITY_GROUP,
  x_view_application    => :VIEW_APPSNAME,
  x_owner        => :OWNER,
  x_meaning      => :MEANING,
  x_description  => :DESCRIPTION);
   else
  fnd_lookup_types_pkg.LOAD_ROW(
  x_lookup_type  => :LOOKUP_TYPE,
  x_security_group      => :SECURITY_GROUP,
  x_view_application    => :VIEW_APPSNAME,
  x_owner        => :OWNER,
  x_meaning      => :MEANING,
  x_description  => :DESCRIPTION);
   end if;
   end; "
```

As in the DOWNLOAD, the UPLOAD statement for child entities may reference any attributes from the parent record.

> **Note:** The BEGIN line is required and only expected for a parent entity UPLOAD section. Otherwise, it is not expected and an error will be generated if it is used to start a child entity UPLOAD section (where the child is not a parent itself) or a single entity UPLOAD section.

Example

```
UPLOAD FND_LOOKUP_VALUE
  " begin
 if (:UPLOAD_MODE = 'NLS') then
   fnd_lookup_values_pkg.TRANSLATE_ROW(
     x_lookup_type  => :LOOKUP_TYPE,
     x_lookup_code  => :LOOKUP_CODE,
     x_security_group      => :SECURITY_GROUP,
     x_view_application    => :VIEW_APPSNAME,
     x_owner        => :OWNER,
     x_meaning      => :MEANING,
     x_description  => :DESCRIPTION);
 else
   fnd_lookup_values_pkg.LOAD_ROW(
     x_lookup_type  => :LOOKUP_TYPE,
     x_lookup_code  => :LOOKUP_CODE,
     x_security_group      => :SECURITY_GROUP,
     x_view_application    => :VIEW_APPSNAME,
     x_owner        => :OWNER,
     x_meaning      => :MEANING,
     x_description  => :DESCRIPTION,
     x_tag   => :TAG);
 end if;
  end;"
```

Data File

A data file is a portable text file. The data file created from a download using the above configuration file would include:

```
# -- Begin Entity Definitions --

DEFINE FND_LOOKUP_TYPE
   KEY   VIEW_APPSNAME VARCHAR2(50)
   KEY   LOOKUP_TYPE VARCHAR2(30)
   BASE  OWNER VARCHAR2(6)
   TRANS MEANING VARCHAR2(80)
   TRANS DESCRIPTION VARCHAR2(240)
   DEFINE FND_LOOKUP_VALUE
     KEY   LOOKUP_CODE VARCHAR2(30)
     BASE  END_DATE_ACTIVE VARCHAR2(10)
     BASE  OWNER VARCHAR2(6)
     TRANS MEANING VARCHAR2(80)
     TRANS DESCRIPTION VARCHAR2(240)
     BASE  TAG VARCHAR2(30)
   END FND_LOOKUP_VALUE
 END FND_LOOKUP_TYPE

# -- End Entity Definitions --

BEGIN FND_LOOKUP_TYPE "FND" "YES_NO"
   OWNER = "SEED"
   MEANING = "Yes or No"

   BEGIN FND_LOOKUP_VALUE Y
     OWNER = "SEED"
     MEANING = "Yes"
   END FND_LOOKUP_VALUE

  BEGIN FND_LOOKUP_VALUE N
     OWNER = "SEED"
     MEANING = "No"
   END FND_LOOKUP_VALUE
END FND_LOOKUP_TYPE
```

Oracle Application Object Library Configuration Files

Oracle Application Object Library provides several configuration files for the Generic Loader that you can use with your setup data.

These configuration files operate on the following data:

- Concurrent program definitions
- Request groups
- Lookup types and lookup values
- Profile options and profile option values

- Flexfields setup data
- Attachments definitions
- Messages
- Security information

Attachments Setup Data Configuration File

Use the file afattach.lct for loading attachments setup data.

The following table lists the entities, sub-entities (if any), and download parameters for this configuration file.

Entity	Sub-entities, if any	Download Parameters
FND_ATTACHMENT_ FUNCTIONS	FND_ATTACHMENT_BLOCKS FND_ATTACHMENT_BLOCK_ ENTITIES FND_DOC_CATEGORY_USAGES	APPLICATION_ SHORT_NAME FUNCTION_NAME FUNCTION_TYPE

The entity definitions are:

```
DEFINE FND_ATTACHMENT_FUNCTIONS
  KEY   FUNCTION_NAME    VARCHAR2(30)
  KEY   FUNCTION_TYPE    VARCHAR2(1)
  KEY   APP_SHORT_NAME    VARCHAR2(50)
  CTX   OWNER       VARCHAR2(7)
  BASE  SESSION_CONTEXT_FIELD  VARCHAR2(61)
  BASE  ENABLED_FLAG      VARCHAR2(1)

  DEFINE FND_ATTACHMENT_BLOCKS
    KEY   BLOCK_NAME    VARCHAR2(30)
    CTX   OWNER     VARCHAR2(7)
    BASE  QUERY_FLAG    VARCHAR2(1)
    BASE  SECURITY_TYPE   VARCHAR2(50)
    BASE  ORG_CONTEXT_FIELD VARCHAR2(61)
    BASE  SET_OF_BOOKS_CONTEXT_FIELD  VARCHAR2(61)
    BASE  BUSINESS_UNIT_CONTEXT_FIELD  VARCHAR2(61)
    BASE  CONTEXT1_FIELD   VARCHAR2(61)
    BASE  CONTEXT2_FIELD   VARCHAR2(61)
    BASE  CONTEXT3_FIELD   VARCHAR2(61)

    DEFINE FND_ATTACHMENT_BLK_ENTITIES
  KEY   BLK_ENTITY  REFERENCES
FND_DOCUMENT_ENTITIES
  BASE  DISPLAY_METHOD     VARCHAR2(1)
  BASE  INCLUDE_IN_INDICATOR_FLAG  VARCHAR2(1)
  CTX   OWNER      VARCHAR2(7)
  BASE  PK1_FIELD     VARCHAR2(61)
  BASE  PK2_FIELD     VARCHAR2(61)
  BASE  PK3_FIELD     VARCHAR2(61)
  BASE  PK4_FIELD     VARCHAR2(61)
  BASE  PK5_FIELD     VARCHAR2(61)
  BASE  SQL_STATEMENT   VARCHAR2(2000)
  BASE  INDICATOR_IN_VIEW_FLAG  VARCHAR2(1)
  BASE  QUERY_PERMISSION_TYPE   VARCHAR2(1)
  BASE  INSERT_PERMISSION_TYPE  VARCHAR2(1)
  BASE  UPDATE_PERMISSION_TYPE  VARCHAR2(1)
  BASE  DELETE_PERMISSION_TYPE  VARCHAR2(1)
  BASE  CONDITION_FIELD    VARCHAR2(61)
        BASE  CONDITION_OPERATOR VARCHAR2(50)
  BASE  CONDITION_VALUE1   VARCHAR2(100)
  BASE  CONDITION_VALUE2   VARCHAR2(100)
    END FND_ATTACHMENT_BLK_ENTITIES
  END FND_ATTACHMENT_BLOCKS

  DEFINE FND_DOC_CATEGORY_USAGES
    KEY CATEGORY_USAGE REFERENCES
    FND_DOCUMENT_CATEGORIES
    BASE ENABLED_FLAG VARCHAR2(1)
    CTX  OWNER   VARCHAR2(7)
 END FND_DOC_CATEGORY_USAGES
END FND_ATTACHMENT_FUNCTIONS

DEFINE FND_DOCUMENT_ENTITIES
  KEY   DATA_OBJECT_CODE  VARCHAR2(30)
  BASE  APP_SHORT_NAME   VARCHAR2(50)
  BASE  TABLE_NAME     VARCHAR2(30)
  BASE  ENTITY_NAME   VARCHAR2(40)
  CTX   OWNER     VARCHAR2(7)
  BASE  PK1_COLUMN   VARCHAR2(30)
  BASE  PK2_COLUMN   VARCHAR2(30)
  BASE  PK3_COLUMN   VARCHAR2(30)
```

```
BASE  PK4_COLUMN   VARCHAR2(30)
  BASE  PK5_COLUMN   VARCHAR2(30)
  TRANS USER_ENTITY_NAME  VARCHAR2(240)
  TRANS USER_ENTITY_PROMPT VARCHAR2(40)
END FND_DOCUMENT_ENTITIES

DEFINE FND_DOCUMENT_CATEGORIES
  KEY   CATEGORY_NAME   VARCHAR2(30)
  BASE  APP_SHORT_NAME   VARCHAR2(50)
  CTX   OWNER      VARCHAR2(7)
  BASE  START_DATE_ACTIVE  VARCHAR2(11)
  BASE  END_DATE_ACTIVE   VARCHAR2(11)
  BASE  ATTRIBUTE_CATEGORY VARCHAR2(30)
  BASE  ATTRIBUTE1   VARCHAR2(150)
  BASE  ATTRIBUTE2   VARCHAR2(150)
  BASE  ATTRIBUTE3   VARCHAR2(150)
  BASE  ATTRIBUTE4   VARCHAR2(150)
  BASE  ATTRIBUTE5   VARCHAR2(150)
  BASE  ATTRIBUTE6   VARCHAR2(150)
  BASE  ATTRIBUTE7   VARCHAR2(150)
  BASE  ATTRIBUTE8   VARCHAR2(150)
  BASE  ATTRIBUTE9   VARCHAR2(150)
  BASE  ATTRIBUTE10   VARCHAR2(150)
  BASE  ATTRIBUTE11   VARCHAR2(150)
  BASE  ATTRIBUTE12   VARCHAR2(150)
  BASE  ATTRIBUTE13   VARCHAR2(150)
  BASE  ATTRIBUTE14   VARCHAR2(150)
  BASE  ATTRIBUTE15   VARCHAR2(150)
  BASE  DEFAULT_DATATYPE_ID VARCHAR2(50)
  BASE  APP_SOURCE_VERSION VARCHAR2(255)
  TRANS USER_NAME     VARCHAR2(255)
END FND_DOCUMENT_CATEGORIES

DEFINE FND_DOCUMENT_DATATYPES
  KEY   DATATYPE_ID   VARCHAR2(50)
  KEY   NAME     VARCHAR2(30)
  CTX   OWNER     VARCHAR2(7)
  BASE  START_DATE_ACTIVE VARCHAR2(11)
  BASE  END_DATE_ACTIVE  VARCHAR2(11)
  TRANS USER_NAME    VARCHAR2(30)
END FND_DOCUMENT_DATATYPES
```

Concurrent Program Configuration File

The concurrent program configuration file afcpprog.lct downloads and uploads concurrent program definitions. It takes as parameters program name and application name.

The following table lists the entities, sub-entities (if any), and download parameters for this configuration file.

Entity	Sub-entities, if any	Download Parameters
PROGRAM	INCOMPATIBILITY	CONCURRENT_PROGRAM_NAME APPLICATION_SHORT_NAME
EXECUTABLE		EXECUTABLE_NAME

The entity definition is:

```
DEFINE PROGRAM
  KEY   CONCURRENT_PROGRAM_NAME   VARCHAR2(30)
  KEY   APPLICATION       VARCHAR2(50)
  CTX   OWNER          VARCHAR2(7)
  TRANS USER_CONCURRENT_PROGRAM_NAME  VARCHAR2(240)
  BASE  EXEC_APPLICATION     VARCHAR2(50)
  BASE  EXECUTABLE_NAME      VARCHAR2(30)
  BASE  EXECUTION_METHOD_CODE   VARCHAR2(1)
  BASE  ARGUMENT_METHOD_CODE    VARCHAR2(1)
  BASE  QUEUE_CONTROL_FLAG            VARCHAR2(1)
  BASE  QUEUE_METHOD_CODE             VARCHAR2(1)
  BASE  REQUEST_SET_FLAG              VARCHAR2(1)
  BASE  ENABLED_FLAG                  VARCHAR2(1)
  BASE  PRINT_FLAG                    VARCHAR2(1)
  BASE  RUN_ALONE_FLAG                VARCHAR2(1)
  BASE  SRS_FLAG                      VARCHAR2(1)
  TRANS DESCRIPTION                   VARCHAR2(240)
  BASE  CLASS_APPLICATION         VARCHAR2(50)
  BASE  CONCURRENT_CLASS_NAME         VARCHAR2(30)
  BASE  EXECUTION_OPTIONS             VARCHAR2(250)
  BASE  SAVE_OUTPUT_FLAG              VARCHAR2(1)
  BASE  REQUIRED_STYLE                VARCHAR2(1)
  BASE  OUTPUT_PRINT_STYLE            VARCHAR2(30)
  BASE  PRINTER_NAME                  VARCHAR2(30)
  BASE  MINIMUM_WIDTH                 VARCHAR2(50)
  BASE  MINIMUM_LENGTH                VARCHAR2(50)
  BASE  REQUEST_PRIORITY              VARCHAR2(50)
  BASE  ATTRIBUTE_CATEGORY      VARCHAR2(30)
  BASE  ATTRIBUTE1                VARCHAR2(150)
  BASE  ATTRIBUTE2                VARCHAR2(150)
  BASE  ATTRIBUTE3                VARCHAR2(150)
  BASE  ATTRIBUTE4                VARCHAR2(150)
  BASE  ATTRIBUTE5                VARCHAR2(150)
  BASE  ATTRIBUTE6                VARCHAR2(150)
  BASE  ATTRIBUTE7                VARCHAR2(150)
  BASE  ATTRIBUTE8               VARCHAR2(150)
  BASE  ATTRIBUTE9             VARCHAR2(150)
  BASE  ATTRIBUTE10              VARCHAR2(150)
  BASE  ATTRIBUTE11               VARCHAR2(150)
  BASE  ATTRIBUTE12               VARCHAR2(150)
  BASE  ATTRIBUTE13              VARCHAR2(150)
  BASE  ATTRIBUTE14               VARCHAR2(150)
  BASE  ATTRIBUTE15               VARCHAR2(150)
  BASE  OUTPUT_FILE_TYPE          VARCHAR2(4)
  BASE  RESTART                   VARCHAR2(1)
  BASE  NLS_COMPLIANT             VARCHAR2(1)
  BASE  CD_PARAMETER           VARCHAR2(240)
  BASE  INCREMENT_PROC            VARCHAR2(61)
  BASE  MLS_EXECUTABLE_APPLICATION  VARCHAR2(50)
  BASE  MLS_EXECUTABLE_NAME          VARCHAR2(50)
  BASE  ENABLE_TIME_STATISTICS        VARCHAR2(1)
  BASE  SECURITY_GROUP_NAME           NUMBER
  BASE  RESOURCE_CONSUMER_GROUP       VARCHAR2(30)
  BASE  ROLLBACK_SEGMENT              VARCHAR2(30)
  BASE  OPTIMIZER_MODE              VARCHAR2(30)

END PROGRAM
```

Flexfields Setup Data Configuration File

Use the file afffload.lct for loading flexfields data.

> **Warning:** Do not modify the data files you download using the flexfields configuration file. You risk corrupting your flexfields data. Oracle E-Business Suite does not support any changes you make to the data files.

The configuration file includes the following entities:

- Value sets
- Descriptive flexfields
- Key flexfields

Flexfield Value Sets

The entity VALUE_SET includes the following table details of table validated value sets, and user exit details of special/pair validated value sets. It includes the values, the normalized value hierarchy, value qualifier values, security rules, security rule lines, security rule usage details, rollup groups, or value hierarchies for the value set.

The key for this entity is FLEX_VALUE_SET_NAME.

Example

```
>FNDLOAD apps/password 0 Y DOWNLOAD @FND:admin/import/afffload.lct
out.ldt \
 VALUE_SET FLEX_VALUE_SET_NAME="Loader_Test"

>FNDLOAD apps/password 0 Y UPLOAD @FND:admin/import/afffload.lct out.ldt
-
```

Descriptive Flexfields

The entity DESC_FLEX includes context column, attribute columns, context, and segment details. This entity references the VALUE_SET for the value set used by a given SEGMENT.

The key is composed of APPLICATION_SHORT_NAME and DESCRIPTIVE_FLEXFIELD_NAME.

Example

```
>FNDLOAD apps/password 0 Y DOWNLOAD @FND:admin/import/afffload.lct
out.ldt \
DESC_FLEX APPLICATION_SHORT_NAME="FND"
DESCRIPTIVE_FLEXFIELD_NAME="FND_FLEX_TEST"

>FNDLOAD apps/password 0 Y UPLOAD @FND:admin/import/afffload.lct out.ldt
-
```

Key Flexfields

The entity KEY_FLEX includes the unique ID column, structure column, segment columns, flexfield qualifier, segment qualifier, structure, Account Generator workflow process, shorthand alias, cross-validation rule, cross-validation rule line, segment, flexfield qualifier assignment, and segment qualifier assignment details.

References VALUE_SET for the value set used by the given segment.

The key is composed of APPLICATION_SHORT_NAME and ID_FLEX_CODE.

Example

```
>FNDLOAD apps/password 0 Y DOWNLOAD @FND:admin/import/afffload.lct
out.ldt \
KEY_FLEX APPLICATION_SHORT_NAME="SQLGL" ID_FLEX_CODE="GL#"

>FNDLOAD apps/password 0 Y UPLOAD @FND:admin/import/afffload.lct out.ldt
-
```

Folders Configuration File

The folder configuration file fndfold.lct downloads and uploads folder definitions.

The following table lists the entities, sub-entities (if any), and download parameters for this configuration file.

Entity	Sub-entities, if any	Download Parameters
FND_FOLDERS	FND_FOLDER_COLUMNS	NAME
	FND_DEFAULT_FOLDERS	OBJECT

The entity definition is:

```
DEFINE FND_FOLDERS

   KEY   FOLDER_ID             NUMBER
   KEY   OBJECT                VARCHAR2(30)
   KEY   FOLDER_NAME           VARCHAR2(80)
   KEY   LANGUAGE              VARCHAR2(4)
   KEY   CREATED_BY            VARCHAR2(4000)
   KEY   CREATOR               VARCHAR2(4000)
   BASE  WINDOW_WIDTH          VARCHAR2(4000)
   BASE  PUBLIC_FLAG           VARCHAR2(1)
   BASE  AUTOQUERY_FLAG        VARCHAR2(1)
   CTX   LAST_UPDATE_DATE      VARCHAR2(11)
   CTX   OWNER                 varchar2(4000)
   BASE  WHERE_CLAUSE          VARCHAR2(2000)
   BASE  ORDER_BY              VARCHAR2(30)

  DEFINE FND_FOLDER_COLUMNS

     KEY   FOLDER_ID            NUMBER
     KEY   ITEM_NAME            VARCHAR2(30)
     BASE  DISPLAY_MODE         VARCHAR2(1)
     BASE  SEQUENCE             VARCHAR2(4000)
     CTX   LAST_UPDATE_DATE     VARCHAR2(11)
     CTX   OWNER                varchar2(4000)
     BASE  ITEM_WIDTH           VARCHAR2(4000)
     TRANS ITEM_PROMPT          VARCHAR2(80)
     BASE  X_POSITION           VARCHAR2(4000)
     BASE  Y_POSITION           VARCHAR2(4000)

  END FND_FOLDER_COLUMNS

  DEFINE FND_DEFAULT_FOLDERS
     KEY   FOLDER_ID              NUMBER
     KEY   OBJECT                 VARCHAR2(30)
     KEY   USER_RESP_TYPE         varchar2(10)
     KEY   USER_ID                varchar2(240)
     CTX   LAST_UPDATE_DATE       VARCHAR2(11)
     CTX   OWNER                  varchar2(4000)
     BASE  APPLICATION_SHORT_NAME VARCHAR2(4000)
     BASE  BEHAVIOR_MODE          VARCHAR2(1)

  END FND_DEFAULT_FOLDERS
```

Additional Considerations

Please note the following when working with folder definitions:

- To change the language you are downloading, set the environment variable NLS_LANG before running the loader.

  Language is a striping column. Folder data is never seeded or translated. As users operate the screen and make customizations, their changes are saved in their current language.

  Do not, for example, create folders in the English language, download the folder definitions and then upload them into another language because all the prompts would remain in English and the WHERE clause saved with the folder may be language sensitive.

- The user who is logged in and creates a given folder is by default the owner of that folder.
- A system administrator can change an owner of a folder using the Administer Folders form under the System Administrator responsibility. A default folder can be assigned to users and responsibilities. The default folder assignments are also handled by the folder loader.
- Make sure the owner and responsibility that is assigned to a folder is in the destination database. If the owner or assigned responsibility of a folder is only on the source database, it will not be uploaded to the destination database.

Download Examples

To download all folders

```
FNDLOAD username/password@database 0 Y DOWNLOAD
$FND_TOP/patch/120/import/fndfold.lct <name of file>.ldt FND_FOLDERS
```

To download folders by "friendly" names

```
FNDLOAD username/password@database 0 Y DOWNLOAD
$FND_TOP/patch/120/import/fndfold.lct <name of file>.ldt FND_FOLDERS
NAME="<name of folder>"
```

Example:

```
FNDLOAD username/password@database 0 Y DOWNLOAD
$FND_TOP/patch/120/import/fndfold.lct <name of file>.ldt FND_FOLDERS
NAME="Receipts Summary Basic"
```

Note: The name of the folder is case-sensitive.

To download folders by internal object names

```
FNDLOAD username/password@database 0 Y DOWNLOAD
$FND_TOP/patch/120/import/fndfold.lct <name of file>.ldt FND_FOLDERS
OBJECT="<internal object name>"
```

Example:

```
FNDLOAD username/password@database 0 Y DOWNLOAD
$FND_TOP/patch/120/import/fndfold.lct <name of file>.ldt FND_FOLDERS
OBJECT="ARXRWRCT"
```

Upload Example

To upload folders:

```
FNDLOAD username/password@database 0 Y UPLOAD
$FND_TOP/patch/120/import/fndfold.lct <name of file>.ldt
```

Lookups Configuration File

Use the file aflvmlu.lct for loading Lookup types and Lookups values.

The following table lists the entities, sub-entities (if any), and download parameters for this configuration file.

Entity	Sub-entities, if any	Download Parameters
FND_LOOKUP_TYPE	FND_LOOKUP_VALUE	VIEW_APPSNAME LOOKUP_TYPE SECURITY_GROUP

The entity definition is:

```
DEFINE FND_LOOKUP_TYPE
  KEY   VIEW_APPSNAME     VARCHAR2(50)
  KEY   LOOKUP_TYPE       VARCHAR2(30)
  CTX   APPLICATION_SHORT_NAME  VARCHAR2(50)
  BASE  CUSTOMIZATION_LEVEL   VARCHAR2(1)
  CTX   OWNER         VARCHAR2(6)
  TRANS MEANING        VARCHAR2(80)
  TRANS DESCRIPTION       VARCHAR2(240)

  DEFINE FND_LOOKUP_VALUE
    KEY   LOOKUP_CODE    VARCHAR2(30)
    BASE  ENABLED_FLAG    VARCHAR2(1)
    BASE  START_DATE_ACTIVE  VARCHAR2(10)
    BASE  END_DATE_ACTIVE   VARCHAR2(10)
    BASE  TERRITORY_CODE   VARCHAR2(2)
    BASE  TAG                   VARCHAR2(30)
    BASE  ATTRIBUTE_CATEGORY  VARCHAR2(30)
    BASE  ATTRIBUTE1     VARCHAR2(150)
    BASE  ATTRIBUTE2        VARCHAR2(150)
    BASE  ATTRIBUTE3       VARCHAR2(150)
    BASE  ATTRIBUTE4       VARCHAR2(150)
    BASE  ATTRIBUTE5       VARCHAR2(150)
    BASE  ATTRIBUTE6       VARCHAR2(150)
    BASE  ATTRIBUTE7       VARCHAR2(150)
    BASE  ATTRIBUTE8        VARCHAR2(150)
    BASE  ATTRIBUTE9        VARCHAR2(150)
    BASE  ATTRIBUTE10       VARCHAR2(150)
    BASE  ATTRIBUTE11       VARCHAR2(150)
    BASE  ATTRIBUTE12        VARCHAR2(150)
    BASE  ATTRIBUTE13      VARCHAR2(150)
    BASE  ATTRIBUTE14       VARCHAR2(150)
    BASE  ATTRIBUTE15       VARCHAR2(150)
    CTX   OWNER             VARCHAR2(6)
    TRANS MEANING          VARCHAR2(80)
    TRANS DESCRIPTION      VARCHAR2(240)
  END FND_LOOKUP_VALUE
END FND_LOOKUP_TYPE
```

Messages Configuration File

Use the file afmdmsg.lct for uploading and downloading messages in a database.

Use the Generic Loader and afmdmsg.lct for transferring messages between databases only. Use the Message Dictionary Generator for moving messages into binary runtime files and readable text files. See: Message Dictionary Generator, page 13-23.

The following table lists the entities, sub-entities (if any), and download parameters for this configuration file.

Entity	Sub-entities, if any	Download Parameters
FND_NEW_MESSAGES		APPLICATION_SHORT_NAME MESSAGE_NAME

The entity definition is:

> **Note:** to change the language you are downloading, set the environment variable NLS_LANG before running the loader.

```
DEFINE FND_NEW_MESSAGES
  KEY   APPLICATION_SHORT_NAME   VARCHAR2(50)
  KEY   MESSAGE_NAME             VARCHAR2(30)
  CTX   OWNER                    VARCHAR2(7)
  CTX   MESSAGE_NUMBER           VARCHAR2(50)
  TRANS MESSAGE_TEXT             VARCHAR2(2000)
  CTX   DESCRIPTION              VARCHAR2(240)
  CTX   TYPE                     VARCHAR2(30)
  CTX   MAX_LENGTH               NUMBER
END FND_NEW_MESSAGES
```

Profile Options and Profile Values Configuration File

Use the file afscprof.lct for loading profile options and profile values.

> **Note:** For downloading in previous releases, a NULL profile option value in the database downloaded as a value of NULL; now, if a NULL value exists in the database, nothing is downloaded for it. For uploading in previous releases, if a NULL profile option value exists in the loader data file (indicated by the absence of a value row or the value was ""), then a NULL value was inserted into the database. Now, if a NULL value exists in the loader data file, the corresponding value is deleted from the database upon uploading.

The following table lists the entities, sub-entities (if any), and download parameters for

this configuration file.

Entity	Sub-entities, if any	Download Parameters
PROFILE	FND_PROFILE_ OPTION_VALUES	PROFILE_NAME APPLICATION_SHORT_NA ME

The entity definition is:

```
DEFINE PROFILE
  KEY   PROFILE_NAME      VARCHAR2(80)
  CTX   OWNER         VARCHAR2(7)
  CTX   APPLICATION_SHORT_NAME   VARCHAR2(50)
  TRANS USER_PROFILE_OPTION_NAME  VARCHAR2(240)
  TRANS DESCRIPTION       VARCHAR2(240)
  BASE  USER_CHANGEABLE_FLAG   VARCHAR2(1)
  BASE  USER_VISIBLE_FLAG    VARCHAR2(1)
  BASE  READ_ALLOWED_FLAG    VARCHAR2(1)
  BASE  WRITE_ALLOWED_FLAG    VARCHAR2(1)
  BASE  SITE_ENABLED_FLAG    VARCHAR2(1)
  BASE  SITE_UPDATE_ALLOWED_FLAG  VARCHAR2(1)
  BASE  APP_ENABLED_FLAG     VARCHAR2(1)
  BASE  APP_UPDATE_ALLOWED_FLAG  VARCHAR2(1)
  BASE  RESP_ENABLED_FLAG    VARCHAR2(1)
  BASE  RESP_UPDATE_ALLOWED_FLAG  VARCHAR2(1)
  BASE  USER_ENABLED_FLAG    VARCHAR2(1)
  BASE  USER_UPDATE_ALLOWED_FLAG  VARCHAR2(1)
  BASE  START_DATE_ACTIVE    VARCHAR2(11)
  BASE  END_DATE_ACTIVE     VARCHAR2(11)
  BASE  SQL_VALIDATION      VARCHAR2(2000)

  DEFINE FND_PROFILE_OPTION_VALUES
    KEY   LEVEL        VARCHAR2(50)
    KEY   LEVEL_VALUE     VARCHAR2(100)
    KEY   LEVEL_VALUE_APP    VARCHAR2(50)
    CTX   OWNER       VARCHAR2(7)
    BASE  PROFILE_OPTION_VALUE  VARCHAR2(240)
  END FND_PROFILE_OPTION_VALUES
END PROFILE
```

Request Groups Configuration File

Use the file afcpreqg.lct for loading request group data.

The following table lists the entities, sub-entities (if any), and download parameters for this configuration file.

Entity	Sub-entities, if any	Download Parameters
REQUEST_GROUP	REQUEST_GROUP_ UNIT	REQUEST_GROUP_NAME APPLICATION_SHORT_NAME

The entity definition is:

```
DEFINE REQUEST_GROUP
  KEY   REQUEST_GROUP_NAME    VARCHAR2(30)
  KEY   APPLICATION_SHORT_NAME  VARCHAR2(50)
  CTX   OWNER               VARCHAR2(7)
  TRANS DESCRIPTION          VARCHAR2(800)
  BASE  REQUEST_GROUP_CODE    VARCHAR2(30)
END REQUEST_GROUP
```

Security Information Configuration File

Use the file afsload.lct for downloading and uploading forms, functions, menus, and menu entries.

The following table lists the entities, sub-entities (if any), and download parameters for this configuration file.

Entity	Sub-entities, if any	Download Parameters
FORM		FORM_APP_SHORT_NAME, FORM_NAME
FUNCTION		FUNC_APP_SHORT_NAME FUNCTION_NAME
MENU	ENTRY	MENU_NAME
ENTRY		[None]

The entity definition is:

```
DEFINE FORM
  KEY   APPLICATION_SHORT_NAME VARCHAR2(50)
  KEY   FORM_NAME VARCHAR2(30)
  TRANS USER_FORM_NAME VARCHAR2(80)
  TRANS DESCRIPTION VARCHAR2(240)
  CTX   OWNER VARCHAR2(7)
END FORM

DEFINE FUNCTION
  KEY   FUNCTION_NAME VARCHAR2(30)
  BASE  FORM REFERENCES FORM
  BASE  TYPE VARCHAR2(30)
  BASE  PARAMETERS VARCHAR2(2000)
  BASE  WEB_HOST_NAME VARCHAR2(80)
  BASE  WEB_AGENT_NAME VARCHAR2(80)
  BASE  WEB_HTML_CALL VARCHAR2(240)
  BASE  WEB_ENCRYPT_PARAMETERS VARCHAR2(1)
  BASE  WEB_SECURED VARCHAR2(1)
  BASE  WEB_ICON VARCHAR2(30)
  TRANS USER_FUNCTION_NAME VARCHAR2(80)
  TRANS DESCRIPTION VARCHAR2(240)
  CTX   OWNER VARCHAR2(7)
END FUNCTION

DEFINE MENU
  KEY   MENU_NAME VARCHAR2(30)
  TRANS USER_MENU_NAME VARCHAR2(80)
  TRANS DESCRIPTION VARCHAR2(240)
  CTX   OWNER VARCHAR2(7)

 DEFINE ENTRY
    TRANS PROMPT VARCHAR2(60)
    TRANS DESCRIPTION VARCHAR2(240)
    CTX   SUBMENU REFERENCES MENU
    CTX   FUNCTION REFERENCES FUNCTION
    CTX   OWNER VARCHAR2(7)
  END ENTRY
END MENU
```

Message Dictionary Generator

The Message Dictionary Generator (FNDMDGEN) is a concurrent program that generates binary runtime files from the database for Oracle E-Business Suite Message Dictionary messages. The following sections describe the operation of the Message Dictionary Generator.

For more information on using the Message Dictionary and creating messages, see the *Oracle E-Business Suite Developer's Guide*.

> **Note:** Use the Generic Loader and corresponding configuration file for uploading and downloading message text files into a database.

Message Repositories

Message information is stored in two different repositories, each of which has its own

format and serves a specific need. Following is a description for each of the message repositories, including the message attributes they store.

Database

The FND_NEW_MESSAGES table in the database stores all Oracle E-Business Suite messages for all languages. Database messages are directly used only by the stored procedure Message Dictionary API. Database message data can be edited using the Messages form.

Database Attributes are: APPLICATION, LANGUAGE, NAME, NUMBER, TEXT, DESCRIPTION

Runtime

A runtime binary file stores the messages for a single application and a single language. The file is optimized for rapid lookup of individual messages by message NAME.

A runtime file is located in:

```
<APPL_TOP>/$APPLMSG/<LANGUAGE>.msb
```

where <APPL_TOP> is the application basepath, APPLMSG is an environment variable whose usual value is "mesg", and <LANGUAGE> is the NLS language code (for example: 'US', or 'F'.). A typical message file would be $FND_TOP/mesg/US.msb.

Runtime Attributes are: NAME, NUMBER, TEXT

Usage

The help that you get when you invoke the Message Dictionary Generator without any program arguments (i.e., FNDMDGEN dbuser/dbpassword 0 Y) is:

```
FNDMDGEN <Oracle ID/password> 0 Y <language codename> [application
shortname] [mode] [filename] \
```

where mode is:

DB_TO_ RUNTIME From Database to Runtime file (.msb)

Wildcards

Either <language codename> or [application shortname] can be wildcarded by passing the value "ALL". The following describes how wildcards are used:

From DB Messages come from the FND_NEW_MESSAGES table. Wildcards match all the messages in the database.

To RUNTIME In the case of wildcards, separate runtime files are created for each combination of language and application.

Generic File Manager Access Utility (FNDGFU)

The Generic File Manager (GFM) is a set of PL/SQL procedures that leverages Oracle HTTP Server functionality to provide generic upload and download capabilities of unstructured data between a client (usually a web browser) and a database.

FNDGFU is an access utility that allows the upload of files from the local file system to the GFM database file system. It supports simple uploads of single files as well as bulk uploads of many files. FNDGFU also offers a download option that provides a convenient and quick means of retrieving the contents of large objects (LOBs) if the file identifier is known.

This utility is used in uploading help files for the Oracle E-Business Suite online help system. For information on downloading and uploading Oracle E-Business Suite help files, see: Downloading and Uploading Help Files, page 7-2.

To delete files loaded to the database run the Purge Obsolete Generic File Manager Data concurrent program.

Usage

FNDGFU is located in the $FND_TOP/bin directory. Putting this directory on your path will allow you to invoke FNDGFU easily.

Upload files to the GFM

To upload files using FNDGFU use the following syntax:

```
FNDGFU <logon> [param] <filenames>
```

where

<logon> Specifies a standard Oracle logon string of the form username/password. To specify a particular database, append an @ sign and the database SID (@database).

[param] Includes the following parameters (in any order) as appropriate:

PROGRAM_NAME=<name> specifies the name of the program on whose behalf the LOB is to be maintained.

PROGRAM_TAG=<name> specifies the program tag, which is a string used by the GFM client program to further categorize the LOB.

LANGUAGE=<language_code> specifies the language of the file.

PLS_CALLBACK=<plsql procedure> specifies the procedure to execute once for each uploaded file. The

procedure must accept file_id as its only parameter. FNDGFU will call the specified procedure after each uploaded file, passing in the new file identifier, for example: PLS_CALLBACK=mypackage.myprocedure.

CONTENT_TYPE=<mime_type> specifies the default mime type to use for uploaded files not qualified by a content map.

CONTENT_MAP=<contentmapfile> specifies a text file that maps filename suffixes onto content types. The text file consists of lines of the form <suffix>=<mime_type> where suffix is any string matched against the end of the filename. For example: ".txt = text/plain", ".html = text/html", and ".ps = application/postscript".

<filenames> Specifies the files to upload. Any number of files may be uploaded.

Download files from the GFM

To download a file using the FNDGFU utility, use the following syntax:

```
FNDGFU <logon> DOWNLOAD=<fileid> [LINE_BREAKS=<mode>] [filename]
```

where

<logon> Specifies a standard Oracle logon string of the form username/password. To specify a particular database, append an @ sign and the database SID (@database).

<fileid> Specifies the identifier of the large object (LOB) to download.

<mode> Specifies how to treat line breaks for a text document. This parameter is ignored for nontext content. The following values are valid:

LF - Line breaks will be represented using "/n" in the downloaded output. This is the default mode if the LINE_BREAK parameter is omitted.

CRLF - Line breaks will be left in the canonical format.

[filename] Specifies the file into which to download. If omitted, downloaded contents are streamed to the standard output.

Example of FNDGFU Upload

The FNDGFU utility can be used to upload new or changed help files. Use the

following arguments to upload help files:

```
FNDGFU <APPS username>/password 0 Y PROGRAM_NAME=FND_HELP
PROGRAM_TAG=<application>:<custom_level> CONTENT_TYPE=<mime_type>
LANGUAGE=<language_code> <filenames>
```

where

<APPS username>/*password*	is the APPS schema username/password. To specify a particular database, append an @ sign and the database SID (@database).
<application>	is the Application short name.
<custom_level>	is the files' customization level. Use the number 100 or above for customized help files. To replace previously uploaded files, use the same customization level when uploading the new files. To override previously uploaded files without deleting them from the database, use a higher customization level.
<mime_type>	is the files' MIME type.
<language_code>	is the files' language code.
<filenames>	is a space-separated list of files to upload, or a filename glob in the current directory.

Enter all arguments on a single command line. They may appear on separate lines here and in the examples that follow depending on the display medium.

Example 1

```
FNDGFU apps/password @devdb 0 Y PROGRAM_NAME=FND_HELP PROGRAM_TAG=GL:100
CONTENT_TYPE=text/html LANGUAGE=US file1.htm file2.htm
```

- connects to apps/*password*@devdb
- identifies uploaded files as part of Oracle General Ledger (GL) help
- identifies the uploaded files' customization level as 100
- identifies their MIME type as text/html
- identifies their language as US English (US)
- uploads the two specified .htm files in the current directory (in UNIX)

Example 2

```
FNDGFU apps/password@custdb 0 Y PROGRAM_NAME=FND_HELP
PROGRAM_TAG=FND:100 CONTENT_TYPE=image/gif *.gif
```

- connects to apps/*password*@custdb

- identifies uploaded files as part of Oracle Application Object Library (FND) help
- identifies the uploaded files' customization level as 100
- identifies their MIME type as image/gif
- does not identify their language, which defaults to userenv('LANG')
- uploads all .gif files in the current directory (in UNIX)

Purging Generic File Manager Data

To purge uploaded files from the Generic File Manager, run the concurrent program, Purge Obsolete Generic File Manager Data.

This concurrent program should also be used to periodically expunge expired data. It is recommended that you schedule this program to run every day or so, using the default parameter values.

Purge Obsolete Generic File Manager Data

To purge uploaded files from the Generic File Manager, run the concurrent program, Purge Obsolete Generic File Manager Data.

This concurrent program should also be used to periodically delete obsolete data. It is recommended that you schedule this program to run every day or so, using the default parameter values.

Program Parameters

Expired

Enter "Y" if you want to purge expired data only. Enter "N" if you want the purge to include all data. The default is "Y."

Program Name

Enter the program name(s) to process. Leave blank to process all programs.

Program Tag

Enter the program tag(s) to process. Leave blank to process all program tags.

14

Oracle Applications Tablespace Model

Introduction to Oracle Applications Tablespace Model

The Oracle Applications Tablespace Model (sometimes called OATM or the Oracle E-Business Suite Tablespace Model) uses twelve consolidated tablespaces (including three system tablespaces: temporary, system and undo segments) and provides support for locally managed tablespaces. OATM was introduced in Release 11*i*.10. In prior 11*i* releases of Oracle E-Business Suite, each product was allocated two tablespaces, one for data and one for indexes. The Migration Utility is a menu-based PERL program and a series of sizing estimate reports that enables conversion of Oracle E-Business Suite applications schemas either in a single comprehensive migration or a phased, schema-by-schema migration. In general Oracle recommends performing a single comprehensive migration, however this requires a sufficient amount of down time and disk space. Oracle does not support partial migration of tablespaces. You must still migrate all schemas when performing a phased schema-by-schema migration.

With OATM, each database object is mapped to a tablespace based on its Input/Output characteristics, which include object size, life span, access methods and locking granularity. This model allows for easier maintenance, and reduced space usage for Oracle E-Business Suite.

Another configuration supported with Locally Managed Tablespaces, User Extent Management, is only relevant when Dictionary Tablespaces have been migrated to Locally Managed Tablespaces. Because migration to OATM is performed at the object level and not at the tablespace level, User Extent Management is not relevant. The default recommended configuration in OATM is Uniform Extent Management.

Advantages of Migrating to OATM

Migrating database objects to OATM provides the following benefits:

- Fewer and more consolidated tablespaces

- Locally Managed Tablespaces
- Accounts for the I/O characteristics of an object
- Reclaims space after migration
- Real Application Cluster (RAC) Support

Fewer and More Consolidated Tablespaces

OATM contains twelve locally managed tablespaces for all products, including temporary tablespace, system tablespace, and undo segments. The previous tablespace model contained two tablespaces for each Oracle product resulting in hundreds of tablespaces.

Locally Managed Tablespaces

This model provides support for either Uniform or Auto-allocate extent management, available with Locally Managed Tablespaces. Locally Managed Tablespaces have benefits over Dictionary Tablespaces in the previous model and allow for the sizes of extents to be determined automatically by the system (Auto-allocate). Alternatively, all extents can have the same size (Uniform) in and override object storage options.

OATM implements Automatic segment-space management, a simpler and more efficient way of managing space within a segment. It completely eliminates any need to specify and tune the PCTUSED, FREELISTS, and FREELISTS GROUPS storage parameters for schema objects created in the tablespace. Automatic segment-space management delivers better space utilization than manual segment-space management, and is self tuning because it scales by increasing the number of users, as well as instances. For a Real Application Clusters environment, automatic segment-space management enables dynamic affinity of space to instances, which avoids the hard partitioning of space inherent with using free list groups.

Uniform Extent Size

The value for uniform extent size should be carefully selected based on system requirements. For production environments and large tablespaces like transaction tables, transaction indexes, interfaces, summaries, archives, and media, a uniform extent size of 1MB or 10MB (with caution) should be considered. Choosing an extent size that is too small can result in frequent extensions and performance degradation of the system.

The Release 12 Rapid Install production database is delivered out-of-the-box with locally managed tablespaces with uniform extent sizes of 128Kb. If this size is not the best match for the characteristics of your system, you can follow subsequent re-migration steps to create new tablespaces with the desired uniform extent size and migrate objects to those new tablespaces. The OATM Migration Utility supports all

possible configurations of locally managed tablespaces. You have the flexibility to override the default recommendation of uniform extent size with Auto-Allocate extent management as per your requirements. Uniform extent size is also configurable.

> **Note:** The Oracle database server does not as yet support altering the extent management type or uniform extent size after the locally managed tablespaces have been created. Therefore, if the originally used extent management type or uniform extent size needs to be modified, re-creation of the tablespaces and re-migration of objects is the only available option.

Re-migration steps

Use the customization option to change names of existing OATM tablespaces as listed in the following table:

Tablespace Type	Old Tablespace Name	New Tablespace Name
Transaction Tables	APPS_TS_TX_DATA	APPS_TS_TX_DATA_1MB
Transaction Indexes	APPS_TS_TX_IDX	APPS_TS_TX_IDX_1MB
Reference	APPS_TS_SEED	
Interface	APPS_TS_INTERFACE	APPS_TS_INTERFACE_1MB
Summary	APPS_TS_SUMMARY	APPS_TS_SUMMARY_1MB
Nologging	APPS_TS_NOLOGGING	
Advanced Queuing/AQ	APPS_TS_QUEUES	
Media	APPS_TS_MEDIA	APPS_TS_MEDIA_1MB
Archive	APPS_TS_ARCHIVE	APPS_TS_ARCHIVE_1MB

Please note that for the types of tablespaces for which you do not wish to modify the uniform extent size, you should not change the tablespace name and objects will remain in the respective tablespace with the originally selected extent size and no attempt will be made to migrate them. Sizes of those tablespaces should be ignored in the sizing report and 'alter tablespace' statements removed from the 'create tablespace' script before the script is run.

From the main menu, select option 7. Run Customization Steps:

Customization

1. Register new tablespace - tablespace type
2. Change name of the existing tablespace
3. Register object classification
4. Change object classification

Please select "2. Change name of existing tablespace".

```
Enter tablespace type: TRANSACTION_TABLES
Enter new tablespace name: APPS_TS_TX_DATA_1MB
Tablespace name for TRANSACTION_TABLES changed to APPS_TS_TX_DATA_1MB.

Do you want to continue changing tablespace names?[Y]:

Enter tablespace type: TRANSACTION_INDEXES
Enter new tablespace name: APPS_TS_TX_IDX_1MB

...
```

The same steps should be performed for all tablespace types for which you want to change the uniform extent size. When the customizations are complete, please run the steps for the utility described beginning in the section Introduction to the Oracle Applications Tablespace Migration Utility, page 14-9.

- Run the sizing process and create new tablespaces.
- Run migration command generation.
- Complete post-migration steps and drop old tablespaces that have no remaining segments.

I/O Characteristics of an Object

OATM takes into account the following object I/O characteristics of an object:

- size
- life span
- access methods
- locking granularity

Automatic Segment-space Management

Automatic segment-space management is a simpler and more efficient way of managing space within a segment. It completely eliminates any need to specify and tune the PCTUSED, FREELISTS, and FREELISTS GROUPS storage parameters for

schema objects created in the tablespace. Automatic segment-space management delivers better space utilization than manual segment-space management, and is self tuning because it scales with the increasing number of users and instances. For a Real Application Cluster (RAC) environment, automatic segment-space management enables dynamic affinity of space to instances, which avoids the hard partitioning of space inherent with using free list groups.

Reclaims Space After Migration

The tablespace migration utility migrates objects from the existing dictionary-managed tablespaces to locally managed tablespaces with automatic segment management and either uniform or Auto-allocate extent management. As a result, space is better utilized and less wasted. Migration of table and index segments from one tablespace to another also reclaims unused space, especially for indexes that are fragmented when the index columns are inserted, updated or deleted frequently.

Real Application Cluster (RAC) Support

OATM facilitates Real Application Cluster (RAC) support because of its reduced number of tablespaces. RAC is an Oracle database feature that harnesses the processing power of multiple interconnected computers where all active instances can concurrently execute transactions against a shared database disk system. The new tablespace model is critical when implementing RAC on Linux, where currently there is a limitation of 255 raw devices.

Additional Benefits

OATM provides the following additional benefits:

- Facilitates administration and configuration ease
- Increases block-packing to reduce the overall number of buffer gets.

OATM Tablespaces

The advantages of OATM's product tablespaces are best understood in terms of the tablespace model that preceded it. This model contained two tablespaces for each Oracle E-Business Suite product. One tablespace was allocated for tables and one for indexes. In this model, the standard naming convention for tablespaces contained the product's Oracle schema name with a suffix of either "D" for "Data" tablespaces or "X" for "Index" tablespaces. For example, the tablespaces APD and APX were the default tablespaces for Oracle Payables tables and indexes, respectively.

In contrast to the previous tablespace model, OATM contains nine default tablespaces for applications objects in addition to Undo, Temp and System database tablespaces. Indexes on transaction tables are held in a separate tablespace dedicated for transaction

table indexes whereas all other indexes are held in the same tablespace as the parent/base table. All Oracle E-Business Suite product schemas now have a default tablespace set to point to the TRANSACTION_TABLES tablespace type for data objects and the TRANSACTION_INDEXES tablespace type for index objects.

The Oracle Applications Tablespace Model uses Locally Managed Tablespaces and supports either Uniform or Autoallocate extent management. Another configuration supported with Locally Managed Tablespaces - User Extent Management, is of relevance only in case of Dictionary Tablespaces that have been migrated over to Locally Managed Tablespaces. Because migration to OATM is performed at the object level and not at the tablespace level, User Extent Management is not relevant. The default recommended configuration in OATM is Uniform Extent Management. The migration utility recommends the default of 128k uniform extents which can be changed to suit the customer database. Note however that if you use another extent size, that size must still comply with the minimum extent size required for your Oracle Database version. If you encounter issues when using a lower size than the default, increasing the uniform extent size may help resolve those issues.

Tablespace types are listed in the following table:

Tablespace Types

Tablespace Type	Tablespace Name	Content
Transaction Tables	APPS_TS_TX_DATA	Tables that contain transactional data.
Transaction Indexes	APPS_TS_TX_IDX	Indexes on transaction tables.
Reference	APPS_TS_SEED	Reference and setup data and indexes.
Interface	APPS_TS_INTERFACE	Interface and temporary data and indexes.
Summary	APPS_TS_SUMMARY	Summary management objects, such as materialized views, fact tables, and other objects that record summary information.
Nologging	APPS_TS_NOLOGGING	Materialized views not used for summary management and temporary objects.

Tablespace Type	Tablespace Name	Content
Advanced Queuing/AQ	APPS_TS_QUEUES	Advanced Queuing and dependent tables and indexes.
Media	APPS_TS_MEDIA	Multimedia objects, such as text, video, sound, graphics, and spatial data.
Archive	APPS_TS_ARCHIVE	Tables that contain archived purge-related data.
Undo	UNDO	Automatic Undo Management (AUM) tablespace. UNDO segments are identical to ROLLBACK segments when AUM is enabled.
Temp	TEMP	Temporary tablespace for global temporary table, sorts, and hash joins.
System	SYSTEM	System tablespace used by the Oracle Database

Tablespace Classification

OATM relies on specific explicit and implicit classification rules that are determined based on storage considerations for the object type in question. The Oracle Tablespace Migration Utility migrates objects based on these rules. The following table contains rules for implicit classifications that are applied in OATM, based on object types. Objects that do not have an implicit classification rule or an explicit object classification are migrated to the default tablespaces of the schema in which they reside.

Explicit Classification Rules

Explicit object classifications are seeded by Oracle based on the I/O characteristics of the object.

Implicit Classification Rules

The following table contains implicit classification rules for the Oracle Applications Tablespace Migration Utility.

Implicit Classification Rules

Object Type	Tablespace_Type
AQ Tables	AQ
IOTs (Index Organized Tables)	Transaction_Tables
Materialized Views	Summary
Materialized View Logs	Summary
All other Indexes	Same Tablespace type as the table
Domain Indexes	Transaction_Indexes
Indexes on Transaction Tables	Transaction_Indexes

Customizations and Extensions

The Oracle Applications Tablespace Migration Utility is primarily designed to migrate tables, indexes, materialized views, materialized view logs and other database objects that are owned by standard Oracle E-Business Suite schemas from their existing tablespace model to OATM. Custom or third party schemas can also be migrated using the Oracle Applications Tablespace Migration Utility, customer preferred methods, or a database management tool such as the Oracle Enterprise Manager (OEM). Custom objects in standard Oracle E-Business Suite product schemas are migrated by default.

The Oracle Applications Tablespace Migration Utility enables the following customizations:

- Changing tablespace names
- Registering custom tablespace types
- Registering custom object-tablespace classifications
- Changing existing object classifications.

Migrating Custom or Third Party schemas

Login to the Forms-based version of Oracle E-Business Suite with the System Administrator Responsibility. Navigate to Security -> ORACLE -> Register and register

the external schema(s) if they are not already registered. Set Privilege to "Enabled".

Preventing Migration of Specific Schemas

In some cases, you may not want to migrate some of your schemas such as non-Oracle schemas that are registered with Oracle E-Business Suite. To accomplish this, you must disable those schemas by accessing System Administrator responsibility -> Security -> ORACLE -> Register and then selecting either "External" or "Disabled" for the schema in question. Conversely, if you want to flag specific schemas for migration, you can enable them by accessing System Administrator responsibility -> Security -> ORACLE -> Register and then selecting "Enabled".

Introduction to the Oracle Applications Tablespace Migration Utility

The Tablespace Migration Utility is a menu-based PERL program that enables you to estimate future space requirement for the tablespaces and to migrate the Oracle E-Business Suite database to OATM. Log files are available for user viewing and are created in the working directory from which your run the PERL program. The Log file name and location are displayed once you choose the required option.

The Tablespace Migration Utility enables you to perform either a single, comprehensive migration of all schemas or a phased, schema-by-schema migration. To minimize downtime, Oracle recommends that you perform the single comprehensive migration of all schemas, however this requires a sufficient amount of down time and available disk space. If , you do not have sufficient down time or disk space to accomplish this, then you can run the phased schema-by-schema migration. Once you migrate an object from its existing tablespace to OATM, this process cannot be reversed. Oracle does not support the rollback of schemas migrated to OATM in a phased schema-by-schema migration. The only method for achieving this result is to recover the migrated schemas from a backup.

> **Note:** Once you initiate migration of one or more schemas to OATM it is not possible to perform additional migrations from a different PERL menu. You must wait until one migration is completed before beginning another.

Planning for Migration

Sizing Requirements

Whenever possible, Oracle recommends the following:

- perform a single comprehensive migration of all schemas instead of performing a phased schema-by-schema migration.

- perform test runs to determine the amount of down time required to perform a comprehensive migration of all schemas
- secure twice as much disk space as your existing space to perform the a single comprehensive migration of all schemas, and to be operational using the new model.

Setting Up the Tablespace Migration Utility

Setting up the Tablespace Migration Utility

When you first install the Tablespace Migration Utility, it does the following:

- Copies the PERL menu script, fndtsmig.pl, to the FND_TOP/bin directory.
- Copies the SQL scripts for the Tablespace Migration utility to FND_TOP/patch/120/sql directory.
- Copies the PLS files for the Tablespace Migration utility to the FND_TOP/patch/120/sql directory and creates packages in the database.
- Compiles the Java files into the packages oracle.apps.fnd.tsmig, in FND_TOP/java/apps.zip file.
- Creates and seeds the following tables:
  - FND_TABLESPACES
  - FND_OBJECT_TABLESPACES
  - FND_TS_SIZING
  - FND_TS_MIG_CMDS
  - FND_TS_MIG_RULES
  - FND_TS_PROD_INSTS

Invoking the Tablespace Migration Utility Main Menu

Invoke the Tablespace Migration Utility main menu by performing the following:

1. Run the fndtsmig.pl PERL script:

   ```
   perl $FND_TOP/bin/fndtsmig.pl.
   ```

2. Provide the following information when prompted to access the Tablespace

Migration Utility main menu:

- OATM configuration file
- APPL_TOP[]
- FND_TOP[/fnddev/fnd/12.0]
- Database Connect String [dummy]: atgtsqa
- Password for your 'SYSTEM' ORACLE Schema
- Password for your 'SYS' ORACLE Schema
- Oracle Application Object Library Schema name [APPS]
- Password for APPS
- APPLSYS Schema Name [APPLSYS]

Information can be provided in an interactive manner or by providing an OATM configuration file with all information already specified. An OATM configuration file has the following format with the following valid tags:

```
APPL_TOP                  - valid APPL_TOP value or $ENV$
FND_TOP                   - valid FND_TOP value or $ENV$
APPS_SCHEMA               - valid apps schema name
APPLSYS_SCHEMA            - valid applsys schema name
ALLOC_TYPE                - UNIFORM/AUTOALLOCATE
UES                        - valid integer (uniform extent size)
DBF_DIR                   - valid directory for generated database file
INDIVIDUAL_DATAFILE_SIZE  - maximum datafile size
NUM_WORKER                - integer (number of concurrent workers.)
MIGRATION_SCHEMA          - % or comma separated list of schemas
CONNECT_STRING            - database connect string
AUTO_START_MIGRATION      - should migration be started automatically
after
                            preparatory steps are completed
Y(default)/N
```

ENV is a reserved word for the OATM configuration file. If ENV is used as a value for a specific token, then that token's real value will be derived from the customer's environment dynamically during the runtime.

Example

```
<!-- OATM Migration Configuration File>
<OATM>
  <APPL_TOP> $ENV$ </APPL_TOP>
  <FND_TOP> $ENV$ </FND_TOP>
  <CONNECT_STRING> atgtsmqa </CONNECT_STRING>
  <APPS_SCHEMA> APPS </APPS_SCHEMA>
  <APPLSYS_SCHEMA> APPLSYS </APPLSYS_SCHEMA>
  <ALLOC_TYPE> U  </ALLOC_TYPE>
  <UES> 1024 </UES>
  <DBF_DIR> /slot05/oracle/atgtsmqadata/ </DBF_DIR>
  <INDIVIDUAL_DATAFILE_SIZE> 2000 </INDIVIDUAL_DATAFILE_SIZE>
  <NUM_WORKER> 8 </NUM_WORKER>
  <MIGRATION_SCHEMA>  %   </MIGRATION_SCHEMA>
  < AUTO_START_MIGRATION >  Y  </ AUTO_START_MIGRATION >
</OATM>
```

If an OATM configuration file is provided, values specified in the configuration file can still be overridden by values provided interactively. A summary of all provided information will be displayed and can be reviewed and corrected if needed.

> **Note:** Please note that the configuration file will not be modified accordingly by information entered interactively. Values provided interactively will have an effect only for the OATM session when provided. No new tags added to the configuration files will be recognized. All values specified in the OATM configuration file above are just an example, not Oracle recommended values.

Understanding the Tablespace Migration Utility Main Menu

The Tablespace Migration Utility main menu lists six required sequential steps and one optional step to migrate your database objects to OATM. These steps are categorized in three phases. In Phase 1, you perform the necessary preparation steps for migrating your objects to OATM. In Phase 2, you perform the necessary steps to migrate your objects to OATM and in Phase 3 you run the required post migration steps.

1. (Preparatory Step) Generate Migration Sizing Reports
2. (Preparatory Step) Create New Tablespaces
3. (Preparatory Step) Generate Migration Commands
4. (Migration Step) Execute Migration Commands
5. (Migration Step) Run Migration Status Reports
6. (Post Migration Step) Run Post Migration Steps
7. (Optional) Run Customization Steps

8. (Optional, batch mode) Run Migration in Batch Mode

Steps 1, 2, 3 can be executed while Oracle E-Business Suite is still available to users. Step 4 must be executed when Oracle E-Business Suite is not available to users and steps 5 and 6 must be completed before making Oracle E-Business Suite available to users again. If you choose to perform optional Step 7 you should do so before the other steps.

> **Caution:** Oracle highly recommends that you back up your database after performing Step 3, which is the final preparatory step. You should have a backup copy of your database before performing the subsequent migration steps.

Backing up the Database

Oracle highly recommends that you perform a backup of your database ***twice*** as follows;

1. **Copy of previous tablespace model**. Back up your database before performing any of the migration steps. Because Oracle does not support the rollback of migrated database objects, this is the only available method for restoring your previous tablespace model.

2. **Copy of database that has been prepared for migration**. Backup your database after performing Step 3: Generate Migration Commands. This enables you to migrate your database objects to OATM using your last good copy of a database that has been prepared for migration.

If you choose to use the menu option 8. Migration in Batch mode, you should perform the database backup before the migration and/or after the preparatory steps are completed, if you choose to run the preparatory steps separately.

Phase 1: Preparatory Steps

Step 1: Generate Migration Sizing Reports

Select Step 1: Generate Migration Sizing Reports to access a list of reports that help you to gauge the space requirements for the new tablespace model and that assist you in determining which migration approach best suits your requirements. The reports perform sizing estimation by executing a program that calculates the size of each object using package DBMS_SPACE.UNUSED_SPACE and by populating table FND_TS_SIZING. The sizing reports use the data in this table to display the required information. The Plan Migration menu contains the following options:

1. Calculate total space required by each new tablespace to migrate all Oracle E-Business Suite product schemas

2. Calculate total space required by each new tablespace to migrate each Oracle E-Business Suite product schema (relevant for a schema-by-schema migration)

3. Calculate total space required by each Oracle E-Business Suite schema, with details for each object

4. Display Sizing Exception report

Option 1: Calculate Total Space Required by Each New Tablespace to Migrate all Oracle E-Business Suite Product Schemas

Choose option 1 to calculate the total space required by each new tablespace when performing a single comprehensive migration of all Oracle E-Business Suite product schemas, and to generate the report fndtrep1.txt. Before running the report, the program prompts you to specify whether the information in the sizing table is current or must be updated. Enter the following to calculate space requirements for performing a single comprehensive migration of all schemas:

- Whether the sizing information is current or must be gathered before running the report.

- If you selected "Y" for the previous option, provide the extent management type for the tablespaces, since the space requirements are dependent on this.

- If Uniform Extent Management, provide the uniform extent size for the tablespaces or choose the default value provided on the screen.

Example

```
Sizing Program was last run on 02-SEP-03
Do you want to run the Sizing program again before running the report
[N]:y
Enter the Extent Allocation type A(utoallocate) or U(niform Extent Size)
[U]: U
Enter Uniform Extent Size for the Tablespaces in KBytes[128]:
```

Option 2: Calculate Total Space Required by Each New Tablespace to Migrate Each Oracle E-Business Suite Product Schema

Choose Option 2 to calculate total space required by each new tablespace when migrating individual Oracle E-Business Suite product schemas one at a time, and to generate the report fndtrep2.txt. The program prompts your for the following information:

- Schema name, enter the percent sign (%) for all schemas,

- Whether the sizing information is current or must be gathered before running the report.

- If you selected Y for the previous option, provide the extent management type for the tablespaces, since the space requirements are dependent on this.

- If Uniform Extent Management, provide the uniform extent size for the tablespaces or choose the default value provided on the screen.

Example

```
Enter the Schema name: <HR>

Sizing Program was last run on 02-SEP-03
Do you want to run the Sizing program again before running the report
[N]:
```

Option 3: Calculate Total Space Required by Each Oracle E-Business Suite Schema with Details for Each Object

Choose option 3 to calculate total space required by each Oracle E-Business Suite schema with details for each object and to generate the report fndtrep4.txt. The program prompts your for the following information:

- Schema name, enter the percent sign (%) for all schemas,
- Whether the sizing information is current or must be gathered before running the report.
- If you selected Y for the previous option, provide the extent management type for the tablespaces, since the space requirements are dependent on this.
- If Uniform Extent Management, provide the uniform extent size for the tablespaces or choose the default value provided on the screen.

Example

```
Enter the Schema name: <HR>

Sizing Program was last run on 02-SEP-03
Do you want to run the Sizing program again before running the report
[N]:
```

Option 4: Display Sizing Exception Report

Choose option 4 to generate report fndtrep5.txt, listing all the objects for which sizing estimation generated an error. The program prompts you for the schema name. Enter the percent sign (%) for all schemas.

Example

```
Enter the Schema name: <%>
```

Step 2: Create New Tablespaces

Select Step 2: Create New Tablespaces to create the OATM tablespaces to which you will migrate your database objects. The Create New Tablespaces menu contains the following options:

1. Generate the Tablespace Creation Script
2. Create New Tablespaces

Option 1: Generate the Tablespace Creation Script

Option 1 prompts you for the extent allocation type, such as Autoallocate, Uniform Extent, and Uniform Extent Size, which will be used for creating the new tablespaces. The utility prompts for the name of the directory in which the datafile will be created. For every tablespace created as part of OATM, the utility prompts you for information, such as number and size of the datafiles. The utility will append a sequence number to the tablespace name and a .dbf extension to generated the datafile name. For example, if you enter the datafile directory as "/u01/oradata" and for transaction data tablespace, APPS_TS_TX_DATA, you enter the number of datafiles as 2 and size as 2000M, the utility will create tablespace creation script with 2 datafiles for the tablespace named "/u01/oradata/APPS_TS_TX_DATA01.dbf" and "/u01/oradata/APPS_TS_TX_DATA02.dbf", each of size 2000M. To create datafiles of different sizes or in different locations, you must modify the generated script, crtts.sql.

If you do not have limited disk space, create all tablespaces with the estimated sizes listed in report #1. This will eliminate the need to extend the tablespaces as the migration proceeds. If you do not have enough disk space to create all the tablespaces with the total size, use the estimated values in report #2 <Schema Name> for reference. If your operating system has a limit on the size of a dbf file, ensure you enter a value less than this when prompted.

Example

```
Enter the Extent Allocation type A(utoallocate) or U(niform Extent
Size)[U]:
**************************************************************
The utility will append a sequence number to the tablespace name
and a .dbf extension to generate the datafile names.
Datafile size should not be greater than OS file size limit.
Please edit the generated script to change the file name/size
**************************************************************

Enter the absolute path for the datafiles directory: /u01/oradata
Enter the Number of Datafiles for Transaction data tablespace[1]: 2
Enter the Datafile Size for Transaction data tablespace (MB): 2000
Enter the Number of Datafiles for Transaction index tablespace[1]: 2
Enter the Datafile Size for Transaction index tablespace (MB): 2000
Enter the Number of Datafiles for Reference tablespace[1]: 1
Enter the Datafile Size for Reference tablespace (MB): 2000
Enter the Number of Datafiles for Interface tablespace[1]: 1
Enter the Datafile Size for Interface tablespace (MB): 1700
Enter the Number of Datafiles for Summary tablespace[1]: 2
Enter the Datafile Size for Summary tablespace (MB): 2000
Enter the Number of Datafiles for Nologging tablespace[1]:
Enter the Datafile Size for Nologging tablespace (MB): 60
Enter the Number of Datafiles for Archive tablespace[1]:
Enter the Datafile Size for Archive tablespace (MB): 1400
Enter the Number of Datafiles for Queue tablespace[1]:
Enter the Datafile Size for Queue tablespace (MB): 150
Enter the Number of Datafiles for Media tablespace[1]:
Enter the Datafile Size for Media tablespace (MB): 2000
```

Option 2: Create New Tablespaces

Select option 2 to create new tablespaces by executing the script crtts.sql that was

generated in the previous step. This script does not check the operating system limitation for the maximum size of a file.

Step 3: Generate Migration Commands

Select Step 3: Generate Migration Commands to generate migration commands for your schemas. The Generate Migration Commands menu contains the following options:

> **Caution:** You should not generate migration commands if *migration is already in progress.* OATM utility will prevent the generation of migration commands while generation of the commands or migration process is already in progress. Oracle does not recommend manually updating table FND_TS_MIG_CMDS, especially while you are generating or executing migration commands.

1. Invalid Indexes Report.
2. Generate Migration Commands for all Schemas
3. Generate Migration Commands for a List of Schemas

Option 1: Invalid Indexes Report. Please correct/drop these before generating migration commands

Select Option 1 to generate a report listing all the indexes which are invalid in the Oracle E-Business Suite schemas. This report is stored in fndinvld.txt. You must correct or drop all invalid indexes before generating migration commands for all schemas or for a given schema. This is especially relevant for context indexes. Invalid indexes on an object may cause errors during migration of the base table and invalid context indexes will not be moved.

Option 2: Generate Migration Commands for all Schemas

Select Option 2 to generate the commands for migrating the objects in all the schemas to the correct tablespace. The migration commands are stored in the table FND_TS_MIG_CMDS. You can check the generated log file fndgmcmd <timestamp>.log for errors during the generation process. A threshold object size is calculated based on the sizing data in FND_TS_SIZING to determine whether an object will be moved sequentially or in parallel. Migration commands for all objects with total blocks greater than or equal to threshold blocks are generated with the PARALLEL clause and execution mode as sequential. Migration commands for objects with total blocks less than threshold are generated with NOPARALLEL clause and execution mode as parallel. Partitioned objects are always executed sequentially regardless of their size.

Option 3: Generate Migration Commands for a List of Schemas

Select Option 3 to generate the commands for migrating the objects in a given list of comma separated schema names into the correct tablespace. The migration commands are stored in the table FND_TS_MIG_CMDS. You can check the generated log file

fndgmcmd <timestamp>.log for errors during the generation process. A threshold object size is calculated based on the sizing data in FND_TS_SIZING to determine whether an object will be moved sequentially or in parallel. Migration commands for all objects with total blocks greater than or equal to threshold blocks are generated with the PARALLEL clause and are executed sequentially. Migration commands for objects with total blocks less than threshold are generated with NOPARALLEL clause and are executed in parallel using multiple processes. Partitioned objects are always executed sequentially regardless of their size.

Example

```
Enter a comma separated list of schema names: HR,AP
```

> **Caution:** Oracle highly recommends that you back up your database after performing Step 3, which is the final preparatory step. You should have a backup copy of your database before performing the subsequent migration steps.

Phase 2: Migration Steps

Step 4: Execute Migration Commands

Select Step 4: Execute Migration Commands to execute migration commands for your schemas. The Execute Migration Commands menu contains the following options:

1. Execute Migration Commands for all Schemas
2. Execute Migration Commands for a List of Schemas
3. Migrate CTXSYS Schema

Return to the OATM menu is possible as soon as all migration processes (sequential, parallel, and java process for tables with LONG and LONG RAW columns) have started and you get the prompt – "Press Return key to continue...". Returning to the menu does not mean that migration has been completed. The migration processes are running in the background but you can return to the menu in order to monitor migration status/errors by running the migration progress report. Migration sessions are internally spawned using **nohup** and are immune to any hangup signals. Therefore VPN connection expiration, etc. should not pose any problems.

> **Caution:** You should not execute migration commands if *migration is already in progress.* The OATM utility will prevent generating migration commands as well as starting additional migration execution processes while migration process is in progress. Starting additional migration process can cause errors that can result in data corruption and will seriously impact migration process performance. Oracle does not

recommend manually updating table FND_TS_MIG_CMDS, especially while you are generating or executing migration commands.

Option 1: Execute Migration Commands for all Schemas

Select Option 1 to migrate the objects in all the schemas to the correct tablespaces. You are prompted for the number of parallel processes as input. This option executes the generated commands from FND_TS_MIG_CMDS table which generates the following events in sequence:

1. All constraints, triggers, policies are disabled first and then the queues are stopped.

2. Java program oracle.apps.fnd.tsmig.TSMigration is executed to migrate all the tables with LONG and LONG RAW columns along with their indexes. A log file for the migration of tables with LONG is generated as fndmlong<timestamp>.log.

3. A sequential process is started that executes the script fndemseq.sql to move all the objects generated with execution mode as sequential. A log file for the sequential process is generated as fndemseq<timestamp>.log.

4. Multiple processes are started to execute the SQL script fndemcmd.sql to migrate the objects generated with execution mode as parallel. A log file is generated as fndemcmd<timestamp>.log.

Example

```
Are you sure you want to migrate all schemas[N]: y
Enter the maximum number of parallel processes[4]:10

Starting the Migration process for all schemas. Please wait...

Migration processes for tables with LONG and LONG RAW columns started.
Please monitor the log file
$APPL_TOP/admin/log/fndmlong20050120230037.log
for errors

Sequential migration process started. Please monitor the log file
$APPL_TOP/admin/log/fndemseq20050120230038.log for errors

Parallel migration processes started. Please monitor the log file
$APPL_TOP/admin/log/fndemcmd20050120230048.log for errors

Press Return key to continue...
```

Option 2: Execute Migration Commands for a List of Schemas

Select Option 2 to migrate the objects in a given list of schemas to the correct tablespaces. You are prompted for a list of comma separated schema names and then for the number of parallel process as input. This option executes the generated commands from FND_TS_MIG_CMDS table which generates the following events in sequence:

1. All constraints, triggers, policies are disabled first and then the queues are stopped.

2. Java program oracle.apps.fnd.tsmig.TSMigration is executed to migrate all the tables with LONG and LONG RAW columns along with their indexes. A log file for the migration of tables with LONG is generated as fndmlong<timestamp>.log.

3. A sequential process is started that executes the script fndemseq.sql to move all the objects generated with execution mode as sequential. A log file for the sequential process is generated as fndemseq<timestamp>.log.

4. Multiple processes are started to execute the SQL script fndemcmd.sql to migrate the objects generated with execution mode as parallel. A log file is generated as fndemcmd<timestamp>.log.

Example

```
Enter a comma separated list of schema names: HR,AP

Enter the maximum number of parallel processes [4]: <10>
```

> **Note:** If your migration process terminates before it is completed, please check enqueue/dequeue status of queue - SYSTEM.TBLMIG_MESSAGEQUE by querying the following:
>
> ```
> select NAME, ENQUEUE_ENABLED, DEQUEUE_ENABLED
>          from dba_queues
>          where owner = 'SYSTEM'
>          and name = 'TBLMIG_MESSAGEQUE';
> ```

Option 3: Execute Migration Commands for CTXSYS Schema

The CTXSYS schema is not an APPS schema and in order to be included in the migration process, the CTXSYS schema has to be registered in the following manner: Login to the Forms-based version of Oracle E-Business Suite with the System Administrator Responsibility. Navigate to Security -> ORACLE -> Register and register the CTXSYS schema if it is not already registered. Then Set Privilege to "Enabled". CTXSYS schema objects are not classified by default and will be migrated to Transaction Tables and Transaction Index tablespaces, for CTXSYS tables and indexes respectively. Using Oracle Tablespace Migration Utility customizations steps, it is possible to classify CTXSYS objects to be migrated to the desired tablespace or tablespace type.

Select Option 3 to migrate CTXSYS schema objects to the correct tablespace. You are prompted for the number of parallel processes as input. This option executes the generated commands from FND_TS_MIG_CMDS table, which generates the following events in sequence:

1. All constraints, triggers, policies are disabled first and then the queues are stopped.

2. The Java program oracle.apps.fnd.tsmig.TSMigration is executed to migrate all the tables with LONG and LONG RAW columns along with their indexes. A log file for the migration of tables with LONG is generated as fndmlong.<timestamp>.log.

3. A sequential process is started that executes the script fndemseq.sql to move all the objects generated with execution mode as sequential. A log file for the sequential process is generated as fndemseq<timestamp>.log.

4. Multiple processes are started to execute the SQL script fndemcmd.sql to migrate the objects generated with execution mode as parallel. A log file is generated as fndemcmd<timestamp>.log.

Step 5: Run Migration Status Report

Select Step 5: Run Migration Status Report to run progress and error reports on the migration process. The Run Migration Status Report menu contains the following options:

1. Run Migration Status Report

2. Run Migration Error Report

Option 1: Run Migration Status Report

Select Option 1 to generate a report containing the number of successfully migrated objects, objects in error (if any) and the percentage of completion, per schema, or for all the schemas and a breakdown of the objects per object type. This option prompts you to enter the schema name and generates report fndtrep8.txt.

Example

```
Enter the Schema name[%]: <HR>
```

Option 2: Run Migration Error Report

Select Option 2 to generate a report that provides a list of objects that generated an error during the migration process that includes the error details. This option prompts you to enter the schema name and generates report fndtrep10.txt.

Example

```
Enter the Schema name[%]: <HR>
```

Phase 3: Post Migration Steps

Caution: When a standby database exists and/or you want all transactions to be recoverable on a database, tablespace, or object-wide level, it is recommended that you check that all objects have appropriate values for the logging attribute before and after the OATM migration process to ensure that all transactions are logged and can be recovered through media recovery.

Step 6: Run Post Migration Steps

Select Step 6: Run Post Migration Steps to determine which objects have not yet been migrated to OATM, to enable constraints, triggers, policies, and start queues, and to re-size old tablespaces. The Run Post Migration Steps menu contains the following options:

1. Run Audit Reports
2. Enable the Constraints, Triggers and Policies, and Start Advanced Queues
3. Re-size Old Tablespaces
4. Generate script to drop empty tablespaces

> **Note:** After running post migration steps, you must perform a complete refresh of all materialized views. This is a required manual step that is not included in the Tablespace Migration Utility menu.

Option 1: Run Audit Reports

Select Option 1 to generate a report that provides a list of objects that have not been migrated to the correct tablespace. This option prompts you to enter the schema name and generates the report, fndtrep6.txt.

Example

```
Enter the Schema name[%]: <HR>
```

Option 2: Enable the Constraints, Triggers and Policies, and Start Advanced Queues

Select Option 2 to enable all the constraints, triggers, policies and start queues, and to generate the log file, fndenabl<timestamp>.log is generated. This option prompts you to enter the schema name.

Example

```
Enter the Schema name[%]: <HR>
```

Option 3: Re-size Old Tablespaces

Select Option 3 to reduce the size of the old tablespaces. This option queries the data dictionary for all data files of the previous tablespaces to determine the level at which they can be re-sized, and generates the resize commands in a script resizdb.sql. This script is then executed to resize the data files.

Option 4: Generate Script to Drop Empty Tablespaces

Select Option 4 to generate a script to drop empty tablespaces. This option queries the data dictionary for all the previous tablespaces to determine it they still contain segments. For all the old tablespaces with no remaining segments a drop tablespace ... including contents and datafiles statement is generated in a script fndtsdrp.sql. When

executed, this script drops all empty pre-OATM tablespaces including data files.

> **Note:** Please ensure that there is no residual data in the tablespaces to be dropped prior to running the drop tablespace script to avoid irrevocable loss of data.

Step 7: Run Customization Steps

Select Step 7: Run Customization Steps if you wish to customize tablespaces, tablespace types, and object classifications as required. If you choose run customization steps you should do so before performing any of the other steps for migrating your database objects to OATM. The Run Customization Steps menu contains the following options:

1. Register new tablespace - tablespace type
2. Change name of the existing tablespace
3. Register object classification
4. Change object classification

Option 1: Register new tablespace - tablespace type

Select Option 1 to register any custom tablespace types that are not available by default with OATM. If the tablespace type or tablespace name is already registered, message will be displayed stating that it already exists

Example

```
Enter the tablespace type: CUSTOM_TABLESPACE_TYPE
Enter the tablespace name: CUSTOM_TABLESPACE

Tablespace CUSTOM_TABLESPACE registered.

Do you want to continue registering tablespaces ?[Y]:
```

Selecting the default, "Y" prompts you to enter next tablespace type/name pair. Selecting "N" returns you to the previous menu.

Option 2: Change name of the existing tablespace

Select Option 2: Change name of the existing tablespace to update the tablespace name of any default OATM tablespaces or registered custom tablespaces.

Example

```
Enter the tablespace type: CUSTOM_TABLESPACE_TYPE
Enter the new tablespace name: CUSTOM_TBLSP

Tablespace name for CUSTOM_TABLESPACE_TYPE changed to CUSTOM_TBLSP.

Do you want to continue changing tablespace names?[Y]:
```

Selecting the default, "Y" prompts you to enter next existing/new tablespace pair.

Selecting "N" returns you to the previous menu.

Option 3: Register object classification

Select Option 3: Register object classification to register new object-tablespace classifications. This is relevant only for objects such as tables that require explicit classification. These can include custom tables residing in Oracle E-Business Suite product schemas or those residing in custom schemas. If object classification for the object is already registered, a message will be displayed stating that classification already exists, and will prompt you to enter a new object name. Existing classification can be modified by selecting option 4 from the Run Customization Steps menu.

Example

```
Enter the application short name: FND
Enter the object name: FND_TABLES
Enter the tablespace type: CUSTOM_TABLESPACE_TYPE

Tablespace type CUSTOM_TABLESPACE_TYPE for object FND_TABLES registered.

Do you want to continue registering tablespace types for other
objects?[Y]:
```

Selecting the default, "Y" prompts you to register the next object classification and selecting "N" returns you to the previous menu.

Option 4: Change object classification

Select Option 4: Change object classification to change any existing object-tablespace classifications. If object classification for the object is not already registered, a message will be displayed stating that classification does not exist, and you will be prompted to enter a new object name. New object classification can be entered by selecting Option 3 from the Run Customization Steps menu.

Example

```
Enter the application short name: FND
Enter the object name: FND_LOBS
Enter the tablespace type: MEDIA

Tablespace type for object FND_LOBS changed to MEDIA.

Do you want to continue changing tablespace types for other objects?[Y]:
```

Selecting the default, "Y" prompts you to enter the next object classification modification and selecting "N" returns you to the previous menu.

Step 8: Batch Mode Execution of Migration

Note: Please note that Step 8: Run Migration in Batch Mode is an alternative step to the earlier migration steps. If you have followed OATM menu steps 1-7, you don't need to perform step 8.

Select Step 8: Run Migration in Batch Mode to execute both preparatory steps and migration commands as a single flow. The OATM menu-based design provides you

fine-grained control and maximum flexibility over how the objects will be migrated. If you need to use the OATM utility in a repeated fashion, Step 8: Run Migration in Batch Mode gives you a greater level of automated control.

If you select OATM Batch Mode, the OATM migration utility will first collect all the required information and then perform all necessary checks (including checks for the existence of all required scripts, invalid indexes, database log mode, and so on). If errors are encountered during the check stage, you are prompted to review the logs and correct any issues. Any errors during background execution will be recorded in the status table and the process will be terminated. Once errors are corrected, you can proceed with batch-mode-migration by restarting the process.

Option 1: Run Migration in Batch Mode

Menu option 8, "Run Migration in Batch Mode" has two submenus, one for the invocation of the batch-mode-migration and the other for monitoring the progress of batch mode migration. In batch mode, OATM migration utility will perform the following: a check for invalid indexes, sizing, a check for the disk space usage, a check for relevant system parameters, creation of new tablespaces, generation of migration commands, and execution of migration commands. Based on the value of the AUTO_START_MIGRATION parameter, migration will be either started automatically after preparatory steps are completed (the default); or the process will wait for your input before proceeding with migration execution (in case you want to perform some checks, back up the database, and so on). Each step will be started automatically once the previous step is finished and will be running in the background.

Option 2: Run Migration Monitor

The overall migration status report covers all OATM process phases, that is, the creation of tablespaces, generation of migration commands, and execution of migration commands. For all migration steps, status details such as the parameters passed and the name of the log files are reported, as well as the migration progress based on the total number of the migration commands, number of successfully executed commands and number of commands that failed during execution.

15

Growing a System with Oracle Real Application Clusters

Introduction to Oracle Real Application Clusters

This section gives an overview of the steps required to install Oracle E-Business Suite in an environment that uses Oracle Real Application Clusters (Oracle RAC).

For full details of using Oracle E-Business Suite Release 12 with Oracle RAC, see My Oracle Support Knowledge Document 1072636.1, *Oracle E-Business Suite Release 12 High Availability Documentation Roadmap*. This document lists release-specific documentation you should refer to according to the Oracle E-Business Suite and Oracle Database releases you are using.

Prerequisites

Several steps must be followed to prepare your system for utilizing Oracle RAC to support Oracle E-Business Suite. Key steps are as follows:

1. Install requisite Oracle Clusterware
2. Install requisite Oracle Database software
3. Configure TNS listener
4. Create ASM instances/diskgroups (optional)
5. Convert Oracle Database to Oracle RAC using rconfig
6. Perform post-conversion steps
7. Enable AutoConfig on database tier

8. Establish Applications environment for Oracle RAC

9. Configure Parallel Concurrent Processing

Migrating to Oracle RAC

Conversion of an Oracle database to utilize Oracle RAC is accomplished by running the `rconfig` utility against a sample XML file, ConvertToRAC.xml, which you modify to meet the particular requirements of your system.

In order to validate and test the settings specified for converting to Oracle RAC with rconfig, `rconfig` should be run in "Verify only" mode prior to running the actual conversion. This will perform a test run, in which `rconfig` will validate parameter settings, and report any issues that need to be resolved before the actual conversion is undertaken.

Establishing the Oracle E-Business Suite Environment for Oracle RAC

On the Oracle E-Business Suite side, a number of steps are needed to prepare the environment for Oracle RAC. Follow the detailed instructions in the relevant knowledge document under My Oracle Support Knowledge Document Note 745759.1, *Oracle E-Business Suite and Oracle Real Application Clusters Documentation Roadmap*.

In summary, you need to:

1. Run AutoConfig on the Applications database tier.

2. Check the tnsnames.ora and listener.ora files in the $INST_TOP/ora/10.1.2/network/admin and $INST_TOP/ora/10.1.3/network/admin/<context_name>.

3. Ensure that the correct TNS aliases have been generated for load balance and failover.

4. Ensure that all the aliases are defined using the virtual hostnames.

5. Verify the dbc file located at $FND_SECURE. Ensure that the parameter APPS_JDBC_URL is configured with all instances in the environment, and load_balance is set to YES.

Configuring Parallel Concurrent Processing with Oracle RAC

To take full advantage of Parallel Concurrent Processing (PCP), you will need to have more than one Concurrent Processing node in your environment. If you do not have more than one node, refer to My Oracle Support for details of cloning the application tier.

The key steps in configuring concurrent parallel processing are as follows:

1. Configure PCP on all concurrent processing nodes.
2. Set up transaction managers.
3. Set up load balancing on concurrent processing nodes and run AutoConfig.

Related Topics

Overview of Parallel Concurrent Processing, page 4-143

Managing Parallel Concurrent Processing, page 4-147

16

Integration of Oracle E-Business Suite with Other Products

Integration of Oracle E-Business Suite with Other Products

Oracle E-Business Suite can be integrated with other Oracle products. This chapter highlights some of the key integration points.

Linking to Application Development Framework Applications

You can link to an Application Development Framework (ADF) application from the Oracle E-Business Suite home page, regardless of how or if the ADF application integrates with Oracle E-Business Suite data on the back end. For more information on using ADF applications with Oracle E-Business Suite, see *E-Business Suite Application Development using Oracle Application Framework (OAF) and Application Development Framework (ADF),* My Oracle Support Knowledge Document 563047.1.

Integrating Oracle E-Business Suite with Oracle Access Manager using Oracle E-Business Suite AccessGate

Oracle E-Business Suite supports authentication of application users using Oracle Access Manager through the Oracle E-Business Suite AccessGate application. Oracle E-Business Suite AccessGate is a Java EE application that is deployed to a WebLogic Server instance, and works in conjunction with Oracle Access Manager and Oracle Internet Directory (OID) to enable single sign-on capabilities for your entire enterprise. This application is currently targeted primarily at customers who have already deployed Oracle Access Manager in other organizations within the company, and who wish to extend the authentication capabilities of Oracle Access Manager to Oracle E-Business Suite instances. For information about the recommended integration for your version of Oracle E-Business Suite, see: *Overview of Single Sign-On Integration Options for Oracle E-Business Suite*, My Oracle Support Knowledge Document 1388152.1.

See also: Overview of Single Sign-On Integration, *Oracle E-Business Suite Security Guide*.

Using AppsDataSource, Java Authentication and Authorization Service, and Utilities for Oracle E-Business Suite

Oracle Application Object Library added new standard Java datasource and Java Authentication and Authorization Service (JAAS) features, plus extended error logging routines, to Oracle E-Business Suite in Patch 8571001. These features are meant for use with Java EE programs deployed in application servers on external nodes; that is, nodes other than those where Oracle E-Business Suite middle tier is installed. These are lightweight implementations that can be used on an external application server without needing to install an entire Oracle E-Business Suite instance on the application server machine. For more information, see: *Oracle E-Business Suite Software Development Kit for Java (includes AppsDataSource, Java Authentication and Authorization Service, session management) Readme - Patch 13882058,* My Oracle Support Knowledge Document 974949.1.

Using Oracle GoldenGate with Oracle E-Business Suite for Operational Reporting

You can use Oracle GoldenGate with Oracle E-Business Suite for operational reporting, in which selected tables are replicated from the primary Oracle E-Business Suite database to a secondary (duplicate) instance. This enables read-intensive, custom reports to be run without using resources on the primary Oracle E-Business Suite database. For more information, see: *Deploying Oracle GoldenGate to Achieve Operational Reporting for Oracle E-Business Suite,* My Oracle Support Knowledge Document 1112325.1.

Autovue Document Print Service with File-Type Attachments

Attachments are fully integrated with Oracle's AutoVue Document Print Service, allowing users to print their file-type attachments. This integration also allows an enterprise to support the printing of attachments of different file types from a server where the Auto Document Print Service is configured, without having to install or configure additional software to view and print files in client/server machines.

> **Note:** Oracle E-Business Suite Release 12.2 integration with AutoVue Document Print Service is certified with Oracle AutoVue Release 20.1.1.

In the Attachments feature user interface, a Print icon is available in the Attachments list for file-type attachments. Users can select the Print icon to print the file. Users can use this feature only when the file-type attachment is committed to the database. If the Automatic Save property of an attachment region item is set to False, a user must explicitly commit the transaction of any newly-added file attachment first.

This feature is enabled through the profile option FND Attachment AutoVue Server.

For more information, see the *Oracle Application Framework Developer's Guide* available from My Oracle Support Document 1315485.1.

17

Cloning Oracle E-Business Suite with Rapid Clone

Introduction to Cloning

This section provides an overview of the process used to clone (copy) an Oracle E-Business Suite Release 12.2 system.

> **Note:** An Oracle E-Business Suite system can only be cloned to the same, or a higher, major release of a platform (operating system). For example, you can clone a system running on Oracle Solaris 10 to run on Oracle Solaris 11, but it is not supported to clone a system running on Solaris 11 to run on Solaris 10.

> **Note:** When you are performing cloning, ensure that you specify the actual locations for the directories involved, so that AD utilities can properly identify the directories afterward. Do not specify the directory locations as symbolic links.

Terminology

Cloning is the process of creating a copy of an existing Oracle E-Business Suite system. There are various scenarios that require cloning an Oracle E-Business Suite system:

- **Standard cloning** - Copying an existing Oracle E-Business Suite system, for example, to test updates against a duplicate of a production system.

  > **Note:** Be aware that standard cloning is different from file system cloning. Standard cloning is creating a copy of an Oracle E-Business Suite system using adclone. For example, cloning a complete Oracle

E-Business Suite system to create a test copy from your production environment. In contrast, file system cloning is copying the run file system to the patch file system in Online Patching, and can only be undertaken with the `adop phase=fs_clone` command.

- **System scale-up** - Adding new machines to an Oracle E-Business Suite system to provide capacity for processing increased workloads.
- **System transformations** - Altering system data or file systems, including actions such as platform migration, provisioning of high availability architectures, and data scrambling (or the preferred alternative of data masking, as described in My Oracle Support Knowledge Document 1437485.1).
- **Patching and upgrading** - Delivering new versions of Oracle E-Business Suite components, and providing a mechanism to create rolling environments that minimize downtimes.

An important principle in Oracle E-Business Suite cloning is that the *system* is cloned, rather than the *topology*. Producing an exact copy of the patch level and data is much more important than creating an exact copy of topology, as a cloned system must be able to provide the same output to the end user as the Source System. However, while a cloned system need not have the full topology of its source, it must have available to it all the topology components that are available to the source.

Important: The cloning methodology described in this chapter requires full AutoConfig compliance.

Conventions used in this document include the following:

Term or Usage	Meaning or Action
Source System	Oracle E-Business Suite system being cloned.
Target System	Oracle E-Business Suite system being created as a copy of the Source.
APPLMGR	User that owns the Application Tier File System (APPL_TOP and Application Tier technology stack).
ORACLE	User that owns the Database Tier File System (RDBMS ORACLE_HOME and database files).

Term or Usage	Meaning or Action
CONTEXT_NAME	The CONTEXT_NAME variable refers to the name of the Applications context file. By default, CONTEXT_NAME is <SID>_<HOSTNAME>.
`Monospace Text`	Represents command line text. Type this command exactly as shown.
< >	Text enclosed in brackets represents a variable. Substitute a value for the variable text. Do not type the brackets.

Prerequisite Tasks

Important: Before cloning a system, be sure to complete any in-progress online patching cycle all the way through to cleanup. Then run `fs_clone` to synchronize with the other file system, so that synchronization does not have to be performed in the next patching cycle. For more information, refer to *Oracle E-Business Suite Maintenance Guide*.

First, prepare the Source System by applying any required patches and running AutoConfig.

1. **Verify disk space requirements on Source System**

   Before cloning to a new server, ensure the Source System has enough free disk space. Oracle Fusion Middleware cloning tools require 6GB in /tmp and 6GB under $COMMON_TOP.

2. **Verify OS requirements on Target System**

   Before cloning to a new server, ensure the Target System meets all the requirements for Oracle E-Business Suite Release 12.2 as stated in the Oracle E-Business Suite Release Notes and the Oracle E-Business Suite Installation and Upgrade Notes for each platform.

   Note: For Microsoft Windows, Rapid Clone is not currently certified for use from Domain User Accounts.

3. **Verify Source and Target System software components and versions**

In addition to the Oracle E-Business Suite software requirements (see *Oracle E-Business Suite 12.2 Installation Guide: Using Rapid Install*); the following software component versions must exist on the Source or Target nodes, as applicable. The 'Required Location' column indicates the node where the software component must reside.

Software Requirements

Software Component	Minimum Version	Required Location	Comments
Zip	2.3 (or higher)	All Source System nodes	Download from InfoZip [http://www.info-zip.org/]. Zip must be in your $PATH. If using files larger than 2GB, you should use InfoZip ZIP 3.0 or higher.
Unzip	5.52 (or higher)	All Source System nodes	Download from InfoZip [http://www.info-zip.org/]. Unzip must be in your $PATH. If using files larger than 2GB, you should use InfoZip UNZIP 5.52 or higher.

Software Component	Minimum Version	Required Location	Comments
Operating System utilities	N/A	All Target System nodes	The required operating system utilities for your platform must be in your $PATH when running `adcfgclone.pl`. For example, `make`, `ld`, and `ar` are required utilities for UNIX. Refer to *Oracle E-Business Suite Installation Guide: Using Rapid Install* [http://docs.oracle.com/cd/B53825_02/current/acrobat/121oaig.pdf].
Perl	5.x	All Target System nodes	Use the version of Perl shipped with Oracle Fusion Middleware 11g and Oracle Database 11g, or download from Perl.com [http://www.perl.com/download.csp]. Perl must be in your $PATH, and $PERL5LIB must be set correctly before cloning.

4. **Apply the latest AD patch**

   Refer to My Oracle Support [https://support.oracle.com] to obtain the latest AD patch, if any are available.

5. **Apply the latest AutoConfig template patch**

   Update the Oracle E-Business Suite file system with the latest AutoConfig template files by applying the TXK AutoConfig Template rollup patch to all Application Tier server nodes. For details of the latest AutoConfig Template rollup patch, refer to

Patching AutoConfig, page 3-27.

6. **Apply the latest Rapid Clone patches**

   Update the Oracle E-Business Suite file system with the latest Rapid Clone files by applying the latest Rapid Clone patch to all Application Tier server nodes. Refer to My Oracle Support Knowledge Document 1383621.1, *Cloning Oracle E-Business Suite Release 12.2 with Rapid Clone*, for details of the latest Rapid Clone patches and special considerations for known issues.

   > **Warning:** Failing to use the latest code may jeopardize the success of the clone. If new AD, Autoconfig or Rapid Clone updates are applied to the system, steps 4, 5, 6, 7, and 8 must be executed again in order to apply the new files to the Database Node.

7. **Run AutoConfig on the Application Tier**

   Follow the steps in the section "Run AutoConfig on the Application Tier" in Using AutoConfig Tools for System Configuration, page 3-9 to run AutoConfig on all Application Tier nodes.

8. **Synchronize appsutil on the Database Tier nodes**

   To copy AutoConfig and Rapid Clone files to each Database Node with the admkappsutil.pl utility, follow the steps under the "Copy AutoConfig to the RDBMS ORACLE_HOME" section in Patching AutoConfig, page 3-27.

9. **Run AutoConfig on the Database Tier**

   Follow the steps under the "Run AutoConfig on the Database Tier" section inUsing AutoConfig Tools for System Configuration, page 3-9 to run AutoConfig on the Database Tier nodes.

10. **Maintain snapshot information**

    Log in to each Application Tier node as the APPLMGR user, and run "Update current view snapshot" in AD Administration. Refer to *Oracle E-Business Suite Maintenance Guide* for more information.

Standard Cloning Tasks

Use Rapid Clone to create template files for cloning on the Source System. After copying the Source System to the Target, Rapid Clone updates these templates to include the new Target System configuration settings.

> **Note:** Rapid Clone never changes the Source System configuration.

The cloning process consists of the following three phases, each of which is made up of several logical sections and their steps.

> **Warning:** Before following the steps on this Section, review the "Special Considerations" section in My Oracle Support Knowledge Document 1383621.1, *Cloning Oracle E-Business Suite Release 12.2 with Rapid Clone*.

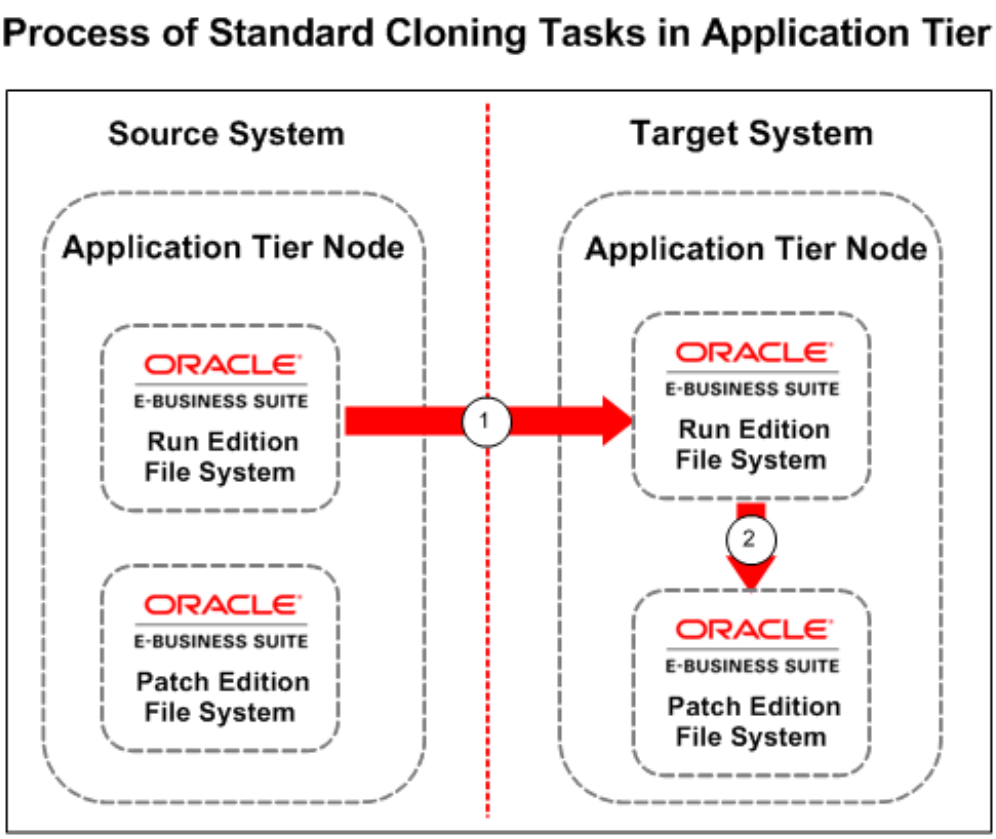
1. Prepare the Source System for Database Tier and Application Tier
2. Copy both Database Tier and Application Tier nodes from the Source System to Target System
3. Configure the Target System for both Database Tier and Application Tier

High Level Standard Cloning Process for Application Tier

In addition to cloning the database tier node, standard cloning process for Application Tier basically involves the following two key processes:

1. Copy Application Tier node from the Source "Run Edition File System" to the Target "Run Edition File System".
2. Copy the Application Tier node from the Target "Run Edition File System" to the Target "Patch Edition File System".

This high level cloning process specific to Application Tier can be illustrated in the following diagram:



Standard Cloning Steps

Use the following steps to perform standard cloning tasks:

1. **Prepare the Source System**

   Execute the following commands to prepare the Source System for cloning:

   > **Note:** Ensure all the processes in the Source System are up before proceeding with the following two steps.

   1. **Prepare the Source System Database Tier for cloning**

      Log on to the Source System as the ORACLE user, and run the following commands:

      ```
      $ cd <RDBMS ORACLE_HOME>/appsutil/scripts/<CONTEXT_NAME>
      $ perl adpreclone.pl dbTier
      ```

   2. **Prepare the Source System Application Tier for cloning**

      Log on to the "Run Edition File System" in the Source System as the APPLMGR user, and run the following commands on each Node that contains an

APPL_TOP:

```
$ cd <INST_TOP>/admin/scripts
$ perl adpreclone.pl appsTier
```

You can use the following command to confirm that the environment variable FILE_EDITION points to the Run Edition File System:

```
$ echo $FILE_EDITION
```

It should return the value:

```
run
```

> **Note:** If you apply a new Rapid Clone or AutoConfig update to the system, you must execute adpreclone.pl again on the Database Tier and Applications Tier to apply the new files into the clone directory structures to be used during the cloning configuration stage. Furthermore, if you patch Oracle Fusion Middleware or make configuration changes to the Oracle E-Business Suite Weblogic Domain, you must execute adpreclone.pl again on the Applications Tier to rebuild the Oracle Fusion Middleware home.

As of Release 12.2, the adpreclone.pl process on the Application Tier creates a complete compressed archive of the Oracle Fusion Middleware and its components as follows:

- A compressed archive of the Oracle WebLogic Server home, Oracle Web Tier Utilities home, Oracle Common Utilities home and the Oracle E-Business Suite home:

  `<COMMON_TOP>/clone/FMW/FMW_Home.jar`

- A compressed archive of the Oracle E-Business Suite Weblogic domain:

  `<COMMON_TOP>/clone/FMW/WLS/EBSdomain.jar`

- The Oracle E-Business Suite Weblogic domain's configuration template:

  `<COMMON_TOP>/clone/FMW/WLS/plan/moveplan.xml`

- A compressed archive of the Oracle Web Tier/Oracle HTTP Server configuration instance:

  `<COMMON_TOP>/clone/FMW/OHS/ohsarchive.jar`

- The Oracle HTTP Server configuration instance's configuration template:

  `<COMMON_TOP>/clone/FMW/OHS/moveplan.xml`

3. **Shut down the Application processes (Windows platform Only)**

Log on to "Run Edition File System" in the Source System as the APPLMGR user, and shut down all the Application processes.

2. **Copy the Source System to the Target System**

   Copy the Application Tier file system from the Source node to the Target node by executing the following steps in the order listed. Ensure the Application Tier files copied to the Target System are owned by the Target APPLMGR user, and that the Database node files are owned by the Target ORACLE user.

   > **Note:** In the copying tasks below, UNIX users should ensure that the symbolic links (soft links) are preserved when copying. On most UNIX platforms, you can accomplish this with the `cp -RH` command. Consult the UNIX `man` page for the `cp` command to check the parameters available on your platform.
   >
   > For example:
   >
   > ```
   > cd /Target_dest_dir/db
   > cp -RH /Source_dir/db/
   > ```
   >
   > Alternatively, you can use the `tar` command to compress the directories into a temporary staging area. UNIX users should ensure that the symbolic links (soft links) are preserved when compressing. On most UNIX platforms, this is the default behavior of the tar command. Consult the UNIX `man` page for the `tar` command to check the parameters available on your platform.
   >
   > Additionally, verify the permissions of the executables under ORACLE_HOME/bin that can potentially be owned by root, such as nmo, nmhs, nmb, and so on.

   1. **Copy the Database Node file system**

      Log on to the Source System Database node as the ORACLE user, and then:

      1. Perform a normal shutdown of the Source System database
      2. Copy the database (.dbf) files from the Source System to the Target System
      3. Copy the Source System database ORACLE_HOME to the Target System
      4. Start the Source Applications system database and Application Tier processes

   2. **Copy the Application Tier file system from the Source "Run Edition File System" to the Target "Run Edition File System"**

      Log on to "Run Edition File System" in the Source System Application Tier

nodes as the APPLMGR user.

Copy the following Application Tier directories from the Source Node to the Target "Run Edition File System" Application Tier Node:

- <APPL_TOP>
- <COMMON_TOP>
- <OracleAS Tools 10.1.2 ORACLE_HOME>

The same Operating System user must own both the Run Edition and Patch Edition File Systems.

> **Warning:** In Release 12.2, you can set the base directory to any desired location. However, the subdirectory structure cannot be changed because of dependencies on both the WLS domain and the dual file system required for online patching. Also, the base directory must be the same across all nodes in multi-node configurations.

Dual File System Directory Structure

In Release 12.2, the following directory structure exists to allocate the Run Edition and Patch Edition File Systems:

> **Note:** Please note that <s_base> and <sid> are user-defined values.

- <s_base>/<sid>/fs1 (for example, /u01/122/prod/fs1)
- <s_base>/<sid>/fs2 (for example, /u01/122/prod/fs2)

Two environment variables, $RUN_BASE and $PATCH_BASE, store these locations. The function (RUN or PATCH) of these two file systems is not static, and their values switch every time when a cutover phase is complete.

As Clone would create a replica of the Source Node, if the Source "Run Edition File System" is the first file system (fs1), the Target "Run Edition File System" would also be the first file system (fs1). Similarly, if the Source "Run Edition File System" is the second file system (fs2), then the Target "Run Edition File System" would also be the second file system (fs2). Therefore, when you perform a cloning task, you always clone and copy the Source "Run Edition File System" to create the Target "Run Edition File System", but the directory location of the "Run Edition File System" can be pointing either to <s_base>/<sid>/fs1 or <s_base>/<sid>/fs2 based on the Source "Run Edition File System" base directory.

When copying the files, use the values of $RUN_BASE and $PATCH_BASE

variables to determine if the "Run Edition File System" should be copied to fs1 or fs2. For example,

- The Source "Run Edition File System" has the following values:
  - `$RUN_BASE=/u01/122/prod/fs2`
  - `$PATCH_BASE =/u01/122/prod/fs1`
- The Target `<s_base>` directory will be `/d05/test`.

In this situation, copy the Source "Run Edition File System" into the Target `/d05/test/fs2` directory to initially act as "Run Edition File System" and into the Target `/d05/test/fs1` to act as "Patch Edition File System".

3. **Configure the Target System**

   Run the following commands to configure the Target System. You will be prompted for specific Target System values such as SID, paths, and ports.

   1. **Configure the Target System database server**

      Log on to the Target System as the ORACLE user and enter the following commands:

      ```
      $ cd <RDBMS ORACLE_HOME>/appsutil/clone/bin
      $ perl adcfgclone.pl dbTier
      ```

   2. **Configure the Target System Application Tier server nodes**

      1. Log on to the "Run Edition File System" in the Target System as the APPLMGR user and enter the following commands:

         ```
         $ cd <COMMON_TOP>/clone/bin
         $ perl adcfgclone.pl appsTier
         ```

         At the prompt, "Target System Base Directory", enter the location of the base directory. For example: `/u02/r122`.

      2. Run adpreclone.pl on the "Run Edition File System" in the Target System.

      3. Copy Oracle E-Business Suite application directories from the "Run Edition File System" to the "Patch Edition File System" in the Target System.

         Note: Make sure you pay attention to symbolic links under <OracleAS Tools 10.1.2 ORACLE_HOME> and <COMMON_TOP> directories when copying files.

      4. Log on to the "Patch Edition File System" in the Target System as the

APPLMGR user and enter the following commands:

```
$ cd <COMMON_TOP>/clone/bin
```

```
$ perl adcfgclone.pl appsTier
```

The "Patch Edition File System" must be aware of the location of the "Run Edition File System" that will be configured in the next step. At the prompt, "Location of Run System Context File", enter the absolute path to the Context File for the "Run Edition File System" that was created in the previous step. For example: `/u02/r122/fs1/inst/apps/PROD_atgserver/appl/admin/PROD_atgserver.xml`.

Warning: If adcfgclone.pl fails during the configuration portion, review the following before any new cloning attempt:

- Complete Oracle Fusion Middleware Home
- Complete INST_TOP directory
- Verify if Oracle Inventory contains the following Oracle Home entries:
  - <FMW_HOME>/oracle_common
  - <FMW_HOME>/webtier
  - <FMW_HOME>/Oracle_EBS-app1

If any of the above Oracle Home entries is already registered in Oracle Inventory, you can execute the following command to unregister or detach that Oracle Home:

```
$ ./runInstaller -detachhome
ORACLE_HOME=<Oracle Home Location>
```

For Example:

```
$ cd
/u02/r122/fs1/FMW_Home/oracle_common/oui/bi
n
```

```
$ ./runInstaller -detachhome
```

```
ORACLE_HOME=/s0/r122/at1/FMW_Home/oracle_co
mmon
```

In addition to the unregistration process, ensure to delete the FMW_HOME directory structure.

Finishing Tasks

This section lists tasks that may be necessary depending on your implementation and intended use of the cloned system.

1. **Update profile options**

   Rapid Clone updates only site level profile options. You must manually update any other profile options set to instance specific values.

2. **Update printer settings**

   If the new cloned system needs to utilize different printers, update the Target System with the new printer settings now.

3. **Update Workflow configuration settings**

   Cloning an Oracle E-Business Suite instance does not update the host and instance-specific information used by Oracle Workflow. Review the table and column names listed in the following table to check for any instance-specific data in the Oracle Workflow configuration of the Target System.

Workflow Configuration Settings

Table Name	Column Name	Column Value Details
WF_NOTIFICATION_ATTRIBUTES	TEXT_VALUE	If value starts with "http://<old web host>:", update to new web host.
WF_ITEM_ATTRIBUTE_VALUES	TEXT_VALUE	If value starts with "http://<old web host>:", update to new web host.
WF_SYSTEMS	GUID	Using the Workflow Administrator Web Applications responsibility, create a new system defined as the new global database name.
WF_SYSTEMS	NAME	Replace value with the database global name.

Table Name	Column Name	Column Value Details
WF_AGENTS	ADDRESS	Update database link with the new database global name.
FND_FORM_FUNCTIONS	WEB_HOST_NAME	Update with the new web host name.
FND_FORM_FUNCTIONS	WEB_AGENT_NAME	Update to point at the new PL/SQL listener name.
FND_CONCURRENT_REQUESTS	LOGFILE_NAME	Update with the correct path to the log file directory.
FND_CONCURRENT_REQUESTS	OUTFILE_NAME	Update with the new directory path on the Target System.

4. **Verify the APPLCSF variable setting**

   Source the APPS environment and confirm that the variable APPLCSF (identifying the top-level directory for concurrent manager log and output files) points to a suitable directory. To modify, change the value of the s_applcsf variable in the context file and then run AutoConfig.

5. **Update the SESSION_COOKIE_DOMAIN value in ICX_PARAMETERS**

   If the Target System is in a different domain name than the Source System and SESSION_COOKIE_DOMAIN is not null in the Source System, then update the value of SESSION_COOKIE_DOMAIN to reflect the new domain name.

6. **Re-Implement SSL and SSO configuration**

   If the Source System is Secure Sockets Layer (SSL) or Oracle Access Manager-enabled, and you want the Target to be SSL or Oracle Access Manager-enabled, then reconfigure the Target according to the SSL or Oracle Access Manager documentation. Otherwise, if you do not want the Target to be SSL or Oracle Access Manager-enabled, then follow the same SSL or Oracle Access Manager documentation to undo the SSL or Oracle Access Manager setup.

Advanced Cloning Options

This section describes various advanced cloning procedures required under certain circumstances.

Option 1: Refreshing a Target System

You may need to refresh the Target System periodically to synchronize it with changes to the Source System.

> **Note:** Back up the Target context file on the Target System before refreshing the Database or Application tiers.

To refresh the Target System, perform the following steps as described in previous sections:

> **Note:** The tasks in this section must be executed on both Run Edition and Patch Edition File Systems.

1. Prepare the Source System, page 17-8.
2. Copy the Source System to the Target System, page 17-10.
    1. If the Application Tier File System in the APPL_TOP, COMMON_TOP, or technology stack needs to be refreshed, copy the portion of the Application Tier File System that has been updated.
    2. If the RDBMS ORACLE_HOME or the database needs to be refreshed, copy the Database Node File System. If refreshing the database, be sure to refresh the ORACLE_HOME at the same time.
3. Configure the Target System, page 17-12.

    Specify the existing Target System context file when running adcfgclone.pl commands.

    > **Note:** Ensure that you review additional customized WLS and OHS configuration parameters from the Fusion Middleware Console in the Target System.

    1. Configure the Target System database server by logging in to the Target System as the ORACLE user and entering the following commands to configure and start the database:

        ```
        $cd <RDBMS ORACLE_HOME>/appsutil/clone/bin

        perl adcfgclone.pl dbTier <Database Target context file>
        ```

        Where Database Target context file is `<RDBMS ORACLE_HOME>/appsutil/<Target CONTEXT_NAME>.xml`.

    2. Configure the Target System Application Tier server nodes by logging in to the

Target System as the APPLMGR user and entering the following commands:

```
$ cd <COMMON_TOP>/clone/bin

$ perl adcfgclone.pl appsTier <APPL_TOP Target context
file>
```

Where APPL_TOP Target context file is
<INST_TOP>/appl/admin/<Target CONTEXT_NAME>.xml.

4. If any of the above ORACLE_HOME entries is already registered in Oracle Inventory, execute the following command to unregister or detach that ORACLE_HOME:

   ```
   $ ./runInstaller -detachhome ORACLE_HOME=<Oracle Home
   Location>
   ```

   For example:

   ```
   $ cd /u02/r122/fs1/FMW_Home/oracle_common/oui/bin

   $ ./runInstaller -detachhome
   ORACLE_HOME=/s0/r122/at1/FMW_Home/oracle_common
   ```

5. Perform the standard finishing tasks, page 17-14.

Option 2: Cloning a Multi-Node System

Overview

This procedure allows the Source System or Target System to be a multi-node system. The essence of cloning a multi-node system involves the following major steps:

1. Clone the Database Tier node.

2. Clone the primary Application Tier node from the Source "Run Edition File System" to the Target "Run Edition File System".

3. Clone the primary Application Tier node from the Target "Run Edition File System" to the Target "Patch Edition File System".

4. Add additional Application Tier nodes in the secondary Application Tier node "Run Edition File System".

   This is achieved by cloning the primary Application Tier node from the Target "Run Edition File System" to the secondary Application Tier node in the Target "Run Edition File System".

5. Add additional Application Tier nodes in the secondary Application Tier node "Patch Edition File System".

   This is achieved by cloning the primary Application Tier node from the Target "Patch Edition File System" to the secondary Application Tier node in the Target

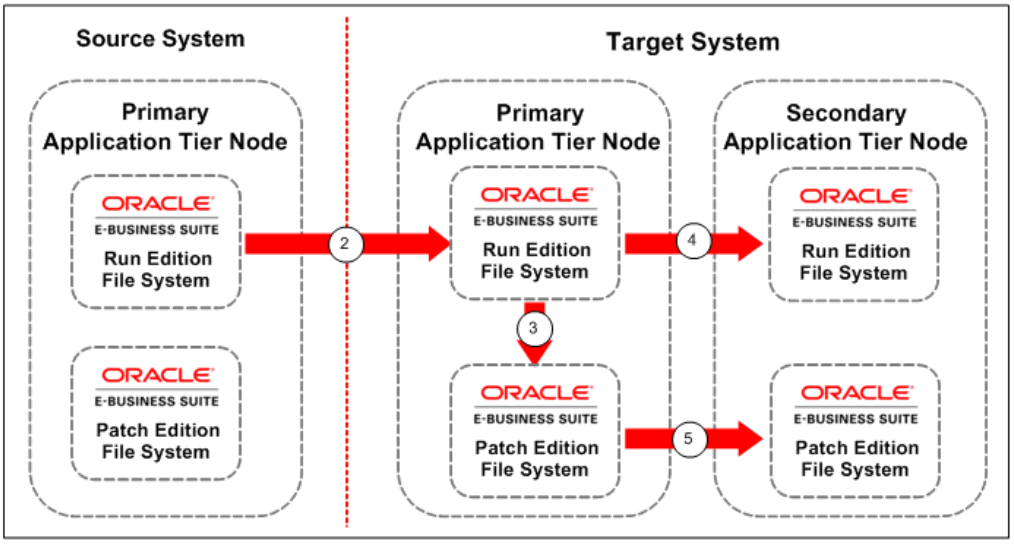
"Patch Edition File System".

> **Note:** Steps 4 and 5 can be repeated until your desired nodes are all added to the Target System.

> **Warning:** Make sure you set Secure Shell (ssh) on all your Application Tier nodes if adop utility is used to handle multiple nodes. For information on how to set up Secure Shell, see the Patching Procedures chapter, *Oracle E-Business Suite Maintenance Guide*.

The high level process of cloning a multi-node system in Application Tier as described above from step 2 to step 5 can be illustrated in the following diagram:

Process of Cloning a Multi-Node System in Application Tier



Instructions on Cloning a Multi-Node System

Detailed instructions on how to perform each task described earlier are explained as follows:

1. Perform the standard Prerequisite Tasks, page 17-3.

   Carry out these steps on all Source and Target nodes.

2. Perform a full clone (including the prepare, copy and configure steps) of the Database Tier node. Refer to Standard Cloning Tasks, page 17-6.

3. **Clone the Primary Application Tier Node from the Source "Run Edition File System" to the Target "Run Edition File System"**

1. Log on to the primary Application Tier node in the Source "Run Edition File System" and run the following command:

   ```
   $ perl adpreclone.pl appsTier
   ```

2. Choose the Target machine. Target System should be on a new host other than any of the source nodes

3. Copy Oracle E-Business Suite applications from the Source "Run Edition File System" fs1 to the Target "Run Edition File System" fs1.

   - If the Source "Run Edition File System" is in fs1, then the Target "Run Edition File System" should also be configured under fs1 If the Source "Run Edition File System" is in fs2, then the Target "Run Edition File System" should also be configured under fs2.

   - Execute `echo $RUN_BASE` with any of the Source file system sourced to determine which file system (fs1 or fs2) is the "Run Edition File System" current associated.

4. Configure the Target "Run Edition File System" by running:

   ```
   $ perl adcfgclone.pl appsTier
   ```

   - Ensure that Port Pool provided for the Target fs1 is different from that of any of the Source nodes.

   - The Weblogic domain of the Target primary node would be configured with all the Source managed servers and machines.

     Perform the following Post Cloning steps for the Target "Run Edition File System" to ensure that managed servers and machines of the Source secondary nodes are deleted from the Target primary node:

     1. Log in to the Oracle WebLogic Server Administration Console.

     2. Under the Domain Structure section, select **Environment > Servers**.

        Click on the **Lock & Edit** button in the 'Change Center' panel and delete the managed servers whose machine is configured as `<hostname>_<offset>`. For example, managed servers corresponding to your source secondary nodes. Click on the **Activate Changes** button in the 'Change Center' panel to activate the changes after deletion.

     3. Under the Domain Structure section, select **Environment > Machines**.

        Click on the **Lock & Edit** button in the 'Change Center' panel and delete the machine whose machine is configured as `<hostname>_<offset>`. For example, machines corresponding to your source secondary nodes.

Click on the **Activate Changes** button in the 'Change Center' panel to activate the changes after deletion.

After the deletion, you should have only one machine associated with Target "Run Edition File System".

4. **Clone the Primary Application Tier Node from the Target "Run Edition File System" to the Target "Patch Edition File System"**

    1. Run the following command on the Target "Run Edition File System" which was configured in the previous step:

        ```
        $ perl adpreclone.pl appsTier
        ```

    2. Copy Oracle E-Business Suite applications from the Target "Run Edition File System" to the Target "Patch Edition File System".

    3. Run the following command to configure the Target "Patch Edition File System":

        ```
        $ perl adcfgclone.pl appsTier
        ```

    4. Ensure that Port Pool provided for Target "Patch Edition File System" is different from that of Target "Run Edition File System".

5. **Add Additional Application Tier Nodes in the Non-Shared Secondary Application Tier Node "Run Edition File System" and Then in the Non-Shared Secondary Application Tier Node "Patch Edition File System"**

    Follow the instructions in the next section, "Option 3: Adding a New Node to an Existing System," to add a new node in the "Run Edition File System" and then in the "Patch Edition File System" of the Target system.

6. Repeat step 5 to add the desired number of secondary nodes.

    Important: If you are using a load balancer to distribute processing across all the Application Tier nodes, each node should have the value of the `s_webentryhost` context variable set to be that of the load balancer's hostname.

Option 3: Adding a New Node to an Existing System

You can use Rapid Clone to clone a node and add it to the existing Applications system, a process also known as scale up or scale out. The new node may run the same services as the Source node or different services.

Note: The tasks in this section must be executed on both the Run and Patch edition File Systems. For more details, follow the instructions for the Application Tier in the Standard Cloning Tasks, page 17-6.

Note: These steps also apply to shared secondary Application Tier nodes. Refer to My Oracle Support Document 1375769.1, "Sharing The Application Tier File System in Oracle E-Business Suite Release 12.2," for detailed instructions on adding nodes on a shared file system.

Before adding a new node, you should update the tcp.invited_nodes parameter of sqlnet.ora to include the host.domain of the new node being added.

1. Run adpreclone.pl on both the Run and Patch Edition File Systems in the primary Application tier node of the Target system.

2. Copy both Run and Patch Edition File Systems to the target secondary node.
   - The secondary node must be on a different host.
   - If the Target primary node's "Run Edition File System" is in fs1, then the secondary node's "Run Edition File System" should also be in fs1. If the Target primary node's "Run Edition File System" is in fs2, then the secondary node's "Run Edition File System" should also be in fs2

3. Configure both Run and Patch Edition File Systems in the Target node.
   1. Log in as APPLMGR to the "Run Edition File System" in the Target node and enter the following commands:

      Warning: Ensure that you enter the correct password for "Weblogic AdminServer". Rapid Clone will not be able to validate the password at this stage but it will fail later. Refer to *Cloning Oracle E-Business Suite Release 12.2 with Rapid Clone*, My Oracle Support Knowledge Document 1383621.1 for details.

      Warning: Before executing this step, ensure the Admin Server of the Target System primary node is running. This is required for adcfgclone.pl to properly reregister the new node on the Target System.

```
$ cd <COMMON_TOP>/clone/bin
$ perl adcfgclone.pl appsTier
```

Where

"Are you adding a node (yes/no)" is `yes`

"Do you want to startup the Application Services for mult35? (y/n) [y]:" is `y`

2. Log in as APPLMGR to the "Patch Edition File System" in the Target node and enter the following commands:

   Warning: Ensure that you enter the correct password for "Weblogic AdminServer". Rapid Clone will not be able to validate the password at this stage but it will fail later. Refer to *Cloning Oracle E-Business Suite Release 12.2 with Rapid Clone*, My Oracle Support Knowledge Document 1383621.1 for details.

   Warning: Before executing this step, ensure the Admin Server of the Target System primary node is running. This is required for adcfgclone.pl to properly reregister the new node on the Target System.

   ```
   $ cd <COMMON_TOP>/clone/bin
   $ perl adcfgclone.pl appsTier
   ```

   Where:

   "Are you adding a node (yes/no)" is `yes`

4. Register the new topology from the newly added Application Tier node.

   1. Log in as APPLMGR to the "Run Edition File System" in the primary Application tier node and on all other configured secondary nodes of the Target system and execute the following commands:

      1. Run AutoConfig:

         ```
         $ perl $AD_TOP/bin/adconfig.pl
         -contextfile=$CONTEXT_FILE -product=ad
         ```

      2. Register the managed servers of the newly added Application Tier node with the Oracle HTTP Server:

         ```
         $ perl $FND_TOP/patch/115/bin/txkSetAppsConf.pl
         -ctxfile=$CONTEXT_FILE \
         -outfile=$APPLRGF/TXK/txkSetAppsConf.log
         ```

   2. Log in as APPLMGR to the "Patch Edition File System" in the primary Application tier node and all other configured secondary nodes of the Target system and execute the following commands:

1. Run the adSyncContext script to synchronize the database context file with the one in the file system:

   ```
   $ perl $AD_TOP/bin/adSyncContext.pl
   -contextfile=$CONTEXT_FILE
   ```

2. Register the managed servers of the newly added Application Tier node with the Oracle HTTP Server:

   ```
   $ perl $FND_TOP/patch/115/bin/txkSetAppsConf.pl
   -ctxfile=$CONTEXT_FILE \
   -outfile=$APPLRGF/TXK/txkSetAppsConf.log
   ```

After adding new nodes, refer to My Oracle Support Knowledge Document 1375686.1, *Using Load-Balancers with Oracle E-Business Suite Release 12.2*, for details of how to set up load balancing.

> **Note:** If SQL*Net Access security is enabled in the existing system, you first need to authorize the new node to access the database through SQL*Net. Refer to the Oracle Applications Manager online help for instructions on how to accomplish this.

Option 4: Cloning an Oracle RAC System

> **Note:** Cloning a RAC-enabled system is currently not certified with Rapid Clone.

Option 5: Adding a Node to an Existing Oracle RAC Cluster

As of Release 12, Rapid Clone no longer supports the migration of a Database Tier to Oracle RAC. Refer to My Oracle Support Knowledge Document 1570554.1, *Adding and Deleting Oracle RAC Nodes for Oracle E-Business Suite Release 12.2.x on Oracle Database 11gR2*, for instructions on how to perform this task.

Option 6: Cloning the Database Separately

Some situations require the database to be recreated separately, without using Rapid Clone. A typical scenario may be when a system downtime is not feasible, or an advanced database replication tool like RMAN is used to copy the database in hot backup mode.

This section documents the steps that allow manual creation of the Target database control files within the Rapid Clone process. You should use this method for databases located on raw partitions, or when cloning a hot backup. Follow all the steps in Standard Cloning Tasks, page 17-6, but replace the step to Configure the Target System

Database Server, page 17-12 with the following steps:

1. **Log in as ORACLE to the Target System**

2. **Configure the Database Oracle Home**

   ```
   $ cd <RDBMS ORACLE_HOME>/appsutil/clone/bin
   $ perl adcfgclone.pl dbTechStack
   ```

3. **Create the Target Database control files manually**

   In this step, you copy and recreate the database using your preferred method, such as RMAN restore, Flash Copy, Snap View, or Mirror View.

4. **Start the Target Database in open mode**

5. **Run the library update script against the database**

   ```
   $ cd <RDBMS ORACLE_HOME>/appsutil/install/<CONTEXT NAME>
   $ sqlplus "/ as sysdba" @adupdlib.sql <libext>
   ```

   Where <libext> should be set to `sl` for HP-UX, `so` for any other UNIX platform, or `dll` for Windows.

6. **Configure the Target Database**

   The database must be running and open before performing this step.

   ```
   $ cd <RDBMS ORACLE_HOME>/appsutil/clone/bin
   $ perl adcfgclone.pl dbconfig <Database Target Context File>
   ```

   Where "Database Target Context File" is: <RDBMS ORACLE_HOME>/appsutil/<Target CONTEXT_NAME>.xml.

   > **Note:** The dbconfig option will configure the database with the required settings for the new Target, but it will not recreate the control files.

Automating Rapid Clone

This section describes various cloning procedures that can be automated while cloning Oracle E-Business Suite with Rapid Clone. These procedures include:

- Cloning the Context File Only, page 17-25
- Cloning the Context File in Non-Interactive Mode, page 17-25
- Running adcfgclone.pl in Non-Interactive Mode, page 17-26

Please note that directory names `/u04/SOURCE` and `/u04/TARGET` used in the

command lines later in each procedure are given as examples. You should execute the commands with directory names suitable to your business.

Cloning the Context File Only

Run the following commands to automate the context file cloning procedure:

> **Note:** This command will not affect the Source System.

- **Database Tier**

  ```
  perl adclonectx.pl contextfile=/u04/SOURCE/db/tech_st/11.2.0/
  appsutil/SOURCE_ap9041pc.xml outfile=/u04/TARGET/db/tech_st/
  11.2.0/appsutil/TARGET_ap9041pc.xml
  ```

  This command will create `/u04/TARGET/db/tech_st/11.2.0/appsutil/TARGET_ap9041pc.xml`.

- **Applications Tier**

  ```
  perl adclonectx.pl contextfile=/u04/SOURCE/inst/apps/SOURCE_
  ap9041pc/appl/admin/SOURCE_ap9041pc.xml outfile=/u04/TARGET/inst/
  apps/TARGET_ap9041pc/appl/admin/TARGET_ap9041pc.xml
  ```

  This command will create `/u04/TARGET/inst/apps/TARGET_ap9041pc/appl/admin/TARGET_ap9041pc.xml`.

Cloning the Context File in Non-Interactive Mode

Run the following commands to automate the context file cloning in non-interactive mode:

> **Note:** Ensure that you add all the variables prompted during a normal interactive session. You might want to execute the commands several times until it is clean without any variable issues.

- **Database Tier**

  ```
  perl adclonectx.pl contextfile=/u04/SOURCE/db/tech_st/11.2.0/
  appsutil/SOURCE_ap9041pc.xml outfile=/u04/TARGET/db/tech_st/11.2.0/
  appsutil/TARGET_ap9041pc.xml pairsfile=/d01/stage/pairsfile.txt
  noprompt
  ```

- **Applications Tier**

  ```
  perl adclonectx.pl contextfile=/u04/SOURCE/inst/apps/SOURCE_
  ap9041pc/appl/admin/SOURCE_ap9041pc.xml outfile=/u04/TARGET/inst/
  apps/TARGET_ap9041pc/appl/admin/TARGET_ap9041pc.xml pairsfile=
  /u04/TARGET/pairsfile.txt noprompt
  ```

  The pairsfile is a text file only contains variables and values. For example:

- s_base=/u04/TARGET
- s_at=/u04/TARGET/apps/apps_st/appl

Running adcfgclone.pl in Non-Interactive Mode

Run the following commands to execute adcfgclone.pl using a pre-created context file:

- **Database Tier**

```
perl adcfgclone.pl dbTier /u04/TARGET/db/tech_st/11.2.0/
appsutil/TARGET_ap9041pc.xml
```

- **Applications Tier**

```
perl adcfgclone.pl appsTier /u04/TARGET/inst/apps/TARGET_
ap9041pc/appl/admin/TARGET_ap9041pc.xml
```

Other Cloning Procedures

This section describes the cloning procedures that can be executed separately. All the cloning procedures described here can also be run in non-interactive mode.

> **Note:** If you have installed Oracle E-Business Suite Extensions for Oracle Endeca, see My Oracle Support Knowledge Document 1574273.1, *Installing Oracle E-Business Suite Extensions for Oracle Endeca*, for information about cloning the integrated Oracle Endeca instance.

Cloning Procedures

Available cloning procedures can be summarized as follows:

- **Database Tier**
  - Cloning the technology stack on the Database Tier (`dbTechStack`)

    See: Cloning the Technology Stacks Only, page 17-27
  - Cloning the database (`database`)

    See: Cloning the Database Only, page 17-28
  - Configuring the database only

    See Configuring the Database Only, page 17-28
- **Applications Tier**
  - Cloning the technology stack on the Application Tiers (`atTechStack`)

    See: Cloning the Technology Stacks Only, page 17-27

- Cloning and configuring the APPL_TOP (`appltop`)

  See: Cloning the APPL_TOP Only, page 17-28

Cloning the Technology Stacks Only

Run the following commands to clone and configure the relevant technology stack components:

- **Database Tier**

  Run this command to clone the Database tier technology stack:

  ```
  $ perl adcfgclone.pl dbTechStack
  ```

- **Applications Tier**

  Run this command to clone the Applications tier technology stack, namely the OracleAS 10.1.2 Oracle Home, Oracle Fusion Middleware (FMW) Home, Oracle WebLogic Server domain configuration, and Oracle HTTP Server configuration:

  ```
  $ perl adcfgclone.pl atTechStack
  ```

  Run this command to clone the OracleAS 10.1.2 Oracle Home only:

  ```
  $ perl adcfgclone.pl dev10gHome
  ```

  Run this command to clone the FMW Home binaries only:

  ```
  $ perl adcfgclone.pl fmwHome
  ```

  Run this command to clone the Oracle WebLogic Server domain configuration only:

  ```
  $ perl adcfgclone.pl wlsConfig
  ```

  Run this command to clone the Oracle HTTP Server configuration only:

  ```
  $ perl adcfgclone.pl ohsConfig
  ```

Important Restrictions on Applications Tier Cloning

- The wlsConfig and ohsConfig cloning commands will work only if a context file is passed in as an argument. In other words, for the wlsConfig and ohsConfig commands the context file will not be cloned.

- The fmwHome cloning command should be performed before the wlsConfig command is performed.

- The ohsConfig cloning command cannot be performed before the wlsConfig command is performed.

- Even though the ohsConfig cloning command will work without WebLogic Admin server running, it is advisable to ensure WebLogic Admin Server has been started before performing the ohsConfig cloning command.

Cloning the Database Only

Run the following command to recreate the control file and configure the database.

> **Note:** This command will *not* reconfigure the Database Oracle Home.

```
$ perl adcfgclone.pl database
```

Configuring the Database Only

Run the following command to clean the topology data model and nodes, as well as reconfigure the Oracle E-Business Suite data for the new target.

> **Note:** Ensure that the database is running and open before executing this command.

```
$ perl adcfgclone.pl dbconfig
```

Cloning the APPL_TOP Only

Run the following command to only clone and configure the APPL_TOP only.

> **Note:** This will *not* clone or configure the Applications tier technology stack.

```
$ perl adcfgclone.pl appltop
```

Index

A

AccessGate, 16-1
AD Administration
  and AD Splicer, 11-2
adchkcfg, 3-19
AdminAppServer utility, 1-8
Administering Oracle E-Business Suite security, 1-13
AD Splicer, 11-2, 11-3
adstpall.sh, 4-183
adstrtal.sh, 4-183
ANSI dates
  in time zones, 8-2
AOL/J Setup Test Suite, 1-17
Apache , 1-5
Application Development Framework
  integration with, 12-7
Application Development Framework (ADF), 16-1
Application Server Security, 1-13
AppsDataSource, 16-1
AutoConfig, 1-22, 11-2
  Configuration Editor , 11-2
AutoConfig driver file
  format, 3-21
AutoPatch
  and AD Splicer, 11-2
Autovue Document Print Service, 1-19

C

Client time zone
  definition, 8-1
Cloning, 17-1
  Advanced options, 17-15
  Automating Rapid Clone, 17-24
  Finishing Tasks, 17-14
  Other Cloning Procedures, 17-26
  Prerequisite Tasks, 17-3
  Standard Tasks, 17-6
Common UNIX Printing System (CUPS), 6-28
CONCSUB, 4-45
Concurrent: Report Access Level profile (obsolete), 4-8
Concurrent Manager Operator responsibility, 4-183
Concurrent managers, 4-176
  activating a manager, 4-134
  activating and other control states, 4-152
  assigning work shifts, 4-163
  controlling, 4-134, 4-151, 4-181
  defining, 4-100, 4-148, 4-160
  defining combined specialization rules, 4-169
  defining work shifts, 4-167
  disabling a work shift, 4-102
  Internal concurrent manager, 4-138
  operating system process ID number, 4-157
  Oracle process ID number, 4-157
  PMON cycle, 4-135
  program libraries, 4-101
  reporting on work shifts, 4-115, 4-115
  restarting a manager, 4-134
  role of application name in combined rules, 4-170
  role of application name when defining, 4-160

sleep time, 4-164
specializing - Define Managers form, 4-165
specializing managers, 4-100, 4-116
Standard manager, 4-101
target processes, 4-1
time-based queues, 4-106
viewing actual number of processes, 4-151
viewing manager control processes, 4-155
viewing manager request queue, 4-158
viewing number of running requests, 4-152
viewing status of, 4-151
viewing target number of processes, 4-152
work shifts, 4-102
work shifts and target processes, 4-104, 4-163
work shifts hours, 4-103
work shifts overlap, 4-103
work shifts overlap - same priority, 4-104
work shifts past midnight, 4-103
Concurrent processing
file purging guidelines, 4-230
lifecycle of a request, 4-2
managing files and tables, 4-229
overview, 4-1
profile options, 4-231
programs, 4-71, 4-89
purge files program, 4-233
purging and audit data, 4-231
purging request data, 4-229
System Administrator privileges, 4-225
viewing incompatible tasks, 4-71
when programs start, 4-1
Concurrent programs
and requests, 4-1
behavior of program parameters, 4-54
behavior of report set parameters, 4-54
changing responsibility to see changed effects, 4-54
CONCSUB, 4-45
copying and modifying, 4-50
custom, 4-38
database session control, 4-78
defining, 4-89
defining incompatibility rules, 4-34
disabling, 4-71
displaying parameters - programs vs. report sets, 4-54
enforcement of incompatibility rules, 4-37
example - modifying program parameters, 4-60
execution method, 4-67, 4-71
grouping as a request type, 4-172
grouping as request types, 4-132
incompatible, 4-71, 4-80
modifying incompatible programs list, 4-53
modifying parameters, 4-53
multiple language support, 4-67, 4-71
not displaying parameters, 4-54
parameter sequence, 4-82
program libraries, 4-101
reporting on enabled programs, 4-63
reporting on incompatible programs, 4-63
reporting on program definitions, 4-63
report set incompatibilities, 4-25
role of application name in request types, 4-172
run alone programs, 4-34
running alone, 4-71, 4-71
setting default values, 4-54
spawned vs. immediate, 4-101
subroutines, 4-67
viewing, 4-71
warnings about modifying, 4-50, 4-54
Concurrent requests
changing phase and status, 4-227
changing priority of, 4-228
child requests, 4-2
explained, 4-1
file access privileges, 4-225
lifecycle of, 4-2
multilingual support, 4-217
output file access privileges, 4-225
parent requests, 4-2
phase/status listing, 4-2
phase and status, 4-2
priority of, 4-2
request types, 4-172
role of application name in request types, 4-172
submitting using CONCSUB, 4-45
System Administrator privileges, 4-225
time taken to run, 4-114
viewing output, 4-173
viewing request parameters, 4-54
viewing status of, 4-224

Concurrent request type, 4-132
Configuring Oracle E-Business Suite, 1-1
Conflict domains
  defining, 4-88
  explained, 4-35
Context-sensitive help
  link syntax, 7-7
Controlling access to reports or programs
  Report Groups, 4-8
Custom concurrent programs, 4-38
Custom help
  context-sensitive links, 7-7
  global environment, 7-16
  Help Builder, 7-9
  help navigation trees, 7-8
  updating the search index, 7-8
Custom help files
  linking help files, 7-5
Custom reports
  context-sensitive help, 7-7

D

Data Groups
  Application Object Library requirement, 4-87
  defining, 4-86
DBC files, 1-10
Default parameters
  Shared parameters, 4-20
deployment plans
  Oracle WebLogic Server, 3-45
Developer tools, 12-1
Document sequences
  active sequence definitions, 10-6
  application, 10-5, 10-12
  assigning sequences to document definitions, 10-5, 10-6
  auditability, 10-1
  automatic document numbering - initial value, 10-2
  automatic numbering, 10-8
  automatic vs manual, 10-7
  category, 10-5, 10-12
  category application, 10-10
  category code, 10-10
  category identifies database table, 10-10, 10-10
  defining a sequence, 10-1
  defining document categories, 10-10
  defining documents to be numbered, 10-5
  differences - document numbering vs. entry, 10-7
  disabling a sequence assignment, 10-13
  document categories explained, 10-4
  document definition, 10-12
  Document Flexfield, 10-5
  enabling segments in the document flexfield, 10-14
  end date - document definition, 10-13
  end date - document flexfield, 10-13
  examples - document categories, 10-4
  examples - sequence definitions, 10-2
  initial value of sequence, 10-8
  manual numbering, 10-8
  message displayed by document, 10-8
  method in document flexfield, 10-5, 10-14
  segments and the document flexfield, 10-14
  sequence names, 10-8
  sequences and audit records, 10-8
  sequences explained, 10-1
  sequence start date, 10-8
  sequence type, 10-8
  set of books in document flexfield, 10-5, 10-14
  start date - document definition, 10-13
  start date - document flexfield, 10-13
  type of document numbering, 10-8
  type of sequence numbering, 10-2

E

Exportable profile options for Oracle iSetup, 5-8
External documents, 9-6

F

File Character Set Converter, 11-3
FNDGFU, 13-25
FNDLOAD, 13-1, 13-3
Form
  passing arguments to, 4-29
Form Personalization, 12-1
Forms
  Administer Concurrent Managers, 4-151
  Concurrent Conflicts Domains, 4-88
  Concurrent Request Types, 4-172
  Define Combined Specialization Rules, 4-169

Define Concurrent Manager, 4-160
Define Data Group, 4-86
Define Printer Driver, 6-53
Define Printer Types, 6-47
Define Print Style, 6-50
Define Report Group, 4-65
Define Work Shifts, 4-167
Languages, 9-9
Natural Languages, 9-10
Register Nodes, 4-175
Register Printers, 6-49
Territories, 9-11
Fusion Middleware Control
Oracle Applications Manager, 2-4

G

Generic File Manager, 13-25, 13-28
Generic Loader, 13-1
Oracle Application Object Library
configuration files, 13-9
Globalization support, 9-1

H

Help system
customizing Oracle E-Business Suite help, 7-1
Help System Utility, 7-2
File Name to Help Target Report, 7-4
Help Target to File Name Report, 7-4
Uploading help files, 7-3
Hierarchy type
user profiles, 5-1
Hijrah calendar support, 9-4
Home page, 1-19

I

ICX:Language profile option, 9-3
Incompatible programs
Concurrent programs, 4-34
Internal concurrent manager
CONCSUB command, 4-138, 4-140
CONCSUB - hiding password, 4-143
CONCSUB - using to shut down, 4-142
control states, 4-135
enforces incompatibility rules, 4-37
explained, 4-101
internal monitors, 4-146
operating system control, 4-138
parallel concurrent processing, 4-145
PMON cycle, 4-135
shut down from operating system, 4-142
starting from operating system, 4-139
STARTMGR command, 4-138
when inactive, 4-225
Internal Manager environment, 4-216

J

Java Authentication and Authorization Service (JAAS), 16-1

L

Languages, 9-1
License Manager, 11-2
compared to AD Splicer, 11-3
Lightweight MLS, 9-6
Loaders, 13-1
Log files
access level profile option, 4-225
parallel processing on multiple nodes, 4-147
purge program, 4-233
Logical databases
define for custom applications, 4-62
explained, 4-61
program incompatibility rules, 4-61
Standard logical database method, 4-62
Login page
configuration, 1-2

M

Message Dictionary Generator, 13-23
MIME types
for viewing reports, 4-173
MLS
Multilingual support, 4-217
MLS function, 4-67, 4-71
Multi-language function (MLS function), 4-71
Multilingual external documents, 9-6
Multilingual support
concurrent requests, 4-217
multiple organizations
reporting, 4-218

N

Node
  explained, 4-175
Nodes
  explained, 4-144
  manager's target node, 4-149
  primary and secondary, 4-145

O

OAM
  utilities, 11-2
OATM, 14-1
off-cycle products
  definition, 11-3
Oracle Access Manager, 16-1
Oracle Application Framework, 1-19
Oracle Application Object Library AOL/J Setup Test Suite, 1-17
Oracle Applications Manager, 2-4, 4-194
  AutoConfig, 1-22
  Concurrent Processing, 4-207
  Configuration Overview, 2-4
  creating services (managers), 4-107
  Generic Collection Service, 4-206
  Internal Manager environment, 4-216
  Oracle Enterprise Manager 11g Fusion Middleware Control, 2-4
  Service Instances, 4-195
  Service status, 4-201
  Setup, 2-4
  Site Map, 2-4
Oracle Applications Tablespace Model, 14-1
Oracle E-Business Suite Home Page
  personalizing, 1-4
Oracle Enterprise Manager 11g Fusion Middleware Control, 2-4
Oracle GoldenGate, 16-1
Oracle HTTP Server, 1-5
Oracle iSetup
  exportable profile options, 5-8
Oracle RAC (Real Application Clusters), 15-2
Oracle Real Application Clusters (Oracle RAC), 15-1
Oracle Reports
  bitmapped, 4-71
Oracle Tutor, 7-17
Oracle WebLogic Server
  use in Oracle E-Business Suite, 3-1
Output post processor, 4-219

P

Parallel concurrent processing
  explained, 4-143
  Internal manager, 4-145
  introduced, 4-143
  log files and multiple nodes, 4-147
  managing, 4-147
  operating environments, 4-144
  Oracle RAC, 15-2
  proprietary queuing systems, 4-147
Pasta
  add a new printer type, 6-13
  basic setup for printers, 6-10
  configuration file setup
    defining configuration files for different printers, 6-11
    setting margins, 6-13
    using a different configuration file as the default, 6-11
  defining the configuration file, 6-11
  executable, 6-10
  modifying a printer type to use Pasta, 6-12
  pasta.cfg file, 6-11
  setting margins, 6-13
Pasta Universal Printer, 6-10
PMON cycle
  concurrent managers, 4-135
Portlet Generator, 12-4
Printer setup
  printer types
    modify an existing printer type to use Pasta, 6-13
Printer support
  arguments, 6-55
  arguments for print command, 6-34
  caching of definitions, 6-9
  Command driver method, 6-33
  concurrent managers - restarting, 6-9
  concurrent program print definitions, 6-39
  custom print programs - location, 6-34
  drivers, styles, printer types and platforms, 6-4

end user settings, 6-43
fast-track printer setup with Pasta, 6-10
header pages, 6-52
initialization, 6-56
initialization string, 6-31
initialization string - editing, 6-31
introduction to printing, 6-1
page break problems, 6-32
platform, 6-54
postscript printing, 6-43
predefined types, styles, drivers, 6-31
print command & arguments - example, 6-34
printer / style assignments, 6-39
printer assignments, 6-40
printer driver method, 6-33
printer drivers - assigning, 6-47
printer drivers - defining, 6-53
printer drivers - explained, 6-3
printer drivers - introduction, 6-2
printer drivers - predefined for printers, 6-53
printer drivers - predefined for styles, 6-53
printer drivers - when to define new drivers, 6-53
printers - operating system name, 6-49
printers - registering, 6-49
printer types
modifying a printer type to use Pasta, 6-12
printer types - defining, 6-47
printer types - introduction, 6-2
print style assignments, 6-41
print styles - columns, 6-52
print styles - defining, 6-50
print styles - explained, 6-3
print styles - introduction, 6-2
print styles - predefined, 6-50
print styles - rows, 6-52
Program driver method, 6-33
Program driver method - example, 6-33
program name, 6-55
reset, 6-56
reset string, 6-31
reset string - editing, 6-31
sequence of printing events, 6-4
setting up - forms used, 6-8
setting up printers, 6-8
setting up printers using Pasta, 6-10
shell scripts, 6-34
spool file, 6-31, 6-55
SRW driver, 6-54
SRW driver - customizing, 6-31
SRW drivers - how used, 6-38
SRW drivers - location, 6-38
standard input, 6-36, 6-55
Subroutine driver method, 6-33
System Administrator privileges, 6-42
verifying printer drivers, 6-32
PrintForms, 6-25
Printing, 4-187
Print options
for reports, 4-20
Profile categories, 5-8
profile options
hierarchy types, 5-1
profile hierarchy, 5-1
setting options, 5-4
when changes take effect, 5-4
Profiles, 5-1
Purge Obsolete Generic File Manager Data concurrent program, 13-28

R

Rapid Clone, 17-1
Real Application Clusters, 15-2
Register
concurrent program, 4-71, 4-89
Report Groups
defining, 4-65
example - using a code, 4-32
may consist of, 4-28
Report Security Groups, 4-29
report security groups, report sets, reports, 4-24
responsibility-level vs. form-level, 4-28
using, 4-28
using a code to customize, 4-65
using a code with, 4-29
vs. report sets, 4-7
reporting
multi-org, 4-218
Report parameters
sharing in a report set, 4-26
Reports

Completed Concurrent Requests, 4-114
Concurrent Program Details, 4-63
Concurrent Programs, 4-63
Purge Concurrent Request and/or Manager Data, 4-233
Report Group Responsibilities, 4-33
Report Sets, 4-27
User Profile Option Values, 5-9
Work Shift by Manager, 4-115
Work Shifts, 4-115
Report Security Groups
Report Groups, 4-29
Report Sets
as concurrent programs, 4-17
behavior of program parameters, 4-54
defining, 4-11
displaying parameters - programs vs. report sets, 4-54
example - shared parameters, 4-26
incompatibility rules, 4-25, 4-34
owners of, 4-23
preventing parameters from being changed, 4-54
printing, 4-16
querying in Define Concurrent programs form, 4-25
reporting on, 4-33
reporting on definitions, 4-27
report security groups, report sets, reports, 4-24
request phase and status, 4-17
sharing parameters in a set, 4-26
System Administrator privileges, 4-24
vs. report groups, 4-7
Reports Viewer, 4-226
Request parameters, 4-20
Request security groups, 4-8
Request sets
holding, 4-229
managing, 4-229
print options, 4-20
request parameters, 4-20
restarting, 4-229
running reports in parallel, 4-18
shared parameters, 4-20
Request type, 4-132
Responsibilities
reporting on reports and report sets, 4-33
Role-Based Access Control (RBAC)
with concurrent programs, 4-8
Run Reports form
customizing using codes, 4-32
Run Requests form
example - customizing, 4-32

S

Secure Sockets Layer
use in Oracle HTTP Server, 1-5
Security, 1-1
server, administering, 1-13
server trust level, 1-16
Server security, 1-13, 1-16
server time zone
definition, 8-1
Server trust level, 1-16
Service Management, 4-4
Shared parameters
behavior of
Report Sets, 4-26
changing a value, 4-20
request sets, 4-20
Sharing parameters
in request sets, 4-20
Specializing managers
actions, 4-116
action types, 4-116
defining combined rules, 4-126
defining specialization rules, 4-116
examples of action types, 4-119
examples of combined rules, 4-128
examples of rules, 4-121
explained, 4-116
introduction, 4-116
specialization vs. combined rules, 4-131
using more than one rule, 4-117
ssl.conf
SSL configuration file, 1-5
Standard Report Submission
explained, 4-7
Standard Request Submission
print options, 4-20
request parameters, 4-20
Standard Submission form

customizing, 4-8, 4-29
example - customizing, 4-32
explained, 4-7
list, 4-29
SyncContext tool, 3-13
System Administrator
report set privileges, 4-24

T

Target processes
Concurrent managers, 4-1
Thai calendar support, 9-4
Transaction Managers, 4-101

U

User-Preferred Time Zone Support, 8-1
User profiles, 5-1
assigning hierarchy type, 5-1
examples of, 5-7
reporting on, 5-9
using site level as defaults, 5-5
UTF8, 9-4

V

Viewer Options window, 4-173

W

Web Administration service group, 3-5
Web-Enabled PL/SQL Window, 12-3
WebLogic Server Basic
definition, 3-3
Work Directory, 12-2