

Oracle® One-to-One Fulfillment

Implementation Guide

Release 12

Part No. B28653-01

October 2006

Oracle One-to-One Fulfillment Implementation Guide, Release 12

Part No. B28653-01

Copyright © 2003, 2006, Oracle. All rights reserved.

Primary Author: Michael Meditzky

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments

Preface

1 Introduction

Overview of Oracle One-to-One Fulfillment	1-1
Basic Terminology.....	1-2
Templates, Master Documents and Collateral	1-2
Cover Letters.....	1-3
E-Business Suite Use of Templates, Master Documents and Collateral.....	1-3
Process Flow.....	1-4
Concepts	1-6
Advanced Queuing	1-7
Email Bounceback.....	1-7
Fulfillment Requests.....	1-7
Fulfillment Server.....	1-8
Remote Commands.....	1-9
Fulfillment Channels.....	1-9
Agent Groups.....	1-9
Agent Resources.....	1-9
Email Tracking.....	1-9
Templates.....	1-10
MES Master Documents.....	1-10
Oracle Content Manager Profile Options.....	1-11
Queries.....	1-12
Data Source.....	1-14
Collateral.....	1-14

Fulfillment Request History.....	1-14
Fulfillment Request Status.....	1-15
Physical Collateral.....	1-15
Unsubscribe Option.....	1-15
Resubmit	1-16
Pause and Resume Request Processing.....	1-16
Deletion of Sent Fulfillment Requests.....	1-17
Master Document Editor.....	1-17
High Quality Printing	1-17

2 Dependencies and Integration Points

System Requirements.....	2-1
Fulfillment Server.....	2-1
Output Servers.....	2-2
Mandatory Dependencies.....	2-2
Conditional Dependencies.....	2-7
Related Documentation.....	2-8
Installing Oracle One-to-One Fulfillment.....	2-8
Accessing Oracle One-to-One Fulfillment.....	2-8
User Accounts.....	2-8
Responsibilities.....	2-9

3 Implementation Tasks

Implementation Task Sequence.....	3-1
Defining an Administrator.....	3-5
Creating an Administrator User Account.....	3-6
Assigning a Role to a User Account.....	3-7
Defining an Agent.....	3-8
Creating an Employee.....	3-9
Creating a User Account for an Agent.....	3-11
Assigning or Importing a Resource for an Employee.....	3-12
Configuring Servers	3-13
Creating a Fulfillment Server.....	3-14
Defining an Email Server.....	3-16
Defining a Fax Server.....	3-18
Adding an Available Print Server.....	3-19
Creating a New Print Server for RightFax Printing.....	3-20
Implementing a Fulfillment Server.....	3-20
Creating a Remote Command Script for the Fulfillment Server.....	3-21

4 Integration Tasks

Integrating with Captaris RightFax	4-1
High Level Integration Steps.....	4-1
Sending a Fax Fulfillment Request by RightFax.....	4-2
Sending a Print Fulfillment Request by RightFax.....	4-4

5 Diagnostics and Troubleshooting

Introduction.....	5-2
Common Error Messages.....	5-2
Out of Memory Error.....	5-2
java.lang.NullPointerException or Blank Screen.....	5-3
Fulfillment Server Error Messages and Debug Flags	5-3
Common Fulfillment Server Errors.....	5-3
Fulfillment Server Debug Flags.....	5-5
Stopping and Restarting the Fulfillment Server.....	5-7
Email Issues.....	5-7
java.io.IOException: 500 5.5.1 Command Unrecognized.....	5-8
Characters Appear as '?' in Body of Email	5-8
Frequently Asked Questions.....	5-8
Uploading an RTF Document Fails in Oracle Content Manager.....	5-9
Content Types Not Visible.....	5-9
Cannot Update Existing Content Type.....	5-9
Log Directory Path.....	5-9
TeleSales Java Exception.....	5-9
java.lang.IllegalArgumentException.....	5-10
DoNotContactException.....	5-11
Commands in Background Mode.....	5-12
Viewing Request Processing.....	5-13
"Initializing Processor" Error.....	5-13
Merge Field Errors.....	5-14
Unable to Use Merge Field Delimiters « and ».....	5-14
Table Rows and Merge Fields.....	5-15
Server id and User id in create_fulfillment API Call.....	5-21
Sending a Request Through Multiple Media Types.....	5-22
Debugging.....	5-22
PL/SQL Debugging.....	5-22
Server Debugging.....	5-23
Troubleshooting the RightFax Integration.....	5-24

6 Administration Tasks

Managing Servers.....	6-3
Setting the Fulfillment Server Start Time and Shutdown Time.....	6-4
Setting the Interaction History Outcomes and Results.....	6-5
Enabling the Unsubscribe Option in Email Requests.....	6-5
Enabling Tracking for Email Requests.....	6-6
Setting Up Bounceback.....	6-7
Defining Print Servers and Printers for High-Quality Printing.....	6-9
Configuring a Print Server for High-Quality Printing.....	6-9
Configuring Printers for High-Quality Printing.....	6-10
Managing Fulfillment Agent Groups	6-11
Creating an Agent Group.....	6-11
Adding an Agent to a Group.....	6-12
Removing an Agent from a Group.....	6-13
Assigning a Fulfillment Agent Group to a Fulfillment Server.....	6-14
Managing Oracle Marketing Encyclopedia System Templates.....	6-15
Creating a Template.....	6-15
Assigning a Master Document to a Template.....	6-17
Assigning Collateral to a Template.....	6-18
Updating a Template.....	6-19
Removing a Master Document from a Template.....	6-20
Removing Collateral from a Template.....	6-21
Managing Oracle Marketing Encyclopedia System Master Documents	6-22
Creating an Oracle Marketing Encyclopedia System Master Document Using the Text Editor	6-22
Uploading an Oracle Marketing Encyclopedia System Master Document.....	6-24
Updating an Oracle Marketing Encyclopedia System Master Document.....	6-25
Viewing an Oracle Marketing Encyclopedia System Master Document.....	6-26
Downloading an Oracle Marketing Encyclopedia System Master Document.....	6-26
Assigning a Query to an Oracle Marketing Encyclopedia System Master Document.....	6-27
Removing a Query From an Oracle Marketing Encyclopedia System Master Document....	6-27
Viewing Oracle Marketing Online Collateral	6-28
Managing Oracle Marketing Encyclopedia System Queries.....	6-28
Creating a Query.....	6-29
Updating a Query.....	6-30
Viewing Oracle Marketing Encyclopedia System Master Documents Associated With a Query.....	6-30
Creating a Data Source File from a Query	6-31
Managing Content	6-32
Cover Letter Overview.....	6-32

Understanding Cover Letter Content Types.....	6-33
Queries.....	6-35
Creating and Managing Cover Letter Content Types.....	6-36
Creating Cover Letters.....	6-39
Managing Fulfillment Requests	6-42
Viewing the Status of a Fulfillment Request.....	6-42
Canceling a Fulfillment Request.....	6-43
Pausing or Resuming a Fulfillment Request.....	6-44
Managing Fulfillment Request History	6-45
Viewing the Details of a Fulfillment Request.....	6-45
Viewing Failed Fulfillment Requests.....	6-48
Viewing a Partially Successful Fulfillment Request.....	6-48
Correcting a Malformed Email Job.....	6-49
Viewing Job Details.....	6-50
Resubmitting a Job.....	6-52
Resubmitting a Fulfillment Request.....	6-53
Deleting a History Entry.....	6-53
Searching for a Fulfillment Request	6-54
Using the Quick Find Feature	6-55
Using Merge Fields.....	6-56
Default and Wildcard Merge Delimiter Pairs	6-57
Different Representations of the Same Character	6-57
Using Encrypted Merge Fields	6-58
Creating an RTF Form Document with Merge Fields.....	6-58
Creating a PDF Form Document with Merge Fields.....	6-68
Setting Up High Quality Printing.....	6-69
Configuring the Printer.....	6-69
Setting Up a Pasta Print Server.....	6-70
Verifying the Printer Configuration.....	6-71
Running Concurrent Programs.....	6-72

A Fulfillment Server Commands

Fulfillment Server Startup Script Parameters.....	A-1
Fulfillment Server Commands.....	A-1
Required Flags.....	A-2
Optional Flags.....	A-3
Debug Flags.....	A-6
Additional Java Flags.....	A-8
Language Character Parsing	A-9

B APIs

Introduction.....	B-1
Priority Levels	B-2
Interaction History Logging.....	B-2
Extended Headers.....	B-3
Extended Header XML Format.....	B-3
Start_Request	B-4
Get_Content_Xml.....	B-5
One-to-One Fulfillment Request Types.....	B-6
Example <item> XML Element	B-7
Messages and Notifications.....	B-18
Business Events.....	B-18
Common Messages	B-19
Get_Content_XML Messages.....	B-19
Example Scripts.....	B-21

C Profile Options

Fulfillment Profile Options.....	C-1
----------------------------------	-----

Glossary

Index

Send Us Your Comments

Oracle One-to-One Fulfillment Implementation Guide, Release 12

Part No. B28653-01

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

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

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

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

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

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

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

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

Preface

Intended Audience

Welcome to Release 12 of the *Oracle One-to-One Fulfillment Implementation Guide*.

This guide is intended for personnel who are responsible for implementing and administering the Oracle One-to-One Fulfillment application.

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

- The principles and customary practices of your business area
- Oracle One-to-One Fulfillment
- Oracle Self-Service Web Applications

To learn more about Oracle Self-Service Web Applications, read the *Oracle Self-Service Web Applications Implementation Manual*.

- The Oracle Applications graphical user interface

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

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

TTY Access to Oracle Support Services

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

Documentation Accessibility

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

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

Structure

1 Introduction

This chapter provides an overview of Oracle One-to-One Fulfillment.

2 Dependencies and Integration Points

This chapter describes the system requirements and product dependencies for Oracle One-to-One Fulfillment. It also describes how to access the administrative console for Oracle One-to-One Fulfillment.

3 Implementation Tasks

This chapter provides a high-level overview of the implementation task sequence, and references to procedures that you will do during the implementation.

4 Integration Tasks

The following sections explain how to integrate Oracle One-to-One Fulfillment with other Oracle products and third-party products.

5 Diagnostics and Troubleshooting

This chapter describes how to diagnose and fix issues that might arise during the implementation and usage of Oracle One-to-One Fulfillment.

6 Administration Tasks

This chapter describes task-based procedures for administering Oracle One-to-One Fulfillment.

A Fulfillment Server Commands

B APIs

C Profile Options

Glossary

Related Information Sources

Oracle Application Framework Personalization Guide

This guide describes how to personalize Oracle Application Framework-based Oracle E-Business Suite application pages as an end-user and as a personalization administrator using the Oracle Application Personalization Framework.

Note: Oracle One-to-One Fulfillment does not contain any end-user personalizable regions, and there are no special considerations that you need to be aware of when creating administrator-level personalizations of its regions or pages. For general information about how to create personalizations, refer to the *Oracle Application Framework Personalization Guide*.

Integration Repository

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

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

Do Not Use Database Tools to Modify Oracle Applications Data

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

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

Because Oracle Applications tables are interrelated, any change you make using an Oracle Applications form can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables

get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

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

Introduction

This chapter provides an overview of Oracle One-to-One Fulfillment.

This chapter covers the following topics:

- Overview of Oracle One-to-One Fulfillment
- Concepts

Overview of Oracle One-to-One Fulfillment

Oracle One-to-One Fulfillment provides Oracle E-Business Suite applications with a centralized mechanism for managing fulfillment. Fulfillment is the process of compiling and distributing information to customers.

Oracle One-to-One Fulfillment consists of a set of API calls, a server, and an administration user interface called the Administration Console. The Oracle One-to-One Fulfillment API is used by the E-Business Suite applications to initiate a request for fulfillment processing. The fulfillment request identifies the agent or server, the content, and the channel. The Oracle One-to-One Fulfillment server processes the request. It compiles the personalized content and determines the recipients of email, fax, print and physical collateral.

Each business application has a unique interface for making a fulfillment request on behalf of one or more parties. Examples of outbound correspondence in E-Business Suite include:

- Notification letters
- Dunning letters
- Lease terms
- Product information
- Survey email invitations

- Sales and marketing collateral

Basic Terminology

The following terms explain some basic concepts of Oracle One-to-One Fulfillment.

- **Cover letter or master document:** A document, sent by Oracle One-to-One Fulfillment, which typically includes merge fields that are populated by an associated SQL query.
- **Template:** A high-level definition of one or more Oracle One-to-One Fulfillment master documents, associated queries, and any related collateral.
- **Merge field:** Data elements in an Oracle One-to-One Fulfillment cover letter or master document that are surrounded by merge field delimiters. When a fulfillment request runs, any associated query retrieves the merge data from the table columns or views that are specified in the query. For each merge field that is used in a cover letter or master document, there must be one corresponding column name or alias in the select statement of the query to retrieve the appropriate data.
- **For Oracle Survey:** Invitation or reminder master documents contain at least one merge field to retrieve the survey URL that the recipient must access to participate in the survey. Other merge fields that are often part of invitation and reminder master documents are contact information (title, first name and last name) for each list member, and deployment end date.
- **Query:** A database query for information that is made in structured query language (SQL). Each Oracle One-to-One Fulfillment cover letter or master document may have an associated query. The information that the query gets is put into placeholders called merge fields resulting in the merged document. A query must have a column name or alias for each data element included as a merge field in the cover letter or master document. The query may also contain valid column names or aliases for additional data elements, whether or not the data elements are used in the cover letter or master document. For example, you can construct a single query that requests all information used to populate a survey invitation master document and a survey reminder master document, even if both documents have different merge fields. A single query can be associated with any number of cover letters or master documents. The data from a successfully executed query merges with the document.

Templates, Master Documents and Collateral

A template is a grouping of Master Documents and Collateral. (This concept of templates exists only in Oracle One-to-One Fulfillment, not in Marketing Encyclopedia System.) Master Documents are dynamic or static documents. Dynamic content in master documents is defined by merge fields enclosed within special characters. Merge

fields are tokens that are replaced by data when fulfillment requests are processed. Email and fax master documents can be HTML files or Zip files containing an HTML file plus an image file. Print master documents can be RTF or PDF files.

For RTF documents, merge fields can be entered using the Microsoft Form Field feature. In the form Field Help Text box, the merge field name should correspond to the name or alias used in the SQL query (in uppercase), and should be enclosed within the following text "<?" and "?>", for example <?CURRENT_DATE?>.

For PDF documents, merge fields can be entered using the Form Tool feature. In the Field Properties box, the merge field Name should correspond to the name or alias used in the SQL query.

Oracle One-to-One Fulfillment Master Documents can optionally have a query associated with them. Some E-Business Suite applications, TeleService for example, require an associated query.

Collateral (known as Deliverable in Oracle Marketing Online) is a static fulfillment document that does not contain merged data. Collateral are loaded by the Marketing Online Deliverables tab, and may be of any file format, for example, doc, xls, pdf, and so on.

Cover Letters

The cover letter (also known as content item) authoring screen allows the user to create and edit a cover letter based on the content type setup. A cover letter is the primary document that is sent to a customer by email, fax, or print and mail. The user can specify a cover letter name, description, and the folder where the cover letter is stored. The following attributes are also available for cover letters:

- **Body/Details** - The user can enter an HTML body for the cover letter. If the content type has a query associated with it, the user is given a choice to embed the tags or merge fields into the HTML body. The user also has the option to specify a text part of the email. The text part will be read by non-html compliant email clients. Fax requests will only use the HTML part of the content.
- **Attachments** – The user can search for file attachments from the repository, or upload an attachment from the desktop. By default the attachment is saved to the same location as the cover letter. The user has the option to specify another folder location in which user has the permission to create content.
- **Renditions** – The Oracle One-to-One Fulfillment print solution supports printing of PDF or RTF documents. These PDF or RTF documents may contain merge fields. The user can select PDF or RTF file from the desktop and upload it as the cover letter content.

E-Business Suite Use of Templates, Master Documents and Collateral

Master Documents only:

- TeleService (seeded master document & query used)
- Lease Management

Collateral only:

- Scripting building block that calls Oracle One-to-One Fulfillment

Master Documents and Collateral

- TeleSales

Templates, Master Documents and Collateral:

- Survey
- Collections

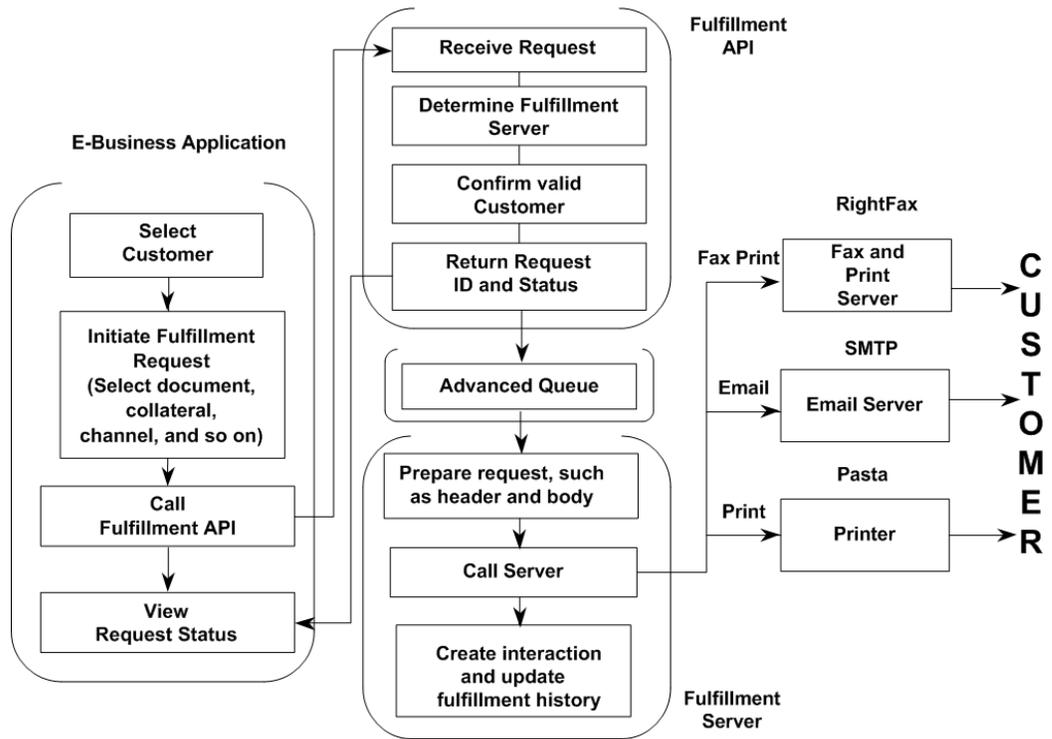
Cover Letters

- Oracle Marketing Online
- Student Systems

Process Flow

The following diagram illustrates the process flow of fulfillment requests made with Oracle One-to-One Fulfillment.

Oracle One-to-One Fulfillment Processing Flow



- The calling applications provide screens or concurrent processes that enable the selection of one or more customers and documents or collateral.
- The calling application calls the Oracle One-to-One Fulfillment APIs, passing in required data about the particular fulfillment request.
- The Oracle One-to-One Fulfillment APIs process the request and return a request ID (or any errors) to the application. If successful, the API formats and places the request on an Advanced Queue to be picked up by the Fulfillment Server.
- The Fulfillment Server monitors the Request Queue for new requests, and pulls the request from the queue.
- The Fulfillment Server downloads document and collateral from MES or OCM, and sends the request to Java Mail APIs or Fax and Print enablers, or XML Publisher for merging and then prints from Oracle Pasta.
- Oracle One-to-One Fulfillment writes request details to Fulfillment History. If flagged to do so by the API, an entry in Interaction History is written against the party_id passed in the request.

Concepts

Topics include:

- [Advanced Queuing](#), page 1-7
- [Email Bounceback](#), page 1-7
- [Fulfillment Requests](#), page 1-7
- [Fulfillment Server](#), page 1-8
- [Remote Commands](#), page 1-9
- [Fulfillment Channels](#), page 1-9
- [Agent Groups](#), page 1-9
- [Agent Resources](#), page 1-9
- [Email Tracking](#), page 1-9
- [Templates](#), page 1-10
- [MES Master Documents](#), page 1-10
- [Queries](#), page 6-35
- [Data Source](#), page 1-14
- [Collateral](#), page 1-14
- [Fulfillment Request History](#), page 1-14
- [Fulfillment Request Status](#), page 1-15
- [Physical Collateral](#), page 1-15
- [Unsubscribe Option](#), page 1-15
- [Resubmit](#), page 1-16
- [Pause and Resume Request Processing](#), page 1-16
- [Deletion of Sent Fulfillment Requests](#), page 1-17
- [Master Document Editor](#), page 1-17

- High Quality Printing, page 1-17

Advanced Queuing

Advanced Queuing is the message queuing functionality of the Oracle database. Advanced Queuing enables asynchronous communication between applications that use queues. A queue is a repository for messages. Queues are stored in queue tables. Each queue table is a database table and contains one or more queues.

Advanced Queuing supports native XML messages and also allows Advanced Queuing operations to be defined in the XML-based Internet-Data-Access-Presentation (IDAP) format. IDAP, an extensible message invocation protocol, is built on Internet standards, using HTTP and email protocols as the transport mechanism, and XML as the language for data presentation.

Business applications submit the fulfillment data in accordance with the Oracle One-to-One Fulfillment API procedures, which format the fulfillment request as an XML message. The API places fulfillment requests in a queue (enqueueing) and the Fulfillment Server retrieves fulfillment requests from a queue (dequeueing).

For more information about Advanced Queuing, see *Oracle 10g Application Developer's Guide - Advanced Queuing*.

Email Bounceback

Oracle One-to-One Fulfillment supports email bounceback processing, the process of returning email to the sending server.

For more details, see *Setting Up Bounceback*, page 6-7.

Fulfillment Requests

A fulfillment request is the data object that is submitted for fulfillment processing. The E-Business Suite application uses the Oracle One-to-One Fulfillment application programming interface (API) to specify the components of the content (for example, a master document from Oracle Marketing Encyclopedia System), the output channel (email, fax, or print), and the user of Fulfillment Server. The resulting data object is placed in the fulfillment request queue. The fulfillment request queue is provided by Advanced Queuing.

Each E-Business Suite application has a unique interface for making a fulfillment request on behalf of one or more parties. For information about how to initiate a fulfillment request, see the user guide or online help for the business application.

Content

The following table briefly describes the different type of content. For a more detailed description, see the relevant section.

Content Type	Location	Description	Set Up in Oracle One-to-One Fulfillment?
Master Document	Oracle Marketing Encyclopedia System	A static or dynamic HTML, Zip, RTF or PDF document.	Yes
Query	Oracle One-to-One Fulfillment	A SQL statement that provides data for a dynamic master document	Yes
Collateral	Oracle Marketing	A static document of any type	No
Template	Oracle One-to-One Fulfillment	A grouping of master documents and collateral documents.	Yes
Cover Letter	Oracle Content Manager	A static or dynamic HTML, Zip, RTF or PDF document.	Yes
Query	Oracle Content Manager	A SQL statement that provides data for a dynamic cover letter.	Yes

Fulfillment Server

Fulfillment Server is a multi-threaded Java server that processes fulfillment requests by de-queuing the request from an Advanced Queue. Business applications use Fulfillment APIs to place fulfillment requests in the queue.

You can specify the time during which the server polls the fulfillment request queues in the database. It does not physically start or stop the server. If the server is running, the server will continue to process retrieved fulfillment requests even when it is not polling the database.

When a fulfillment request is processed, it creates a report in the Fulfillment history and an interaction record in Oracle Customer Interaction History. You can select which outcome reason and result you want to associate with a successful or failed fulfillment request.

Remote Commands

The remote command script lets users interact with the Fulfillment Server without the need to maintain the original command prompt that was used to start the server. To issue a command, type the name of the remote command script and the command.

The Fulfillment Server is configured in the startup script to monitor a port for remote commands. The port is specified in both the startup script for the server and the remote command script. The remote command script runs on the same computer as the Fulfillment Server or on a computer that is on the same network as the Fulfillment Server.

When a command is sent through the remote command script, the computer that issued the command terminates activity after it receives a response to the command. To prevent unnecessary workload for the Fulfillment Server, the computer that issues the remote command retains a connection to the Fulfillment Server only for the duration of the remote command.

Fulfillment Channels

The Fulfillment Server compiles the content and sends the content to an output device. Output include email, fax and print.

For email fulfillment requests, Oracle One-to-One Fulfillment supports any email server that uses SMTP (Simple Mail Transfer Protocol) to transmit email messages. For fax fulfillment requests, Oracle One-to-One Fulfillment supports Captaris RightFax and Oracle Pasta printing.

One or more email servers can be associated with a Fulfillment Server.

Agent Groups

An agent is an employee who uses Oracle software. For agent using an E-Business Suite application to initiate a Fulfillment request, create groups of Fulfillment agents in the Oracle One-to-One Fulfillment Administration Console, and then assign them to a group.

Agent Resources

An employee must be set up as a resource using Oracle CRM Resource Manager. Each employee is assigned a resource ID to work with CRM applications. A valid resource ID is required to successfully record interactions in Customer Interaction History.

Email Tracking

Oracle One-to-One Fulfillment tracks the outcomes of email sent by a single request or a mass request. The Status page for the batch request shows the following statistics:

- Sent
 - Opened
 - Unopened
- Not Sent
 - Malformed
 - Do not contact
 - Other failures
- Bounced
- Unsubscribed
- Resubmitted Malformed

The Fulfillment Server dispatches the email requests to all the recipients of the email. If the Track Email feature is enabled, an invisible reference to an image on a Web server is appended to the email. When a recipient opens the email, the hidden image is downloaded.. However, the hidden image tag references a Java Server Page (JSP), which is responsible for updating the required tables with information that specifies which recipient had read the email. The JSP that handles the hidden image needs to be accessible outside the firewall.

The information that is required to decide which recipient has opened the email is present as a part of the query string in the URL. The JSP parses all the parameters from the query string and updates the database based on the request query string of the request URL. Also, the JSP is responsible for returning a substitute image instead of the one that is requested originally. The substitute image is a transparent image.

Templates

A template is a grouping of material that is useful but not required for distributing master documents and collateral. Collateral is static content.

An template is a package of content that is compiled by the Fulfillment Server. The fulfillment request identifies the template being used to compile the content.

MES Master Documents

A master document is a message sent from Oracle One-to-One Fulfillment. It is an HTML file or Zip file that contains the main content in a fulfillment item.

Examples of master documents include:

- Cover letters
- Service request notifications
- Past due notices
- Lease contracts
- Sales offers

A master document can be static or dynamic. A static master document does not have merge fields, so that each recipient gets the same content. A dynamic master document uses merge fields populated by an associated SQL query to combine information from the database (such as the name of the recipient or account information) into the content, so that each recipient gets personalized content. See *Queries*, page 1-12 for information about queries.

The way in which the E-Business Suite application uses a master document determines whether the master document is used as a standalone or as a part of a template. See *Templates*, page 1-10 for information about templates.

A master document can be created in an HTML editor or word processing application, but it must be saved as an HTML file. It can be uploaded as an HTML file or in a ZIP file.

When editing HTML files in a word processor, remove from the HTML code any propriety information that was inserted by the word processor, otherwise the server will not process the document. For example, saving a Microsoft Word document as HTML inserts Microsoft proprietary information in the <HTML> header tag, that is, a line in the header stating,

```
<META NAME = "Generator" CONTENT="Microsoft Word 97>
```

To remove such propriety information, use a generic text editor such as Microsoft Notepad or WordPad.

Oracle Content Manager Profile Options

Oracle Content Manager has two profile options relevant to Oracle One-to-One Fulfillment, data security and the New Content Items folder.

Data Security

The data security profile option enables and disables security in Oracle Content Manager. By default this option is disabled. To enable folder permissions, do the following:

- Set the security profile to Y.
- Set the necessary folder permissions for users of Oracle One-to-One Fulfillment.

New Content Items Folder

Oracle Content Manager stores new content items in a folder that you need to specify. If these profile options are not set, Content Items cannot be submitted properly.

The following table describes the data security profile options.

Oracle Content Manager Data Security Profiles

Profile Name	Category	Default Value	Comments
IBC: Use Access Control	Content Security	N	<p>Data security is disabled. Any user can create, update or delete a content item. All content items created are approved immediately.</p> <p>If set to Y, data security is enabled. A check is made to see if the user has appropriate permissions to perform the actions.</p>
IBC: Default Home Folder	Content Interaction	COMMON	<p>Specifies the folder where new content items will be created.</p> <p>Note: When data security is enabled, the user must have the required permission on this folder to create, update and delete items in this folder.</p> <p>Permissions can be granted from the Oracle Content Manager Foldering tab.</p>

Related Topics

Oracle Content Manager User Guide

Queries

A query is a SQL statement that extracts data from a database. A query determines:

- Which records meet the defined criteria.
- Which fields of those records the query will return.

The returned data is merged with a master document to create a customized email, fax, or printed mail for each recipient.

A single query can be associated with more than one master document if the merge

field names used in the document match the database column names or aliases.

Use the following syntax when creating a SQL query.

```
SELECT <column name> <alias>,  
  
<column name> <alias>,  
  
<column name> <alias>,  
  
<column name> <alias>  
FROM <table name>  
WHERE <column name> = :<bind variable>
```

When creating a query, note the following conditions:

- Aliases are optional.
- SQL statements typically end with a semicolon (;). However, in this case, the query engine appends the semicolon. Do NOT append a semicolon to the end of this query.
- Column name aliases follow the column names and appear before the comma or the next keyword. Use aliases as substitutes in master documents for the actual column names. Aliases are not required, but can be helpful.
- The table from which the columns are selected follows the FROM keyword.
- The bind variable specifies which row is pulled from the table to give the actual data for each merged document. A colon in front of a bind variable makes it a global variable. This enables the query to recognize it after the value is passed from the user interface. For example:

```
SELECT person_last_name last_name,  
  
party_id customer_number,  
  
creation_date,  
  
phone_number  
  
FROM jtf_contact_points_v  
  
WHERE person_first_name = :fname;
```

Merge fields should match exactly with the Master Document.

Data Source

Creating a data source file is an optional method of saving queried data to be used in a master document. A data source file is a text file that contains tab-separated merge fields that have been obtained by a query. Use the data source file to obtain the columns available from the query so that they can be used as merged fields. This can be helpful when creating master documents. The master document must be an HTML file.

If you enter the previous sample query as the query string, then the data source utility pulls the aliases rather than the column names when it creates the data source file.

The data source utility opens a window that shows the following column headers in a plain background window.

```
last_name      customer_number  creation_date    phone_number
```

Collateral

Collateral items, called deliverables in Oracle Marketing Online, are uploaded using the administrative console for Oracle Marketing Online. You cannot upload or view Oracle Marketing Online deliverables from the Oracle One-to-One Fulfillment administrative console.

Collateral items are sent as attachments to email output. Oracle Marketing needs to upload collateral items. Collateral may be included in fulfillment requests that are created using Oracle Scripting and Oracle TeleSales.

Fulfillment Request History

The Status tab > History sub tab displays requests that a Fulfillment Server is currently processing or has processed, successfully or unsuccessfully. You can view the details of a fulfillment request, re-submit a fulfillment request, and delete a history entry from the history list.

To view requests by outcome, select an outcome level from the View list. The listed outcomes are defined as follows:

- **All:** All Fulfillment requests are displayed in descending order of Request ID.
- **Cancelled:** Fulfillment Requests that have been canceled by the administrator or by the calling application.
- **Failed:** Fulfillment Requests that were not processed successfully. If the request contains more than one job, all jobs within the request failed.
- **Partially Successful:** Fulfillment Requests that contain one or more jobs that could not be processed successfully, and at least one job that was successfully processed. For example, if a customer is listed as "do not contact" for the particular media

channel, then a failure is logged for that job and the outcome of the request becomes "partially successful."

- **Stalled:** Fulfillment Requests that have stalled while being processed by the Fulfillment Server. A request can stall due to many reasons, but if it is stalled because of a database connectivity issue, the server will continue to periodically try to reconnect and, if successful, it will continue to process the request. Most stalled requests are caused by the fulfillment server stopping abnormally, in this case restart the server to continue processing the request.
- **Successful:** Fulfillment Requests that were processed successfully.

Fulfillment Request Status

The Status sub tab provides access to the status of fulfillment requests that have been placed in the request queue, but have not yet been retrieved by the Fulfillment Server. You can view the status of a fulfillment request, cancel a fulfillment request, resubmit the request, and delete a status entry from the status list.

Each status is defined as follows:

- **Submitted:** Fulfillment Requests that have been submitted to the Fulfillment Server for processing
- **Re-submitted:** Fulfillment Requests that have been re-submitted for processing
- **In Progress:** Fulfillment Requests that are currently being processed
- **Paused:** Fulfillment requests that have been partially processed and then paused
- **Resumed:** Fulfillment requests that were paused and subsequently resumed
- **Emails Sent, Writing Log:** This status indicates that the server has completed sending emails for all the request lines. A separate Java thread will now process the lines for logging Interaction History.

This flow uses bulk loading of Interaction History data. The concurrent program "Interaction History Bulk Processor" needs to be run to create Interaction History data for the request lines.

Physical Collateral

APIs enable E-Business Suite applications to pass physical collateral requests. In turn, One-to-One Fulfillment integrates with Order Management to execute these requests.

Unsubscribe Option

One-to-One Fulfillment has an unsubscribe feature for email fulfillment requests. The

email contains an embedded URL link to a Web page so that when users access the Web page, a validity check is performed and users have the option to unsubscribe.

The Unsubscribe option in the Fulfillment Server configuration causes the Fulfillment Server to insert a hypertext link into the content of email that will allow email recipients to modify their contact preferences.

For email requests, the outgoing email contains an embedded URL to a Web page. The URL leads to the Oracle Marketing Online Contact Preferences page, where the recipient can set contact channels. The format of the unsubscribe link is:

```
http://<host>:<port>/OA_HTML/<name of the  
jsp>?PAGE.OBJ.requestId=&PAGE.OBJ.objId=
```

The `request_id` is the numerical identifier for the fulfillment request. If the recipient elects to unsubscribe from email contact or all contact, then Oracle Marketing Online uses the request identifier to update Oracle One-to-One Fulfillment History with this information.

The `party_id` is the numerical identifier of the email recipient. The `party_id` is used when the `amsContactPreferencesMain.jsp` page is called to confirm that the party is valid, and that the appropriate Contact Preferences table within TCA can be modified when the customer unsubscribes. The party identifier is encrypted so that it is not visible to the email recipient. Oracle Marketing Online uses the encryption key in the JTO: Security: Merge Field Encryption Key profile option to decrypt the party identifier and enable access to the party's contact preference settings.

When the recipient accesses the Web page, the recipient can unsubscribe from email. This action is recorded in Oracle Customer Interaction History and, if email tracking is enabled, is reflected in the email statistics for the mass email fulfillment request.

To enable or disable the unsubscribe link, use the Unsubscribe box on the Email Server page for the Fulfillment Server. This option affects the email fulfillment requests that are processed by all email servers assigned to the Fulfillment Server. It cannot be set for an individual email server.

Resubmit

The One-to-One Fulfillment Administration Console has a Resubmit option in the Status tab > History sub tab. This option enables a request to be resubmitted by using the original data.

The One-to-One Fulfillment Administration Console provides a Resubmit option for jobs in the Jobs Details page. This option enables a job (that was sent to a single customer) to be resubmitted using the original data. The user is able to select an alternative delivery channel and contact address for the recipient.

Pause and Resume Request Processing

One-to-One Fulfillment has a pause and continue function that administrators can use

to pause an in-progress batch request, and later resume processing the request. This function is available in the Request Status page.

Deletion of Sent Fulfillment Requests

A Delete option is in the Status tab > History sub tab for deleting canceled, completed, and failed requests.

Master Document Editor

The One-to-One Fulfillment Master Document Editor simplifies the administrators' job of creating and publishing static and dynamic documents for communicating with customers.

The Master Document editor provides a fast and simple mechanism for creating master documents, creating and associating a query and uploading into the Management Encyclopedia System.

Text can be written in Rich Text Format, which supports multiple fonts, sizes and styles, and URL links and image insertion.

High Quality Printing

The high quality print Oracle solution enables the uploading of RTF or PDF Form documents, and offers E-Business Suite Applications the ability to initiate requests to print RTF or PDF Form documents or cover letters.

An Administrator creates an RTF document using Microsoft Word (or opens an existing document), uses the Form feature to insert the relevant personalized data (merge fields) in the appropriate positions in the letter or table, and then saves the document as an RTF file. When the document has been uploaded into the Content Repository, it is available for sending. Static PDF or RTF documents, such as attachments, can be printed in addition to the primary personalized document.

Oracle One-to-One Fulfillment also supports printing documents and collateral by way of Captaris RightFax.

Oracle Pasta

Oracle Pasta is an Oracle printing solution that uses an executable called FNDPSTAX. The FNDPSTAX tool submits print requests to the operating system printing functions:

- lp on UNIX and Linux
- print on Windows

To use the High-Quality Printing feature, you must first be able to print from the Fulfillment Server host machine using either lp or print, depending on your operating system. It is the responsibility of the local system administrator to configure the

operating system and network for printing.

XPDF pdftops

XPDF is an open source viewer for PDF files. It includes a PDF-to-Post Script conversion tool called pdftops. Pasta uses PDF to PS to convert the resulting PDF files into PS before spooling them to the printer. Download XPDF from <http://www.foolabs.com/xpdf/>.

XML Publisher

XML Publisher is an Oracle toolset that provides utilities to transform RTF and PDF documents into a format called XSL-FO (or eXtensible Stylesheet Language-Formatting Objects). The XSL-FO format is used as a base format that can be transformed back into RTF or PDF (and in later XML Publisher versions) HTML. XSL-FO is not intended for end users like marketers, survey creators, or other Fulfillment users who are responsible for creating Fulfillment Content. It is mentioned here for information purposes only. XML Publisher 4.3 contains the necessary RTF conversion utilities.

RTF and PDF

Due to the reliance on the XML Publisher utilities, only Fulfillment requests that contain strictly RTF or PDF documents can be printed through the High-Quality Print solution. Requests that contain any other file types can only be printed by the existing RightFax printing solution.

Dependencies and Integration Points

This chapter describes the system requirements and product dependencies for Oracle One-to-One Fulfillment. It also describes how to access the administrative console for Oracle One-to-One Fulfillment.

This chapter covers the following topics:

- System Requirements
- Mandatory Dependencies
- Conditional Dependencies
- Related Documentation
- Installing Oracle One-to-One Fulfillment
- Accessing Oracle One-to-One Fulfillment

System Requirements

This section describes the system requirements for Oracle One-to-One Fulfillment.

Topics include:

- Fulfillment Server, page 2-1
- Output Servers, page 2-2

Fulfillment Server

The Fulfillment Server processes the fulfillment requests that have been submitted by the business applications. All of the files required to use the Fulfillment Server are installed when Oracle Applications is installed. You can run the Fulfillment Server on the applications server. You can also run the Fulfillment Server from a separate computer that has access, by a network, to the Oracle Applications environment and database.

Output Servers

The Fulfillment Server sends fulfillment content to an output device. Output devices include email, fax, and print servers. In addition to the standard technology that is installed with Oracle Applications, the following technology is required to successfully implement Oracle One-to-One Fulfillment:

- For email output, any outgoing email server that complies with SMTP (Simple Mail Transfer Protocol), for example, UNIX Sendmail
- For fax or print output, Captaris RightFax Enterprise Server

See Integration Tasks, page 4-xii for information about integrating Captaris RightFax Enterprise Server with Oracle One-to-One Fulfillment.

Mandatory Dependencies

Mandatory dependencies are required for Oracle One-to-One Fulfillment to function. Oracle One-to-One Fulfillment relies on several products for data and functionality. The following products must be installed and implemented before you begin the implementation of Oracle One-to-One Fulfillment:

- Oracle Human Resource Management System, page 2-2
- Oracle Common Application Components, page 2-3
- Oracle Customer Interaction History, page 2-3
- Oracle Marketing Encyclopedia System, page 2-3

Required for use with Oracle Collections, Oracle Lease Management, Oracle TeleSales, Oracle TeleService, Oracle Scripting, and Oracle Student System.

- Oracle Content Manager, page 2-4
Required for use with Oracle Marketing.
- Oracle Marketing, page 2-4
Required for use with Oracle Marketing, Oracle Scripting, and Oracle TeleSales.
- Business Applications, page 2-4

Oracle Human Resource Management System

Oracle Human Resource Management System is used to create the employees (agents) that will use the business application to submit a fulfillment request. Fulfillment agent groups are created in the Oracle One-to-One Fulfillment administrative console and assigned to a Fulfillment Server.

For information about Oracle Human Resource Management System, see *Implementing Oracle HRMS* and *Using Oracle HRMS - The Fundamentals*.

Oracle Common Application Components

Certain business applications require that employees also be resources. The Resource Manager module in Oracle Common Application Components is used to import employees from Oracle Human Resource Management System as resources. Fulfillment agent groups are created in the Oracle One-to-One Fulfillment administrative console and assigned to a Fulfillment Server.

For information about Oracle Common Application Components, see *Oracle Common Application Components Implementation Guide* and *Oracle Common Application Components User Guide*.

Oracle Customer Interaction History

Oracle Customer Interaction History enables the creation of history records for fulfillment requests. You can use the Oracle One-to-One Fulfillment administrative console to set the interaction history outcomes and results for fulfillment requests that have been processed by the Fulfillment Server.

For information about Oracle Interaction History, see *Oracle Interaction History Implementation Guide*.

Oracle Marketing Encyclopedia System

Oracle Marketing Encyclopedia System stores master documents and queries that are used to specify fulfillment content. The master documents and queries stored in Oracle Marketing Encyclopedia System are used by the following business applications:

- Oracle Collections
- Oracle Lease Management
- Oracle TeleSales
- Oracle TeleService
- Oracle Scripting
- Oracle Student System

The Oracle One-to-One Fulfillment administrative console provides an interface for

- Creating master documents using a document editor and uploading those master documents to Oracle Marketing Encyclopedia System
- Uploading externally created master documents to Oracle Marketing Encyclopedia System

- Creating queries in Oracle Marketing Encyclopedia System

For information about Oracle Marketing Encyclopedia System, see the chapter *Overview of Oracle Marketing Encyclopedia System* in *Oracle iSupport Implementation and Administration Guide*.

Oracle Content Manager

Oracle Content Manager stores templates (similar to master documents in Oracle Marketing Encyclopedia System) and queries that are used to specify fulfillment content. The templates and queries stored in Oracle Content Manager are used by Oracle Marketing.

The Oracle One-to-One Fulfillment administrative console provides an interface for

- Creating templates using a document editor and uploading those templates to Oracle Content Manager
- Uploading externally created templates to Oracle Content Manager
- Creating queries in Oracle Content Manager

For information about Oracle Content Manager, see *Oracle Content Manager Implementation and Administration Guide*.

Oracle Marketing

Oracle Marketing is used to upload collateral items. Collateral may be included in fulfillment requests that are created using Oracle Marketing, Oracle Scripting, and Oracle TeleSales.

For more information, see *Oracle Marketing Implementation Guide* and *Oracle Marketing User Guide*.

Business Applications

Fulfillment requests originate from a business application. The following business applications integrate with Oracle One-to-One Fulfillment:

- Oracle Collections, page 2-5
- Oracle Lease Management, page 2-5
- Oracle Marketing, page 2-5
- Oracle Scripting, page 2-6
- Oracle Student System, page 2-6
- Oracle TeleSales, page 2-6

- Oracle TeleService, page 2-7

Oracle Collections

Oracle Collections is a debt recovery application. Agents use Oracle Collections to manage delinquent accounts and collect payments. Agents also use Oracle Collections to send correspondence, such as a request for payment or an account statement, to a party.

Oracle Collections uses Oracle One-to-One Fulfillment to process single or mass requests by email, fax, and print output channels. The content of the correspondence is stored in Oracle Marketing Encyclopedia System as reusable master documents. The master documents can be personalized using merge fields and queries.

Oracle Collections generates a fulfillment request when an automatic work item is generated or an agent chooses to send a master document. Oracle Collections also records the status of the fulfillment request in Oracle Customer Interaction History.

For information about Oracle Collections, see *Oracle Collections Implementation Guide* and *Oracle Collections User Guide*.

Oracle Lease Management

Oracle Lease Management is a asset management application. Agents use Oracle Lease Management to manage the leases for assets. Agents also use Oracle Lease Management to send correspondence, such as the terms of a lease, to a party.

Oracle Lease Management uses Oracle One-to-One Fulfillment to process single requests by email. The content of the correspondence is stored in Oracle Marketing Encyclopedia System as reusable master documents. The master documents can be personalized using merge fields and queries.

Oracle Lease Management generates a fulfillment request when an agent chooses to send a master document. Oracle Lease Management also records the status of the fulfillment request in Oracle Customer Interaction History.

For information about Oracle Lease Management, see *Oracle Lease Management Implementation Guide* and *Oracle Lease Management User's Guide*.

Oracle Marketing

Oracle Marketing is a campaign management application. Marketing professionals use Oracle Marketing to define, execute, and monitor marketing campaigns. Marketing professionals also use Oracle Marketing to send correspondence, such as an offer, to marketing targets.

Oracle Marketing uses Oracle One-to-One Fulfillment to process single or mass requests by email. The content of the correspondence is stored in Oracle Content Manager as reusable template (similar to master documents in Oracle Marketing Encyclopedia System). The templates can be personalized using merge fields and queries. Collateral items (also called deliverables) are uploaded using Oracle Marketing.

Oracle Marketing generates a fulfillment request when a user chooses to send a template. Oracle Marketing also records the status of the fulfillment request in Oracle Customer Interaction History.

For information about Oracle Marketing, see *Oracle Marketing Implementation Guide* and *Oracle Marketing User Guide*.

Oracle Scripting

Oracle Scripting is a script and survey management application. Survey administrators use Oracle Scripting to define, execute, and monitor surveys. Survey administrators also use Oracle Scripting to send correspondence, such as invitations and reminders, to survey targets.

Oracle Scripting uses Oracle One-to-One Fulfillment to process mass requests by email. The content of the correspondence is stored in Oracle Marketing Encyclopedia System as reusable master documents. The master documents can be personalized using merge fields and queries. Optional collateral items (also called deliverables) are uploaded using Oracle Marketing.

Oracle Scripting generates a fulfillment request when the survey campaign is deployed. Oracle Scripting also records the status of the fulfillment request in Oracle Customer Interaction History.

For information about Oracle Scripting, see *Oracle Scripting Implementation Guide* and *Oracle Scripting User Guide*.

Oracle Student System

Oracle Student System is a student information management application. School officials use Oracle Student System to manage the entire life cycle of a student from inquiry through graduation. School officials also use Oracle Student System to send correspondence, such as a notifications and statements, to a party.

Oracle Student System uses Oracle One-to-One Fulfillment to process single or mass requests by email, fax, and print output channels. The content of the correspondence is stored in Oracle Marketing Encyclopedia System as reusable master documents. The master documents can be personalized by using merge fields and queries.

Oracle Student System uses concurrent processes to generate a fulfillment requests. Oracle Student System also records the status of the fulfillment request in Oracle Customer Interaction History.

For information about Oracle Collections, see *Oracle Student System Implementation Guide* and *Oracle Student System User's Guide*.

Oracle TeleSales

Oracle TeleSales is a contact management application. Agents use Oracle TeleSales to manage inbound telesales and outbound telemarketing. Agents also use Oracle TeleSales to send correspondence, such as cover letters or collateral items, to a party.

Oracle TeleSales uses Oracle One-to-One Fulfillment to process single requests by email, fax, and print output channels. The content of the correspondence is stored in Oracle Marketing Encyclopedia System as reusable master documents. The master documents can be personalized using merge fields and queries. Collateral items (also called deliverables) are uploaded using Oracle Marketing.

Oracle TeleSales generates a fulfillment request when an agent chooses to send a master document or an electronic collateral item. Oracle TeleSales also records the status of the fulfillment request in Oracle Customer Interaction History.

For information about Oracle Collections, see *Oracle TeleSales Implementation Guide* and *Oracle TeleSales User Guide*.

Oracle TeleService

Oracle TeleService is a problem tracking application. Agents use Oracle TeleService to manage service requests. Agents also use Oracle TeleService to send correspondence, such as a status, to a party.

Oracle TeleService uses Oracle One-to-One Fulfillment to process single requests by email. The content of the correspondence is stored in Oracle Marketing Encyclopedia System as reusable master documents. The master documents can be personalized using merge fields and queries. Seeded templates are used to join master documents and collateral item in a single package.

Oracle TeleService generates a fulfillment request when an agent chooses to send a template. Oracle TeleService also records the status of the fulfillment request in Oracle Customer Interaction History.

For information about Oracle Collections, see *Oracle TeleService Implementation Guide* and *Oracle TeleService User Guide*.

Conditional Dependencies

Conditional dependencies provide desired functionality to Oracle One-to-One Fulfillment but are not required for Oracle One-to-One Fulfillment to function.

The following conditional dependencies apply to Oracle Interaction Center:

Oracle Marketing

Oracle Marketing can be used to allow parties to set their contact channels. The Unsubscribe option in the Fulfillment Server configuration causes the Fulfillment Server to insert a hypertext link into the content of email that will allow email recipients to modify their contact preferences.

For email requests, the outgoing email contains an embedded URL to a Web page. The URL leads to the Contact Preferences page in Oracle Marketing Online. The Contact Preferences page allows the recipient to set contact channels.

Note: Oracle Marketing is required to upload collateral items. Collateral may be included in fulfillment request that are created using Oracle Marketing, Oracle Scripting, and Oracle TeleSales.

Related Documentation

The latest product documentation is available on Oracle *MetaLink* at <http://metalink.oracle.com>.

Installing Oracle One-to-One Fulfillment

To install Release 12, you must use Oracle Applications Rapid Install.

The Rapid Install is intended for customers who are installing Oracle Applications for the first time or upgrading to Release 12. It contains the family packs or product minipacks for all products in Oracle Applications. The Rapid Install is provided on CD-ROMs and is available from Oracle Store at <http://oraclestore.oracle.com>.

For information about installing Oracle Applications using Rapid Install, see *Oracle Applications Installation Guide: Using Rapid Install*. For information about upgrading Oracle Applications using Rapid Install, see *Oracle Applications Upgrade Guide: Release 11i to Release 12*.

Accessing Oracle One-to-One Fulfillment

This section includes the following topics:

- User Accounts, page 2-8
- Responsibilities, page 2-9

User Accounts

An application user is an authorized user of Oracle Applications who is uniquely identified by a user name. After the user account has been defined, the application user can sign on to Oracle Applications at the CRM Home Page, Personal Home Page, or Forms login.

Note: Oracle Applications is installed with a system-defined user name and password.

- User name: sysadmin
- Password: sysadmin

To sign on to Oracle Applications, application users enter a user name and a password. The password, which is assigned by the system administrator, is temporary. When application users sign on for the first time, they receive a prompt to change the password. Access to specific functionality and data are determined by the responsibilities assigned to the user account.

Responsibilities

A system administrator assigns one or more responsibilities to an application user. A responsibility is a level of authority that allows a user to access specific functionality and data in Oracle Applications. Oracle Applications is installed with predefined responsibilities. A system administrator can modify a predefined responsibility or create custom responsibilities.

The following table describes the predefined responsibilities that are used to implement Oracle One-to-One Fulfillment.

Responsibility	Function	Interface
CRM HTML Administration	Assign the JTF_ADMIN_FM role to the user account.	HTML
CRM Resource Manager	Create an employee (if Oracle Human Resource Management System is not installed). Import an employee as a resource.	Forms
One-to-One Fulfillment Administrator	Define Fulfillment Servers and output servers. Create a fulfillment agent group. Associate a fulfillment agent group with a Fulfillment Server. Create queries. Create and/or upload master documents. Create content types and cover letters (content items). View request statuses and history.	HTML
Fulfillment Test Harness	Verify single request and mass request. Create fulfillment APIs and Fulfillment Server setup.	HTML

Responsibility	Function	Interface
System Administrator	Create a user for administering Oracle One-to-One Fulfillment. Set values for profile options.	Forms
HRMS Manager, for example US HRMS Manager	Create an employee (Only if Oracle Human Resource Management System is installed).	Forms

In the Forms interface, if an application user has only one responsibility, then the related menu or application (if there is only one function in the menu) appears after the user signs on. If an application user has more than one responsibility, then a list of available responsibilities appears after the user signs on. To switch responsibilities, choose Switch Responsibility from the File menu.

In the HTML interface, an application user must select a default responsibility (even if the user has only one responsibility). The next time the application user signs on, the tabs related to the default responsibility appear. To switch responsibilities, go to Navigation Preferences in your profile (Profile icon). In the Switch Responsibilities section, select another responsibility from the Current Responsibility list.

Implementation Tasks

This chapter provides a high-level overview of the implementation task sequence, and references to procedures that you will do during the implementation.

This chapter covers the following topics:

- Implementation Task Sequence
- Defining an Administrator
- Creating an Administrator User Account
- Assigning a Role to a User Account
- Defining an Agent
- Creating an Employee
- Creating a User Account for an Agent
- Assigning or Importing a Resource for an Employee
- Configuring Servers
- Creating a Fulfillment Server
- Defining an Email Server
- Defining a Fax Server
- Adding an Available Print Server
- Creating a New Print Server for RightFax Printing
- Implementing a Fulfillment Server
- Creating a Remote Command Script for the Fulfillment Server

Implementation Task Sequence

Implement Oracle One-to-One Fulfillment in the sequence outlined in the following table. The Number column indicates the step order. The Required column indicates

whether a step is required. The Description column describes a high-level step and, where applicable, provides a reference to a more detailed topic in this document. The Responsibility column indicates the Oracle Applications user account responsibility that is required to complete the step.

Sequence of Implementation Procedures

Step	Required	Description	Responsibility
1	Required	<p>Create an Oracle One-to-One Fulfillment administrator.</p> <p>Use this user account to perform the remaining steps.</p> <p>See: "Defining an Administrator", page 3-1</p>	System Administrator
2	For Collections, Lease Management, Marketing, Scripting, Student System, TeleSales, and TeleService	<p>Add email output channels.</p>	One-to-One Fulfillment Administrator
3	Required	<p>Set up profile options.</p> <p>See: "Fulfillment Profile Options", page C-1</p>	System Administrator
4	Required	<p>Create agent users for the business application.</p> <p>See the implementation guide of the business application.</p>	Varies

Step	Required	Description	Responsibility
5	Required	<p>Create a fulfillment agent group.</p> <ol style="list-style-type: none"> 1. Define an agent group. 2. Associate the group with a Fulfillment Server. 3. Assign agents. <p>See: "Managing Fulfillment Agent Groups", page 6-11</p>	One-to-One Fulfillment Administrator
6	For Marketing, and TeleSales (Optional for Scripting)	<p>Create collateral items and upload them using Oracle Marketing.</p> <p>See: <i>Oracle Marketing User Guide</i></p>	Oracle Marketing Administrator
7	For Collections, Lease Management, Scripting, Student Systems, TeleSales, and TeleService	<p>Create master documents and upload them into Oracle Marketing Encyclopedia System.</p> <p>See: "Managing Oracle Marketing Encyclopedia System Master Documents", page 6-22</p>	One-to-One Fulfillment Administrator

Step	Required	Description	Responsibility
8	For dynamic content in MES master documents	<p>Create a query in Oracle Marketing Encyclopedia System.</p> <p>See: "Managing Oracle Marketing Encyclopedia System Queries", page 6-28</p>	One-to-One Fulfillment Administrator
9	For TeleService, and Scripting (Survey component)	<p>Create a template.</p> <ul style="list-style-type: none"> Associate one or more master documents with the fulfillment template Associate one or more collateral items with the fulfillment template <p>See: "Managing Oracle Marketing Encyclopedia System Templates", page 6-15</p>	One-to-One Fulfillment Administrator
10	For dynamic content in OCM templates	<p>Create a query in Oracle Content Manager.</p>	One-to-One Fulfillment Administrator
11	For RightFax fax and print output channels	<p>Integrate Captaris RightFax with Oracle One-to-One Fulfillment.</p> <p>See: "Integrating with Captaris RightFax", page 4-1</p>	Not applicable
12	For Marketing	<p>Upload content blocks and images.</p>	

Step	Required	Description	Responsibility
13	For Marketing	Create a content type.	
14	For Marketing	Create a cover letter and select and upload attachments. See: "Creating Cover Letters", page 6-39	
15	For Pasta (Oracle) print output	Set up Pasta XPDF.	

Related Topics

- "Defining an Administrator", page 3-1

Defining an Administrator

Do the procedures in the following table to create a user account for administering Oracle One-to-One Fulfillment. The Step column indicates the step order. The Required column indicates whether a step is required. The Description column describes a high-level step and, where applicable, provides a reference to a more detailed topic in this document. The Responsibility column indicates the Oracle Applications user account responsibility required to complete the step.

Step	Required?	Description	Responsibility
1	Yes	Create an Oracle Applications user account and assign responsibilities. See: "Creating an Administrator User Account", page 3-1	System Administration

Step	Required?	Description	Responsibility
2	Yes	<p>Assign the JTF_FM_ADMIN role to the user account.</p> <p>See: "Assigning a Role to a User Account", page 3-1</p>	CRM HTML Administration

Creating an Administrator User Account

Use the following procedure to create a user account for administering Oracle One-to-One Fulfillment.

Login

Self-Service Login URL

Responsibility

System Administrator

Prerequisites

None

Steps

1. In the Navigator window, on the Functions tab, choose **Security >User >Define**.
The User window appears.
Use the following guidelines to define Oracle Applications user names:
 - Use only one word.
 - Use only alphanumeric characters ('A' through 'Z', and '0' through '9').
 - Use only the set of characters that your operating system supports for filenames.
2. In the User Name field, enter the name of the user.

The password is temporary. When a user signs on to Oracle Applications for the first time, the message "Your password has expired" appears and the user is prompted to set a new password.

Use the following guidelines to define Oracle Applications passwords:

- Use at least five characters and no more than 100 characters.
 - Use only alphanumeric characters (A through Z, and 0 through 9).
3. In the Password field, enter the password for the user account and then press **Tab**.
The cursor remains in the Password field.
 4. Enter the password again to verify it.
 5. To use this account to submit concurrent programs for Oracle One-to-One Fulfillment, select an employee from the Person field to associate with this user account.
 6. In the Responsibilities tab, add the following responsibilities:
 - One-to-One Fulfillment Administrator
 - Fulfillment Test Harness

After you save the user record, it is not possible to delete an assigned responsibility. Oracle Applications maintain audit data for assigned responsibilities.

To deactivate an assigned responsibility, set the effective end date (in the Effective Dates - To field) of the assigned responsibility to the current date. To activate an assigned responsibility, clear or reset the effective end date.
 7. From the **File** menu, choose **Save**.
You may close the Users window.

Assigning a Role to a User Account

The JTF_FM_ADMIN role provides access to the tabs in the Oracle One-to-One Fulfillment administration console. You need the JTF_FM_ADMIN role *in addition to* the One-to-One Fulfillment Administrator responsibility. See Creating an Administrator User Account, page 3-1 for information about assigning a responsibility to a user account.

Use the following procedure to assign the JTF_FM_ADMIN role to a user account.

Login

HTML Login URL

Responsibility

CRM HTML Administration

Prerequisites

The CRM HTML Administration user must have the following roles:

- JTF_SYSTEM_ADMIN_ROLE, to access user maintenance functions
- JTF_FM_ADMIN, to assign the role to another user

Steps

1. Click the Users tab > Registration sub tab.
2. Click the User Maintenance side panel link.
The Users page appears.
3. In the Enter Search Criteria fields, select Username, enter the user name, and then click **Go**.
The matching user names appear.
4. Click a user name link.
The User Details page appears.
5. Click **Roles**.
The User-Role Mapping page appears.
6. Move JTF_FM_ADMIN from the Available Roles list to the Assigned Roles list.
7. Click **Update**.
The User-Role Mapping page refreshes.

Defining an Agent

To use Oracle One-to-One Fulfillment, agents must be defined as "users" and assigned to a Fulfillment agent group. Use the Forms application to do define an agent in the following sequence:

1. Create an employee.
2. Create an application user, and assign Responsibilities to the user.
3. Create a Resource. A Resource is required to record data in Interaction History.

Creating an Employee

You can create an employee in two ways:

- "Creating an Employee in Oracle Human Resource Management Systems", page 3-9
- "Creating an Employee in Resource Manager", page 3-10

Note: If your system has Oracle Human Resource Management (HRMS), then you must use it create employees. If your system does not have Oracle HRMS, then use the Oracle Resource Manager to create employees.

Creating an Employee in Oracle Human Resource Management Systems

Use this procedure to create an employee in Oracle Human Resource Management Systems.

Login

Self-Service Login URL

Responsibility

System Administrator

Prerequisites

Create at least one administrator.

Steps

1. In the Navigator window on the Functions tab, select **People >Enter and Maintain**.
The Find Person window appears.
2. Click **New**.

The People window appears.

3. In the Last field, type the employee's last name.
4. From the Gender list, select Male or Female.
5. From the Type list, select **Employee**.
6. Other fields may be required, depending on the requirements of your business.
7. Select **File > Save**.

Creating an Employee in Resource Manager

If Oracle Human Resource Manager is not installed on your system, then use the following procedure to create an employee. This procedure populates the same database tables as the People window in Oracle Human Resource Management Systems.

Login

Self-Service Login URL

Responsibility

System Administrator

Prerequisites

Create at least one administrator.

Steps

1. In the Navigator window on the Functions tab, select **Maintain Employee > Employee**.
The Find Person window appears.
2. Click **New**.
The People window appears.
3. In the Last field, type the employee's last name.
4. From the Gender list, select Male or Female.
5. From the Type list, select **Employee**.

6. Other fields may be required, depending on the requirements of your business.
7. Select **File >Save**.

Creating a User Account for an Agent

Use the following procedure to create a user account for agents who will use Oracle Applications.

Login

Self-Service Login URL

Responsibility

System Administrator

Prerequisites

Create an employee. See "Creating an Employee", page 3-9.

Steps

1. In the Navigator window, on the Functions tab, choose **Security >User >Define**.
The User window appears.

Use the following guidelines to define Oracle Applications user names:

- Use only one word.
- Use only alphanumeric characters ('A' through 'Z', and '0' through '9').
- Use only the set of characters that your operating system supports for filenames.

2. In the User Name field, enter the name of the user.

The password is temporary. When a user signs on to Oracle Applications for the first time, the message "Your password has expired" appears and the user is prompted to set a new password.

Use the following guidelines to define Oracle Applications passwords:

- Use at least five characters and no more than 100 characters.
- Use only alphanumeric characters (A through Z, and 0 through 9).

3. In the Password field, enter the password for the user account and then press **Tab**.
The cursor remains in the Password field.
4. Enter the password again to verify it.
5. To use this account to submit concurrent programs for Oracle One-to-One Fulfillment, select an employee from the Person field to associate with this user account.
6. In the Responsibilities tab, assign the appropriate responsibilities for the eBusiness Suite Applications that the agent will use. No specific Fulfillment Responsibility is required.

After you save the user record, it is not possible to delete an assigned responsibility. Oracle Applications maintain audit data for assigned responsibilities.

To deactivate an assigned responsibility, set the effective end date (in the Effective Dates - To field) of the assigned responsibility to the current date. To activate an assigned responsibility, clear or reset the effective end date.

7. From the **File** menu, choose **Save**.
You may close the Users window.

Assigning or Importing a Resource for an Employee

A resource is required to record data in Interaction History. Use the following procedure to create a resource for an employee.

Login

Self-Service Login URL

Responsibility

System Administrator

Prerequisites

Create the employee. See *Creating an Employee*, page 3-9.

Steps

1. In the Navigator window on the Functions tab, select **Maintain Resources >Import Resources**.
The Selection Criteria window appears.

2. From the Resource Category list, select **Employee**.
3. Enter any available selection criteria, such as the name of the employee.
4. Click **Search**.

The Search Results lists the names of all employees who match the search criteria.
5. Clear the Select option of any listed employees for whom you do not want to create a resource.
6. Click Create **Resource**.

The Default Values window appears. Optionally, you can add or modify this information later.
7. To accept the defaults, click **OK**.

The Selected Resources window appears. The Comments field indicates whether the resource is a new record, a duplicate record, or a duplicate record with a new role definition. The Select options are selected automatically.
8. Clear the Select options of any employees whom you do not want to save as a resource.
9. Click **Save Resources**.

A transaction number appears in the Transaction Number field. The transaction number is associated with each resource that was created during this transaction. More than one resource can have the same transaction number.
10. To view the details about a resource, select the resource and then click **Details**.

The Resource window appears.
11. Verify the following information:
 - The name of the resource in the Name field.
 - The name of the employee in the Source Name field.
 - The user name in the User Name field.
12. Assign any roles and groups that the eBusiness Application may require. Fulfillment agents do not require specific roles and groups.

Configuring Servers

Use the Server sub tab to configure Fulfillment Servers and output servers.

Tasks

You can perform the following tasks:

- "Creating a Fulfillment Server", page 3-14
- "Defining an Email Server", page 3-16
- "Defining a Fax Server", page 3-18
- "Adding an Available Print Server", page 3-19
- "Creating a New Print Server for RightFax Printing", page 3-20

Concepts

- Fulfillment Server, page 1-8
- Email Tracking, page 1-9
- Fulfillment Channels, page 1-9

Creating a Fulfillment Server

Use this procedure to create and configure a Fulfillment Server.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Install and implement a fulfillment output server.

Steps

1. Select the Server tab.
The Servers page appears.
2. Click **Create**.

The Server Details page appears.

3. Enter the details about the server.
 1. In the Server Name field, type a name for the server.

The name will appear in the list of servers on the Servers page. You cannot change this field after the server is saved.
 2. Optionally, from the Start Time lists, select the hour and minute for the Fulfillment Server to start polling the fulfillment request queues in the database. It does not physically start the server. If running, the server will continue to process retrieved fulfillment requests even when it is not polling the database.
 3. Optionally, from the Shutdown Time lists, select the hour and minute of the time at which the Fulfillment Server stops polling the fulfillment request queues in the database. It does not physically stop the server. If running, the server will continue to process retrieved fulfillment requests even when it is not polling the database.
 4. Optionally, in the Description field, enter a description of the server.

This field is for informational purposes only.
4. In the Interaction History group, select the outcome and result that will be recorded in Oracle Interaction History for fulfillment requests processed by the server.
 1. From the Request Outcome list, select the outcome that you want to associated with a fulfillment request that has been retrieved from the request queue by the Fulfillment Server.
 2. From the Request Success Result list, select the result that you want to associate with a fulfillment request that has been successfully processed by the Fulfillment Server.
 3. From the Request Failure Result list, select the result that you want to associate with a fulfillment request that has not been successfully processed by the Fulfillment Server.
5. Click **Continue**.

The server is saved and the Email Servers page appears.
6. To define an email server, see *Defining an Email Server*, page 3-16.

Defining an Email Server

To send fulfillment items by email, set the email server profiles for the Fulfillment Server and assign one or more email servers to the Fulfillment Server.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Set the email server profiles for the Fulfillment Server.

Steps

1. If you want the Fulfillment Server to insert a hypertext link into the email content that will allow recipients to access their contact preferences in Oracle Marketing, then select **Unsubscribe**.
2. Enter the HTML Unsubscribe text and URL. This text and link will appear at the end of the HTML-format email message.
3. Enter the Text Unsubscribe text and URL. This text and link will appear at the end of the text-format email message.
4. If you want the Fulfillment Server to track statistics for email requests, then select **Track Email**.
5. Enter the email Tracking URL. This is the URL of the JSP that tracks opened email messages.
6. In the Return Address field, type the "reply to" email address that will be used in the email messages sent by this email server.

The business application can override this address by providing a reply email address, otherwise the Fulfillment Server will use this address. If the recipient replies to the email, then the reply will be sent to this address.
7. In the From Address field, type the originating email address for emails processed using the Fulfillment Server.

The business application can override this address by providing an originating

email address, otherwise the Fulfillment Server will use this address.

This value in this field, preceded by the value in the Display Name field (*see next step*), will appear in the From field in the email.

8. In the Display Name field, type the name that you want to precede the From Address field value in the From field of emails processed using the Fulfillment server.
9. In the Summary of Email Server area, click **Add Email Server**.
The Create Email Server page appears.
10. In the Email Server Name field, type a descriptive name for the email server.
This name will appear in the list of email servers on the Email Servers page. You cannot change this field after the definition is saved.
11. In the Outgoing Mail Server field, type the name of the SMTP server.
This is the host name of the SMTP server.
12. In the Port Number field, type the port number that the Fulfillment Server will use to communicate with the email server.
13. Optionally, in the Description field, type a description of the email server.
This field is for informational purposes only.
14. To enable the bounceback email feature, in the Bounceback Management group, enter the following parameters:
 1. Enable Bounceback: Select to enable bounceback processing.
 2. IMAP Host: Type the IMAP Host Name.
 3. IMAP Port: Type the IMAP Port Number.
 4. Account Name: Type the name of the bounceback email account.
 5. Account Password: Type the bounceback account password.
 6. Run Frequency: Type the frequency with which the process should repeatedly check for bounceback messages in the email account, and then from the list select the frequency in Hours or Minutes.
 7. Return Path: Type the email address to send the bounceback notices. The Return Path overrides the default return path.
15. Click **Continue**.

The Fax Servers page appears.

16. To define a fax server, see Defining a Fax Server, page 3-18.

Defining a Fax Server

To send fulfillment items by fax, assign one or more fax servers to the Fulfillment Server.

Note: Oracle One-to-One Fulfillment supports only Captaris RightFax for fax server output.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Set the email server profiles for the Fulfillment Server.

Steps

1. Click **Add Fax Server**.
The Create Fax Server page appears.
2. In the Fax Server Name field, type a descriptive name for the fax server (for example, Fax 4th Fl Room 1234).
This name will appear in the list of fax servers on the Fax Servers page. You cannot change this field after the definition is saved.
3. In the Fax Server field, type the server name (for example, //faxserver.domainname.com).
4. In the Prefix field, type the prefix number that must be dialed before the fax number is dialed (for example, 9).
5. In the Return Fax Number field, type the fax number that should be used for return faxes.

6. Optionally, in the Description field, type a description of the fax server
This field is for informational purposes only.
7. Click **Continue**.
The Printers page appears.
8. To send fulfillment items by printer, assign one or more print servers to the Fulfillment Server. Do one of the following:
 - To add an available print server, see Adding an Available Print Server, page 3-19.
 - To create a new print server, see Creating a New Print Server for RightFax Printing, page 3-20.

Adding an Available Print Server

Use the following procedure to add an existing print server to the Fulfillment Server.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Set the email server profiles for the Fulfillment Server.

Steps

1. In the Print Servers page, from the Available Print Servers list, select a print server.
2. Click **Add Print Server**.
The page refreshes. The selected print server appears in the list of print servers.
3. Click **Update** to save.
4. Click **Done**.

Creating a New Print Server for RightFax Printing

Use the following procedure to create a new print server and add it to the Fulfillment Server.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Set the email server profiles for the Fulfillment Server.

Steps

1. In the Print Servers page, click **Create New Print Server**.
The Create Print Server page appears.
2. In the Print Server Name field, type a descriptive name for the print server (for example, RightFax Server).
This name will appear in the list of print servers on the Printers page. It can be different from the actual printer name or alias. You cannot change this field after the definition is saved.
3. In the Connection String field, for a RightFax Print Server, enter the RightFax server's URL.
4. From the Server Type list, select **RightFax**.
5. Click **Apply**.
The Print Servers page appears.

Implementing a Fulfillment Server

Use the following procedures to implement a server for Oracle One-to-One Fulfillment.

- "Creating a Remote Command Script for the Fulfillment Server", page 3-21

Creating a Remote Command Script for the Fulfillment Server

The remote command script allows you to interact with the Fulfillment Server without the need to maintain the original command prompt used to start the server. When you want to issue a command, you type the name of the remote command script and the command.

The Fulfillment Server is configured in the startup script to monitor a port for remote commands. The port is specified in both the startup script for the server and the remote command script. The remote command script runs on the same computer as the Fulfillment Server or on a computer that is on the same network as the Fulfillment Server.

When a command is sent through the remote command script, the computer that issued the command waits for a response and terminates activity after it receives the response. To prevent unnecessary workload for the Fulfillment Server, the computer that issues the remote command only retains a connection to the Fulfillment Server for the duration of the remote command.

Use the following procedure to create a remote command script and operate a running Fulfillment Server by remote command.

Login

Not applicable

Responsibility

Not Applicable

Prerequisites

Identify an available port for sending commands to the Fulfillment Server by the startup script. The port must be specified in the remote command script and the startup script. For assistance, contact your Oracle Applications system administrator or database administrator.

Steps

1. Stop (use the server parameter "kill") the Fulfillment Server if it is running.
2. Use a text editor, such as Microsoft Notepad, to add the engine.CommandPort parameter to the startup script.

The engine.CommandPort flag is used to identify the port number, on the computer that is hosting the Fulfillment Server, that the Fulfillment Server will monitor for remote commands.

In a UNIX startup script, add:

```
-Engine.CommandPort=<port_number> /
```

In a DOS startup script, add:

```
-Engine.CommandPort=<port_number>
```

3. Use a text editor to create a separate text file that will act as the remote command script.

Every remote command script must have, at a minimum, the following:

- Reference to the executable Java class that runs the Remote Monitor, an internal monitor in the Fulfillment Server
- A port for sending commands to the Fulfillment Server

Optionally, if the remote command script will reside on a computer other than the computer that is hosting the Fulfillment Server, then you must also specify the name of the computer.

See Guidelines, page 3-22 for more information.

Note: Always check for syntax errors, such as misspellings and extra spaces.

4. If the remote command script is for UNIX (a shell script), save the file with a *.sh extension.
5. If the remote command script is for DOS (a batch file), save the file with a *.bat extension.
6. Start the Fulfillment Server (run the startup script).
7. At a different command prompt, send a command using the remote command script.

For example, enter: <remote_command_script_name> <command>

8. Verify that the Fulfillment Server received the command.

Guidelines

The following is an example of a UNIX shell script for issuing commands to a Fulfillment Server that is running in the background.

```
#!/bin/csh
```

```
java oracle.apps.jtf.fm.engine.remote.RemoteCommand -port <port_number>  
[-host <remote_hostname>] $*
```

The following is an example of a DOS batch file for issuing commands to a Fulfillment Server that is running in the background.

```
java oracle.apps.jtf.fm.engine.remote.RemoteCommand -port <port_number>
[-host <remote_hostname>] %*
```

The value of the port parameter must match the value of the engine.CommandPort parameter in the startup script. If the remote command script is on a computer other than the computer that is hosting the Fulfillment Server, then specify the name of the computer in the host parameter.

The remote command also works from a command prompt with the following command:

```
java oracle.apps.jtf.fm.engine.remote.RemoteCommand -port <port_number>
[-host <remote hostname>] <command>
```

Integration Tasks

The following sections explain how to integrate Oracle One-to-One Fulfillment with other Oracle products and third-party products.

This chapter covers the following topics:

- Integrating with Captaris RightFax
- High Level Integration Steps
- Sending a Fax Fulfillment Request by RightFax
- Sending a Print Fulfillment Request by RightFax

Integrating with Captaris RightFax

Oracle One-to-One Fulfillment supports only the third-party application Captaris RightFax for sending collateral by fax. Use the following procedures to integrate these two applications.

- High Level Integration Steps, page 4-1
- Sending a Fax Fulfillment Request by RightFax, page 4-1
- Sending a Print Fulfillment Request by RightFax, page 4-1

High Level Integration Steps

Integrate Oracle One-to-One Fulfillment with RightFax in the following sequence:

1. Install RightFax Enterprise Server.
2. Install fax board driver from RightFax fax board driver CD-ROM.
3. Install RightFax Integration Module.

4. Install RightFax Oracle CRM Connector.
5. Install RightFax XML Interface.
6. Install RightFax Java API Interface.
7. Configure IIS to create virtual directory to support rfwebcon.dll.
8. If the fulfillment server is running, use the server start/stop script to shutdown the server.
9. Download the RightFax zip file from the following URL:
<http://www.captaris.com/support/downloads/rightfax/9/index.html#Oracle>
The file Oracle_Patch.zip contains java API from Captaris and is required to be in the CLASSPATH for the One-to-One Fulfillment server in order to process a fax or print request successfully.
The administrator should copy this zip file to a location where the Fulfillment server is able to access it. The recommended location is \$OA_JAVA/3rdparty directory.
The fully qualified path to Oracle_Patch.zip should be appended to the Fulfillment server CLASSPATH. Verify that there is no reference to the older versions of RightFax zip file RFJavaInt.zip in the Fulfillment server CLASSPATH.
10. Configuration change is now complete. Use the server start/stop script to start the server. One-to-One Fulfillment server is now ready to process fax requests.

Sending a Fax Fulfillment Request by RightFax

The design of the Fulfillment Server allows the user to define a fax solution based on a static API call. This fax solution, written in Java, is called from the Server as specified in the startup script. If the class exists, the Fulfillment Server loads the specified class file and associated class files, with all supporting classes, in the classpath. The Fulfillment Server then matches the class name in the startup property definition `-Dengine.FaxEnabler=your.fax.Enabler` with the startup script. The class `your.fax.Enabler` is the user-defined fax solution. To use this class with the Fulfillment Server, the class must meet two requirements:

- Have a default constructor without parameters.
- Implement the interface `oracle.apps.jtf.fm.engine.disp.FaxEnabler` (available in the `apps.zip` file).

The `oracle.apps.jtf.fm.engine.FaxEnabler` interface has one method,
`public String faxContents(String Sender,`

```
String SenderFax,  
  
String SenderPhone,  
  
String SenderCompany,  
  
String Recipient,  
  
String RecipientFax,  
  
String RecipientPhone,  
  
String RecipientCompany,  
  
String Message,  
  
HtmlContent[] Contents)  
  
throws EnablerFailedException;
```

- Sender is the person who is sending the fax. This should be defined as an optional field that can take the empty string "" as an argument value.
- SenderFax is the fax number of the person sending the fax. This should be defined as an optional field that can take the empty string "" as an argument value.
- SenderPhone is the phone number of the person sending the fax. This should be defined as an optional field that can take the empty string "" as an argument value.
- SenderCompany is the company of the person sending the fax. This should be defined as an optional field that can take the empty string "" as an argument value.
- Recipient is the person who will be receiving the fax. This should be defined as an optional field that can take the empty string "" as an argument value.
- RecipientFax is the fax number that will be dialed to send the fax. This parameter must have a value.
- RecipientPhone is the phone number of the person who is to receive the fax. This should be defined as an optional field that can take the empty string "" as an argument value.
- RecipientCompany is the company of the person who is to receive the fax. This should be defined as an optional field that can take the empty string "" as an argument value.
- Message is the message that appears on the cover sheet of the fax being sent. This

should be defined as an optional field that can take the empty string "" as an argument value.

- Contents are the actual contents to fax.

Sending a Print Fulfillment Request by RightFax

The design of the Fulfillment Server allows the user to define a print solution that is based on a static API call. This print solution, written in Java, will be called from the Server as specified in the startup script. The Server will load the specified class file and associated class files (assuming that the class exists), with all supporting classes, in the classpath. The Server recognizes the class name from the startup property definition '-Engine.PrintEnabler=your.print.Enabler' in the Server startup script; the class 'your.print.Enabler' being the user-defined print solution. To use with the Fulfillment Server, this class must meet two requirements:

- It must have a default constructor; one that has no parameters.
- It must implement the interface 'oracle.apps.jtf.fm.engine.disp.PrintEnabler' (available in the apps.zip file).

The 'oracle.apps.jtf.fm.engine.PrintEnabler' interface has one method:

```
public String printContents(String PrinterName, HtmlContent[] Contents)
throws EnablerFailedException;
```

PrinterName is the name/alias of the Printer to send the print job to.

Contents are the actual contents that are to be printed.

Diagnostics and Troubleshooting

This chapter describes how to diagnose and fix issues that might arise during the implementation and usage of Oracle One-to-One Fulfillment.

This chapter covers the following topics:

- Introduction
- Common Error Messages
- Out of Memory Error
- `java.lang.NullPointerException` or Blank Screen
- Fulfillment Server Error Messages and Debug Flags
- Common Fulfillment Server Errors
- Fulfillment Server Debug Flags
- Stopping and Restarting the Fulfillment Server
- Email Issues
- `java.io.IOException: 500 5.5.1 Command Unrecognized`
- Characters Appear as '?' in Body of Email
- Frequently Asked Questions
- Uploading an RTF Document Fails in Oracle Content Manager
- Content Types Not Visible
- Cannot Update Existing Content Type
- Log Directory Path
- TeleSales Java Exception
- `java.lang.IllegalArgumentException`
- `DoNotContactException`
- Commands in Background Mode

- Viewing Request Processing
- "Initializing Processor" Error
- Merge Field Errors
- Unable to Use Merge Field Delimiters « and »
- Table Rows and Merge Fields
- Server id and User id in create_fulfillment API Call
- Sending a Request Through Multiple Media Types
- Debugging
- PL/SQL Debugging
- Server Debugging
- Troubleshooting the RightFax Integration

Introduction

Use the following information to diagnose and fix issues you might encounter when implementing and using Oracle One-to-One Fulfillment.

- Common Error Messages, page 5-2
- Fulfillment Server Error Messages and Debug Flags, page 1-8
- Email Issues, page 5-7
- Frequently Asked Questions, page 5-8
- Troubleshooting the RightFax Integration, page 5-24

Common Error Messages

Refer to the following common error messages for instructions on correcting errors.

Out of Memory Error

An out of memory error is received when running JTFFMCTL.SH.

Solution

Reset JTF_FM_MAX_PROCESSES to 9 or less. The server is designed to run a maximum of nine processes.

java.lang.NullPointerException or Blank Screen

If you receive the following exception:

- java.lang.NullPointerException
or
- You are able to send collateral using the Fulfillment Manager, but the system stops and the monitor displays a blank screen. In Fulfillment Manager, the request shows as completing successfully. The email is sent, but the forms do not return a Collateral Sent message or return to a Java Server Page.

Solution

1. Set the profile JTF_FM_DONE_DEST to a Java Server Page to which you want to return (such as CntctSumryMain.jsp).
2. Restart the Apache server.

Fulfillment Server Error Messages and Debug Flags

Refer to the following topics for descriptions of Fulfillment Server error messages and debug flags.

- Common Fulfillment Server Errors, page 5-1
- Fulfillment Server Debug Flags, page 5-1
- Stopping and Restarting the Fulfillment Server, page 5-1

Common Fulfillment Server Errors

The following table lists errors that the Fulfillment server most commonly writes to the logs. For further information regarding each error, see the exception trace files.

Error	Description
tried to execute bind query >	Unable to execute Bind Query
no columns returned in result set --	No records returned from query

Error	Description
unable to create temp file for	Could not create a temporary file on the local file system
no bind value for merge field	Merge field exists for which there is no matching bind (replacement text)
unable to get form fields	Unable to determine the form fields in a PDF document
unable to merge PDF with XFDF	Merge of bind data with PDF document failed
multiple HTML files in zipped content	Correct format is one HTML file and one or more image files (gif, jpeg, and so on)
maximum number of images has been reached for content	Memory constraints limit the maximum number of images that can be associated with an HTML file
no HTML in zipped content --	Correct format is one HTML file and one or more image files (gif, jpeg, and so on)
no images in zipped content	Correct format is one HTML file and one or more image files (gif, jpeg, and so on)
could not read any zip file entries	Invalid Zip format
could not process zip file	Invalid Zip format
unable to download with query id	No query associated with the content
invalid extended_header format	Invalid extended header XML format
empty header_name in extended_header	Missing name in name-value pair
unable to create email address from	Invalid string for use as email address
unable to send email -- refer to error	Unable to send an email, refers to an error in the error log
unable to get text/plain content	Unable to get text content

Error	Description
unable to get content bind	Error in getting bind information associated with a Content item
could not get fnd_file_id for request	fnd_file_id not present for a content item in the XML
unrecognized mime type for dynamic content	Dynamic Content (capable of being merged) can only be of type ['text/html','richtext',pdf]
unable to merge legacy data	Merge failure occurs when trying to review a sent document
unable to download with fnd file id	Data not found in fnd_lobs for the file_id
unable to create directory	Unable to create a directory in the file system

Fulfillment Server Debug Flags

The Fulfillment Server can be debugged at runtime by using a debug flag in the Fulfillment Server startup script or in the command line. The debug information is written to the Fulfillment Server events log, with debug=log, or to Central Logging in the database.

In some cases, through use of third-party debugging such as Java Mail API, debug information is written to the console. The correct syntax for such flags is -Ddebug=<flag>[, <flag>]. The following table lists individual settings of the Fulfillment Server debug flags.

Flag	Description
full	Turn on all debug flags (*including 'log' which disables Central Logging).
off	Turn off all debugging.
content	Write request content, as downloaded from the database, to the local disk on which the server is running. The directory is called 'contents' and is under the log directory.

Flag	Description
email	Print details of email delivery the Events log and to the console. Not all information printed to the console is recorded in the Events log.
fax	Debug fax delivery.
history	Debug the writing of Fulfillment Request History, Fulfillment Content History, and Interaction History.
log	Debug the creation of local disk log files. *Disables the use of Central Logging.
monitor	Debug the operations of the Fulfillment Server internal monitors. All vital internal processes of the server are handled by these monitors.
print	Debug sending print jobs.
processor	Debug the functions of the server main thread.
query	Debug queries executed by the server. These are written to the Events log.
request	Debug requests that are processed by the server. Details of the request are written to the Events log.
header	Debug extended headers. Each is written as part of the email header for email sent by the server. Additional details of the request are written to the header such as request id, party_id, and so on.
jdbc	Debug JDBC calls made by the server. Java communicates with the database through JDBC.
fault	Debug the functioning of the server fault monitor. This provides details of request fault detection and handling.

Flag	Description
mass	Debug Mass Request processing.
failure	Debug failures that occur as a normal part of processing a request. The failure is written to the Error log to record a stack trace from where the failure originated. This skews the actual error count because an exception is created to track the stack trace of the failure.
merge	Debug the merge field parsing and insertion of bind variables within dynamic request content (Master Documents and email subject field).
queue	Debug reading and writing to Advanced Queues (AQ). The XML-based requests and responses are written to the Events log.
bind	Debug the use of bind data that is associated with the content of a fulfillment request.
labels	Debug PDF Labels requests.

Stopping and Restarting the Fulfillment Server

Stop and restart the Fulfillment server for changes to take effect to the settings, parameters and flags of the following servers:

- Fulfillment Server
- Email Server
- Print Server
- Fax Server

Email Issues

Use the following solutions to resolve these known issues with the Fulfillment Server sending email.

java.io.IOException: 500 5.5.1 Command Unrecognized

This type of Java IOException, recorded in the Fulfillment Server's error log, usually means that the email server did not recognize the sent command. This behavior may happen because the email server did not recognize the common SMTP command 'EHLO' but instead used the older 'HELO' command.

Solution

1. Use the Fulfillment Administration Console to configure the Fulfillment Server to send email through an outbound email server that recognizes the 'EHLO' command.
2. Restart the Fulfillment Server.

Characters Appear as '?' in Body of Email

The Fulfillment Server may parse characters or symbols incorrectly for some natural languages, other than English, that are not based on US ASCII. The Fulfillment Server sends email that contains unreadable characters, which appear as question marks. This situation may have the following causes:

- The Java Virtual Machine requires instructions on how to resolve these characters through the operating system.
- The Fulfillment Server startup script contains merge delimiters such as `«` and `»`. Previous versions of the Fulfillment Server recognized these merge field delimiters, but the current version does not recognize them.

Solution

- Use the configuration settings listed in Language Character Parsing, page A-9 in Fulfillment Server Commands and Startup Scripts, page A-1.
- Use the following flags in the Fulfillment Server startup script.

```
-Dmerge.LeftDelimiterString="&laquo;" -Dmerge.RightDelimiterString="&raquo;"
```

Frequently Asked Questions

Refer to the following frequently asked questions for help in resolving issues.

Uploading an RTF Document Fails in Oracle Content Manager

When uploading an RTF document in Oracle Content Manager, the upload fails.

Solution

This failure can occur when you have Microsoft Word open when you click the Submit for Approval button. The solution is to close Word and then click Submit again.

Content Types Not Visible

Why can I not see some content types in the list when I create a content item?

Solution

Only active content types appear in the list. It does not include inactive content types.

Cannot Update Existing Content Type

Why am I unable to update content types that are being used by content items?

Solution

Content types that are being used by content items cannot be updated.

Log Directory Path

The Fulfillment server is writing log files to the `./logs` subdirectory instead of `./log`. How can I change this path?

Solution

The default directory for error messages and logs is `./log`. This directory is determined by the `oracle.apps.jtf.fm.engine.logging.logger` class in the file `apps.zip`. If logs are written to a different directory (such as `./logs`) instead of `./log`, then change the `logger.class`. Open that class and see how `DEF_LOG_PATH` and `LogPath` variables are set. Use the `-Dengine.LogPath` switch to change the logging default directory. The format is: `-Dengine.LogPath=<directory>`

TeleSales Java Exception

The Fulfillment log receives a Java exception when trying to send collateral from Telesales. The Telesales popup window indicates that it was successful, however, there

is nothing in the history for the collateral, and the following error message appears in the Fulfillment log file.

```
'could not complete process for request '10011'  
oracle.apps.jtf.fm.engine.request.WorkNotCompletedException: failed to  
process part of request '10011' at  
java.lang.Throwable.fillInStackTrace(Native Method)
```

Solution

Use the following steps to check your Return Address setup for the Fulfillment server that is in use.

1. Log in using the Fulfillment Responsibility.
2. Select Server tab > server name > **Email Servers**.
3. Check that the Return Address setup is a valid email address to which you want to send all email replies.

java.lang.IllegalArgumentException

A java.lang.IllegalArgumentException message occurs while trying to start the FM Server Script. What is the problem?

Solution

This condition happens because of an improperly written Fulfillment Server startup script. The PrintEnabler and the FaxEnabler take the class file name from the script. If the file name is not correct, then the method that receives this classfile name will throw the exception. All the -D flags in the script should be written in either of the following ways:

- The Flags should be separated by a "\" and then a new line should contain the next flag. For example:

```
java -ms32m -mx64m\  
  
-Dengine.LogLevel=9\  
  
-Dengine.LogRequests=true\  
  
-Dpkg.ident=true\  
  
-Dengine.PrintEnabler=oracle.apps.jtf.fm.engine.rightfax.RfPrintEnab  
lerImpl\  

```

```
-Engine.FaxEnabler=oracle.apps.jtf.fm.engine.rightfax.RfFaxEnablerI  
mpl\  

```

```
oracle.apps.jtf.fm.engine.processor.Processor
```

- The Flags should be separated by spaces only. There should be no "\" or a carriage return anywhere in the Java call to the server. For example:

```
java -ms32m -mx64m -Engine.LogLevel=9 -Engine.LogRequests=true  
-Dpkg.ident=true
```

```
-Engine.PrintEnabler=oracle.apps.jtf.fm.engine.rightfax.RfPrintEnab  
lerImpl  
-Engine.FaxEnabler=oracle.apps.jtf.fm.engine.rightfax.RfFaxEnablerI  
mpl
```

```
oracle.apps.jtf.fm.engine.processor.Processor
```

DoNotContactException

A DoNotContactException message occurs while trying to send a request through the Fulfillment Server.

The outcome of the request is FAILURE and the following error appears in the Error.log file:

```
Unable to dispatch 'EMAIL' for request '<request_id>'
```

```
oracle.apps.jtf.fm.engine.disp.DoNotContactException: unable to email
```

Solution

When a request is submitted to the Fulfillment server, the server checks the APPS.HZ_CONTACT_PREFERENCES table for the values of "DO_NOT", "EMAIL" or "CONTACT" for PREFERENCE_CODE and CONTACT_TYPE, respectively, to determine if the email should not be sent to the customer. If the values are present in the table for the party_id that is specified in the request, then the request fails.

To change these values, do the following procedure:

1. Run the following query in the database for the party_id that is in use.

```
SELECT distinct b.contact_type  
FROM apps.hz_contact_preferences b,  
apps.hz_contact_points c  
WHERE  
b.preference_code = 'DO_NOT'
```

```

AND
((b.contact_level_table = 'HZ_PARTIES' and b.contact_level_table_id
= &party_id)
OR
(b.contact_level_table = 'HZ_CONTACT_POINTS'
AND c.contact_point_id = b.contact_level_table_id
AND c.owner_table_name = 'HZ_PARTIES'
AND c.owner_table_id = &party_id
AND c.status = 'A'))
AND nvl(b.preference_start_date,sysdate -1) < sysdate
AND nvl(b.preference_end_date,sysdate+1) > sysdate
ORDER BY b.contact_type

```

2. The Contact_type would appear as 'EMAIL' or 'CONTACT'. Remove the 'EMAIL'/'CONTACT' restriction or change the 'DO_NOT' restriction.
3. Resend the request.
It should now succeed.

Commands in Background Mode

How can I send commands to the Fulfillment Server while in background mode?

Solution

Do the following procedure:

1. If the Fulfillment server is running in console mode, shut it down by typing **kill**. If the Fulfillment Server is running in background mode, kill the process. To be safe, you might want to do a `ps -ef | grep java` to make sure that no other Fulfillment Server processes are running. Also, you need to be able to use a port to send commands through a remote command process. You may need to confirm with the database administrator which port is available.
2. To reference the port you will use to accept commands for the Fulfillment Server, add the following line to the startup script, where 99999 represents the selected port.

```
-Dengine.CommandPort=99999
```
3. Create a remote command script that is basically one line. Include the port to monitor in this script.

For UNIX:

```
#!/bin/csh java oracle.apps.jtf.fm.engine.remote.RemoteCommand
-port99999 /host <remote hostname>/ $*
```

For DOS:

```
java oracle.apps.jtf.fm.engine.remote.RemoteCommand -port99999 /host
<remote hostname>/ %*
```

4. Start the server by running the Server script in the console mode (not in background). Open another window and test the remote script by sending commands to it. If you name the remote script `RS.sh`, type a command like `RS.sh stat` or `RS.sh kill`. If the server that is running in console mode accepts these commands, everything is set up correctly. You can then type **RS.sh kill** to shut down the server.
5. If the remote functionality tested successfully, so you can now run the server in the background. If the Fulfillment Server Script was `FS.sh`, you could run `FS.sh &`.
6. When the server is running in the background, you can send commands to it by using your remote script `RS.sh`.

Viewing Request Processing

How can I see what Fulfillment Server processes are processing requests?

Solution

Type the command **stat** at the Fulfillment server command prompt to see what processes are processing requests.

"Initializing Processor" Error

When I try to start the Fulfillment Server, I receive the following error message. What is the problem?

```
INITIALIZING PROCESSOR ...

GETTING A DATABASE CONNECTION FOR SERVER CONFIGURATION ...

Processor cannot be started

oracle.apps.jtf.fm.engine.processor.ProcessorException

oracle.apps.jtf.fm.engine.processor.Processor.

initStoredConfiguration
```

Solution

Check for spelling mistakes and syntax errors in the Fulfillment startup script.

Merge Field Errors

What can I do if I encounter either of the following problems when my master documents and cover letters use merge fields?

- The merge fields do not get merged in email messages, fax messages or printed documents sent by Oracle One-to-One Fulfillment.
- The email messages, fax messages or printed documents show the merge fields surrounded by the "?" character, despite the use of the valid characters "<<" and ">>" when creating the master document or cover letter in the Administration Console.

Solution

This issue occurs with Windows NT 4.00.1381, Service Pack 6 with Internet Explorer 5.5.50. The browser is not able to take the standard ASCII value of the "<<" and ">>" merge field delimiters, and therefore the master document or Oracle Content Manager Template contains the "?" delimiters instead of "<<" and ">>".

Use one of the following workarounds:

- Instead of using the "Create and Upload" Master Document/OCM Template option within the One-to-One Fulfillment Administration Console, use the "Upload" Master Document/OCM Template option. Create the document using Microsoft Word 2000, insert the appropriate merge fields with the "<<" and ">>" special character delimiters and then save the document as a Web Page (for example, an HTML file). Use the Upload Master Document/OCM Template option to upload the file.
- If using the "Create and Upload" Master Document/OCM Template page, ensure that the desktop being used has Windows 2000 or a different platform configuration.

Unable to Use Merge Field Delimiters « and »

Prior to Release 11.5.8, some versions were missing the left angle quote « and the right angle quote ». The workaround is to insert the delimiter flags in the Fulfillment startup script as,

```
-Dmerge.LeftDelimiterString="&laquo;"
```

```
-Dmerge.RightDelimiterString="&raquo;"
```

Insert the flags after the Java or JRE command and before the name of the executed class. The class is 'oracle.apps.jtf.fm.engine.processor.Processor', in earlier releases, and 'oracle.apps.jtf.fm.FulfillmentServer' in later releases. The classes are case sensitive.

Table Rows and Merge Fields

How can I switch from repeating to non-repeating HTML table rows with merge fields?

Solution

Enhancements for this functionality are available in patches for Releases 11.5.8 and 11.5.9, and are part of the base product for Release 11.5.10 and later.

Code Line	Patch Number	Patch Description
11.5.8	3138990	Repeating row merge field enhancements
11.5.9	3121835	Oracle One-to-One Fulfillment Release 11.5.9.0.3
11.5.10	Not applicable	Included in Oracle One-to-One Fulfillment Release 11.5.10

In previous releases Oracle One-to-One Fulfillment treated all HTML source documents containing merge fields embedded in table rows as repeating rows. For example, if users wanted to create an itemized order form, they could create the HTML source document with an HTML table that represented the itemized list. The HTML table could have merge fields in the following columns:

ITEM_NUMBER, ITEM_DESCRIPTION, QUANTITY, UNIT_PRICE, TOTAL_PRICE

Example HTML Code

```
<table border="1" summary="Order Table">
<tr>
<th>Item Number</th>
<th>Item Description</th>
<th>Quantity</th>
<th>Unit Price</th>
<th>Total Price</th>
</tr>
```

```

<tr>
<td>{ (ITEM_NUMBER) }</td>
<td>{ (ITEM_DESCRIPTION) }</td>
<td>{ (QUANTITY) }</td>
<td>{ (UNIT_PRICE) }</td>
<td>{ (TOTAL_PRICE) }</td>
</tr>
<table>

```

Users can then use a key, such as ORDER_NUMBER, to associate a SQL query to the document that retrieves multiple rows of data from the database, as in the following example:

```

SELECT ITEM_NUMBER,

ITEM_DESCRIPTION,

QUANTITY,

UNIT_PRICE,

TOTAL_PRICE,

FROM ORDERS

WHERE ORDER_NUMBER = :my_order_number

```

When Fulfillment processes this document, it parses the HTML source document and identifies a table containing merge fields. As Fulfillment executes the query, it repeats the line containing the merge fields for each row that is returned from the query. In this way, a single HTML table row in the source document is custom expanded to all the results that are returned from the associated query.

Because this feature repeated any HTML table row with merge fields, it prevented some users from using HTML tables for document layout.

Solution

To accommodate users who want to use HTML tables for document layout, and also to support repeatable fields, from Release 11.5.10 onward Fulfillment can create both repeatable merge field rows and non-repeatable merge field rows. To keep the existing result of every table row containing repeated merge fields, do not make any changes. To enable the non-repeating row feature, use the following procedure.

Note: To use this feature, users must edit the HTML source directly. Microsoft Word does not allow direct HTML editing, so users will need

to use a plain-text editor instead of Word. Also, Oracle has made every effort to keep Fulfillment HTML 4.01-compliant so that no user agents (browsers) will be affected by any of these changes.

1. Add new styles

Two empty styles identify rows as repeating (jtorep) and non-repeating (jtonorep). To specify that rows can be repeating or non-repeating, add the following styles between the <head> </head> tags of your HTML source document.

```
<style type="text/css">
jtonorep {}
jtorep {}
</style>
```

2. Optionally, change the default behavior.

The default behavior is to expand automatically every row that contains merge fields. If your document has many rows with merge fields, but you want to repeat only some of those rows, you can change the default behavior to repeat only rows that contain merge fields when a tag specifies to repeat those rows.

To change the default behavior, add the following <meta> tag to the <head> section of the HTML source document:

```
<meta name="jto_tables" content="no_repeat">
```

If this meta tag is not in the HTML source document, then the default behavior is used. Adding this tag changes only the default behavior.

3. Identify the repeating and non-repeating rows.

After you have designed your document and you can see whether or not you want to set the default behavior to repeat, you are ready to modify the HTML table rows to indicate which rows should repeat and which rows should not repeat, as in the following example.

If in the previous order form example, you want to add a single row at the bottom of the table to show the order total, you would have the following HTML code:

```
<table border="1" summary="Order Table">
<tr>
<th>Item Number</th>
<th>Item Description</th>
<th>Quantity</th>
<th>Unit Price</th><th>
Total Price</th>
</tr>
<tr>
```

```

<td>{ (ITEM_NUMBER) }</td>
<td>{ (ITEM_DESCRIPTION) }</td>
<td>{ (QUANTITY) }</td>
<td>{ (UNIT_PRICE) }</td>
<td>{ (TOTAL_PRICE) }</td>

</tr>

<tr>

<td colspan="4">&nbsp;</td>
<td><b>{ (ORDER_TOTAL) }</b></td>

</tr>

</table>

```

The SQL query would change to:

```

SELECT  ITEM_NUMBER,
        ITEM_DESCRIPTION,
        QUANTITY,
        UNIT_PRICE,
        TOTAL_PRICE,
        ORDER_TOTAL
FROM ORDERS
WHERE ORDER_NUMBER = :my_order_number

```

PCDATA cannot be inserted: The default behavior repeats not only the item rows, but the order total row is repeated for each row that the query returns. To change the default, first add the new styles as described Step 1. Next, depending on whether or not you changed the default behavior described as in Step 2, you need to make one of the following changes.

Changing Default Behavior (Without Step 2 Addition)

The first row (the row with the table headers or <th> tags) does not have any merge fields, so it will not be repeated.

The second row will automatically be repeated because it contains merge fields, and you have not changed the default behavior. It can stay as it is.

The third row will also be repeated automatically, so you need to explicitly instruct it not to repeat. To do this, you need to use the `jtonorep` class style that was defined in step one. Change the third row from,

```

<tr>

<td colspan="4">&nbsp;</td>
<td><b>{ (ORDER_TOTAL) }</b></td>

</tr>

```

to

```

<tr class="jtonorep">

<td colspan="4">&nbsp;</td>
<td><b>{ (ORDER_TOTAL) }</b></td>

```

```
</tr>
```

This change specifies the class for this `<tr>` as `jtonorep`. When Fulfillment sees this tag, it will not repeat this row. It will only write the row once regardless of the number of rows that the query returns.

Altered Default Behavior Change

If you changed the default behavior as in Step 2, no rows will repeat. You need to tell Fulfillment which row to repeat.

If the second row is the row that you want to repeat, change it from,

```
<tr>
<td>{ (ITEM_NUMBER) }</td>
<td>{ (ITEM_DESCRIPTION) }</td>
<td>{ (QUANTITY) }</td>
<td>{ (UNIT_PRICE) }</td>
<td>{ (TOTAL_PRICE) }</td>
</tr>
```

to

```
<tr class="jtorep">
<td>{ (ITEM_NUMBER) }</td>
<td>{ (ITEM_DESCRIPTION) }</td>
<td>{ (QUANTITY) }</td>
<td>{ (UNIT_PRICE) }</td>
<td>{ (TOTAL_PRICE) }</td>
</tr>
```

The line `<class="jtorep">` was added to the `<tr>` element for the second row to tell Fulfillment to repeat this row for each result row from the SQL query.

Additional Information

Whether or not you changed the default behavior by including the `<meta>` tag from Step 2, you can use both attributes `class="jtonorep"` and `class="jtorep"` in the same document. These attributes override the default behavior and specify what to disregard. Also, these attributes only take affect where there are table rows with merge fields. If there are no merge fields, no rows are repeated.

Because of HTML 4.01 specifications, multiple class names can be associated to the class attribute. They simply need to be separated by white space. This is important if you have a nicely formatted HTML document already that is using style sheets. You only need to add `jtorep` or `jtonorep` to the necessary rows separated by white space.

Note: The class attribute must be in lowercase, and the class names must be enclosed in double quotes, as in: `<td class="jtorep">`.


```

</html>
Order Table Example With Altered Behavior
<html>
<head>
<title>Sample Order Table</title>
<meta name="jto_tables" content="no_repeat">
<style type="text/css">
jtonorep {}

jtorep {}

</style>
</head>
<body>
<table border="1" summary="Order Table">
<tr>
<th>Item Number</th>
<th>Item Description</th>
<th>Quantity</th>
<th>Unit Price</th>
<th>Total Price</th>
</tr>
<tr class="jtorep">
<td>{ (ITEM_NUMBER) }</td>
<td>{ (ITEM_DESCRIPTION) }</td>
<td>{ (QUANTITY) }</td>
<td>{ (UNIT_PRICE) }</td>
<td>{ (TOTAL_PRICE) }</td>
</tr>
<tr>
<td colspan="4">&nbsp;&nbsp;&nbsp;&nbsp;</td>
<td><b>{ (ORDER_TOTAL) }</b></td>
</tr>
</table>
</body>
</html>

```

Server id and User id in create_ fulfillment API Call

Is it necessary to pass Server id and User id for the create_ fulfillment API call?

Server_id is not required, but user id is required.

In Release 11.5.8, if a valid One-to-One Fulfillment Server was not associated with the user, API would throw an error. In Release 11.5.9, API checks for the One-to-One Fulfillment Server that is associated with that user. If the API does not find any server associated with the user, then the request is submitted to Server Id 5000 (the Rapid Install One-to-One Fulfillment Server that is available out of the box). The Rapid Install Server must be running to pick up these requests.

Sending a Request Through Multiple Media Types

How can I send the same request through multiple types of media, such as email, fax, and print?

Fulfillment allows sending a document by using the supported media types EMAIL, FAX and PRINT.

When GET_CONTENT_XML is called, the p_media_type parameter can be set to the following combinations of values, depending on the requirements.

- p_media_type := 'EMAIL' or 'FAX' or 'PRINTER'
- or 'EMAIL,FAX,PRINTER'
- or 'EMAIL, FAX'
- or 'EMAIL, PRINTER'
- or 'FAX,PRINTER'

The sequence does not matter. Separate the media types with a comma. To successfully send a message, ensure that proper values are passed for the parameters p_email, p_fax and p_printer.

Debugging

Use the following procedures to set and view debug logging for PL/SQL and the Fulfillment Server.

PL/SQL Debugging

All logging is at the procedure level (Level 2). To debug at the API level, turn on debugging, and set the logging level to Procedure or Statement.

To run a test script and enable debugging, call the following API before calling the Fulfillment APIs.

```
l_session_id          NUMBER :=1000;
```

```
fnd_global.initialize(1_session_id,1000,1000,1000,1000,1000,1000,100,100,100,100,1000);
```

After the logging level is set and the application is initialized, whenever a request is sent to the One-to-One Fulfillment API, a debug is written to the central repository.

To view all debugs, use the following procedure.

Login

HTML Administration

Responsibility

System Administrator

Prerequisites

None

Steps

1. Select the Settings tab > System sub tab > **Messages** link in the Debug Logging group.
The Debug Log Viewer page appears.
2. In the fields, enter search criteria to define the search range.
3. In the Module field, reset the debug logging level by entering **jtf.plsql%JTF_FM% .**
4. Click **View Log**.
The Log Summary and Log Messages appear.

Server Debugging

For the server to write to the central logging repository, set the logging level to one of the following levels:

- Event (Level 3)
- Procedure (Level 2)
- Statement (Level 1)

After a request is processed, use the following procedure to view logs.

Login

HTML Administration

Responsibility

System Administrator

Prerequisites

None

Steps

1. Select the Settings tab > System sub tab > **Messages** link in the Debug Logging group.
The Debug Log Viewer page appears.
2. In the fields, enter search criteria to define the search range.
3. Do one of the following:
 - In the Module field, reset the debug logging level by entering `jtf.fm.engine.logging%`.
 - If the request identifier is known, append it to the Module value, for example, `jtf.fm.engine.logging%12345%` (assuming 12345 is the request identifier).
 - If the job is also known and the request identifier, add it to the Module value also, for example, `jtf.fm.engine.logging%12345%34%` (assuming 35 is the party identifier).

Troubleshooting the RightFax Integration

Q: What can I do if a fax request is sent and the Fulfillment Server is not set up to send faxes?

A: If the Fulfillment Server receives a fax request and is not configured to send faxes, an error will be logged in the Servers error log and Fulfillment History tables will note the failure to send the fax.

Q: What can I do if a print request is sent and the Fulfillment Server is not set up to send print jobs?

A: If the Fulfillment Server receives a print request and is not configured to send print

jobs, an error will be logged in the Servers error log and Fulfillment History tables will note the failure to print.

Administration Tasks

This chapter describes task-based procedures for administering Oracle One-to-One Fulfillment.

This chapter covers the following topics:

- Managing Servers
- Setting the Fulfillment Server Start Time and Shutdown Time
- Setting the Interaction History Outcomes and Results
- Enabling the Unsubscribe Option in Email Requests
- Enabling Tracking for Email Requests
- Setting Up Bounceback
- Defining Print Servers and Printers for High-Quality Printing
- Configuring a Print Server for High-Quality Printing
- Configuring Printers for High-Quality Printing
- Managing Fulfillment Agent Groups
- Creating an Agent Group
- Adding an Agent to a Group
- Removing an Agent from a Group
- Assigning a Fulfillment Agent Group to a Fulfillment Server
- Managing Oracle Marketing Encyclopedia System Templates
- Creating a Template
- Assigning a Master Document to a Template
- Assigning Collateral to a Template
- Updating a Template
- Removing a Master Document from a Template

- Removing Collateral from a Template
- Managing Oracle Marketing Encyclopedia System Master Documents
- Creating an Oracle Marketing Encyclopedia System Master Document Using the Text Editor
- Uploading an Oracle Marketing Encyclopedia System Master Document
- Updating an Oracle Marketing Encyclopedia System Master Document
- Viewing an Oracle Marketing Encyclopedia System Master Document
- Downloading an Oracle Marketing Encyclopedia System Master Document
- Assigning a Query to an Oracle Marketing Encyclopedia System Master Document
- Removing a Query From an Oracle Marketing Encyclopedia System Master Document
- Viewing Oracle Marketing Online Collateral
- Managing Oracle Marketing Encyclopedia System Queries
- Creating a Query
- Updating a Query
- Viewing Oracle Marketing Encyclopedia System Master Documents Associated With a Query
- Creating a Data Source File from a Query
- Managing Content
- Cover Letter Overview
- Understanding Cover Letter Content Types
- Queries
- Creating and Managing Cover Letter Content Types
- Creating Cover Letters
- Managing Fulfillment Requests
- Viewing the Status of a Fulfillment Request
- Canceling a Fulfillment Request
- Pausing or Resuming a Fulfillment Request
- Managing Fulfillment Request History
- Viewing the Details of a Fulfillment Request
- Viewing Failed Fulfillment Requests
- Viewing a Partially Successful Fulfillment Request
- Correcting a Malformed Email Job

- Viewing Job Details
- Resubmitting a Job
- Resubmitting a Fulfillment Request
- Deleting a History Entry
- Searching for a Fulfillment Request
- Using the Quick Find Feature
- Using Merge Fields
- Default and Wildcard Merge Delimiter Pairs
- Different Representations of the Same Character
- Using Encrypted Merge Fields
- Creating an RTF Form Document with Merge Fields
- Creating a PDF Form Document with Merge Fields
- Setting Up High Quality Printing
- Configuring the Printer
- Setting Up a Pasta Print Server
- Verifying the Printer Configuration
- Running Concurrent Programs

Managing Servers

Use the following procedures to administer the Fulfillment, email, print and fax servers.

- Setting the Fulfillment Server Start Time and Shutdown Time, page 6-4
- Setting the Interaction History Outcomes and Results, page 6-5
- Enabling the Unsubscribe Option in Email Requests, page 6-5
- Enabling Tracking for Email Requests, page 6-6

Note: In all Fulfillment server configuration options, for the changes to take effect you must restart any running servers. The information is gathered only once at startup.

See Also

- Configuring Servers, page 3-13

Setting the Fulfillment Server Start Time and Shutdown Time

Use this procedure to set the start and shutdown times for the Fulfillment Server. The start and shutdown times indicate the times when the Fulfillment Server polls the fulfillment request queues in the database. It does not physically start or stop the server. You can set a start and shutdown time if you do not want the Fulfillment Server to process fulfillment requests twenty-four hours a day.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab > Server sub tab.
The Servers page appears.
2. Click a Fulfillment Server name link.
The Server Details page appears.
3. From the Start Time lists, select the hour and minute for the time at which the Fulfillment Server starts polling the fulfillment request queues in the database. It does not physically start the server. If running, the server will continue to process retrieved fulfillment requests even when it is not polling the database.
4. From the Shutdown Time lists, select the hour and minute for the time at which the Fulfillment Server stops polling the fulfillment request queues in the database. It does not physically stop the server. If running, the server will continue to process retrieved fulfillment requests even when it is not polling the database.
5. Click **Continue**.
6. Restart any currently running Fulfillment servers for the changes to take effect.

Setting the Interaction History Outcomes and Results

Use this procedure to set the interaction history outcomes and results for processed fulfillment requests.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab > Server sub tab.
The Servers page appears.
2. Click a Fulfillment Server name link.
The Server Details page appears.
3. From the Request Outcome list, select the outcome that you want to associate with a fulfillment request job that has been processed by the Fulfillment Server.
4. From the Request Success Result list, select the result that you want to associate with a fulfillment request job that has been successfully processed by the Fulfillment Server.
5. From the Request Failure Result list, select the result that you want to associate with a fulfillment request job that has not been successfully processed by the Fulfillment Server.
6. Click **Continue**.
7. Restart any currently running Fulfillment servers for the changes to take effect.

Enabling the Unsubscribe Option in Email Requests

Use this procedure to enable the Fulfillment Server to insert a hypertext link into the email content that will allow recipients to access their contact preferences in Oracle

Marketing.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab.
The Servers page appears.
2. Click a Fulfillment Server Name.
The Server Details page appears.
3. In the side panel, click **Email Servers**.
The Email Servers page appears.
4. Select **Unsubscribe**.
5. In both of the fields:
 - HTML Unsubscribe link and URL
 - Text Unsubscribe link and URL

you can enter a message that tells customers how to unsubscribe, and a URL by which they can unsubscribe.

The fields are used for HTML and Text emails respectively.

6. Click **Update**.
7. Restart any currently running Fulfillment servers for the changes to take effect.

Enabling Tracking for Email Requests

Use this procedure to enable the Fulfillment Server to track email statistics for all email requests.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab.
The Servers page appears.
2. Click a Fulfillment Server name link.
The Server Details page appears.
3. In the side panel, click **Email Servers**.
The Email Servers page appears.
4. Select **Track Email**.
5. In the Tracking URL field, enter the HTML code that includes the URL that is called when recipients open email.
6. Click **Update**.
7. Restart any currently running Fulfillment servers for the changes to take effect.

Setting Up Bounceback

Bounceback management allows users to address outbound email messages so that destination email servers send "bounces" back to a specific account. The address to which the bounced messages are sent contain the request and job information, which allows Oracle One-to-One Fulfillment to identify which email messages bounced.

Email administrators need to update the Sendmail configuration to direct bounced messages into a single bounceback account. The configuration steps that are required for Bounceback Management can be performed on both new and existing Fulfillment Servers. Bounceback management does not affect the functionality that is provided with Tracking or with the Unsubscribe feature.

Bounced messages are recorded. To determine the reason why a message bounced, a user would have to perform an analysis outside of what Oracle One-to-One Fulfillment provides.

Configuring Bounceback In Fulfillment Administration Console

Use the following procedure to configure email bounceback in the Administration console.

Login

Fulfillment Administrator Console

Responsibility

Not Applicable

Prerequisites

Set up Sendmail

Steps

1. Select the Server tab > server name > Email Servers.
The Email Servers page appears.
2. In the Bounceback Management group, select the Enable Bounceback option.
This option instructs the Fulfillment Server to create the bounceback monitoring thread when it starts, and to append the bounceback information as the From address on outgoing email messages.
3. Enter the host name or IP address of the IMAP host where the bounce account has been created.
4. Add the IMAP port. The default IMAP port is 143.
5. Enter the account name and the account password.
6. Select a run frequency. The run frequency indicates how often the bounceback monitoring thread will run. A typical Run Frequency is every two hours. Set this value lower if you need quicker results and are sending large volumes of email. Set this value higher if you are not concerned with recording bounceback messages. The difference is that they will be recorded, but if you only have it run every twelve hours, updates to the counts are only done in that interval.

When running, the thread connects to the bounces account and scans the inbox for

any new messages. For each new message, it updates the database. When you view the request history, you can see the number of bounced messages in a particular request.

7. Enter a Return Path. The return path should be the email address for the bounce account, for example, bounces@yourcompany.com.
8. Restart the Fulfillment server for the changes to take effect.

Defining Print Servers and Printers for High-Quality Printing

The Printer tab is only for users of Oracle E-Business Suite Applications to configure print servers and printers that are used with High-Quality Printing.

Tasks

You can perform the following tasks:

- Configuring a Print Server for High-Quality Printing, page 6-9
- Configuring Printers for High-Quality Printing, page 6-10

Configuring a Print Server for High-Quality Printing

Use the following procedure to create a print server.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Select the Server tab.
2. Click a server name.
3. Click Print Servers.

The Print Servers page appears.

4. Click **Create New Print Server**.

The Create Print Server page appears.

5. In the Print Server Name field, type a descriptive name for the print server.
6. For printing RTF or PDF documents, in the Connection String field, type the URL or IP address of the print server.
7. From the Server Type list, select **Pasta**.
8. Click **Apply**.

The Print Servers page appears.

Configuring Printers for High-Quality Printing

Use the following procedure to associate printers with a printer server.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Select the Server tab.
2. Click a server name.
3. Click Print Servers.
The Print Servers page appears.
4. Click a print server Name.
The Update Print Server page appears.
5. Click **Create Printer**.

The Create Printer page appears.

6. In the Printer Name field, type a descriptive name for this printer.
7. Optionally, in the Description field, type a description of this printer.
8. In the Network Name field, type the printer name that identifies this printer on your local area network.
9. Click **Apply**.

The Update Print Server page appears.

Managing Fulfillment Agent Groups

Use the Group sub tab to create a fulfillment agent group and assign the fulfillment agent group to a Fulfillment Server.

Tasks

You can perform the following tasks:

- [Creating an Agent Group, page 6-11](#)
- [Adding an Agent to a Group, page 6-12](#)
- [Removing an Agent from a Group, page 6-13](#)
- [Assigning a Fulfillment Agent Group to a Fulfillment Server, page 6-14](#)

Creating an Agent Group

Use this procedure to define a group of agents for fulfillment processing.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Create user accounts for agents.

Steps

1. Click the Group tab > Groups sub tab.
The Groups page appears.
2. Click **Create**.
The Create Group page appears.
3. In the Group Name field, enter a unique name for the group.
You cannot change this field after the server is saved.
4. Optionally, in the Description field, enter a description of the group.
This field is for informational purposes only.
5. From the Server Name list, select the name of the Fulfillment Server.

Note: Each fulfillment agent group must be linked to a Fulfillment Server. However, if you have not created a Fulfillment Server, you can return to the Group tab and select a Fulfillment Server later.
6. Add agents to the group.
 1. In the Agent area, in the Name column, type a user name or other search criteria, and then click **Go**.

You may search for a user name using character strings and the wildcard symbol (%). For example, to find user names starting with A, type A%.

The Select Agent page appears. The search results are listed.
 2. To modify your search, type new search criteria in the Search field and then click **Search**.
 3. Click a user name.

The Create Group page appears. The user name appears in the Agents area.
7. Click **Create**.
The Groups page appears.

Adding an Agent to a Group

Use this procedure to add an agent to a fulfillment agent group.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Group tab > Groups sub tab.
The Groups page appears.
2. Click a group name.
The Group Detail page appears.
3. In the Agent area, in the Name column, type a user name or other search criteria, and then click **Go**.

You may search for a user name using character strings and the wildcard symbol (%). For example, to find user names starting with A, type A%.

To clear the search criteria, click the Restore button.

The Select Agent page appears. The search results are listed.
4. To modify the search, type new search criteria in the Search field and then click **Search**.
5. Click a user name.
The Group Detail page appears. The user name appears in the Agents area.
6. Click **Update**.

Removing an Agent from a Group

Use this procedure to remove an agent from a Fulfillment agent group.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Group tab.
2. Click the Groups sub tab.
The Groups page appears.
3. Click a group name.
The Group Detail page appears.
4. In the Agent area, select the Remove box for the agent that you want to remove from the agent group.
5. Click **Update**.
The agent is removed from the agent group and the Groups page appears.

Assigning a Fulfillment Agent Group to a Fulfillment Server

Use this procedure to assign a fulfillment agent group to a Fulfillment Server.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

- Create user accounts for agents.
- Create a fulfillment agent group.
- Configure a fulfillment server.

Steps

1. Click the Group tab > Groups sub tab.
The Groups page appears.
2. Click a group name.
The Group Detail page appears.
3. From the Server Name list, select a Fulfillment Server.
4. Click **Update**.

Managing Oracle Marketing Encyclopedia System Templates

Use the Template sub tab to manage groupings of Oracle Marketing Encyclopedia System master documents and OMO collateral items.

Tasks

You can perform the following tasks:

- Creating a Template, page 6-15
- Assigning a Master Document to a Template, page 6-17
- Assigning Collateral to a Template, page 6-18
- Updating a Template, page 6-19
- Removing a Master Document from a Template, page 6-20
- Removing Collateral from a Template, page 6-21

Creating a Template

Use this procedure to create a grouping of Oracle Marketing Encyclopedia System master documents and Oracle Marketing collateral items (also called deliverables).

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Create master document and at least one master document collateral.

Steps

1. Click the MES Templates tab > Template sub tab.
The Templates page appears.
 2. Click **Create**.
The Create Template page appears.
 3. Enter the details about the template.
 1. In the Template Name field, type a unique name for the template.
 2. From the Status list, choose the status of the template.
 - Active: The template is available for fulfillment requests.
 - Inactive: The template is not available for fulfillment requests.

You have the following options:
 3. Optionally, in the Description, type a description of the template.
This field is for informational purposes only.
 4. Click **Continue**.
The Template Master Document window opens.
4. Add one or more master documents to the template.
 1. In the Master Documents area, in the Name column, type a master document name or other search criteria, and then click **Go**.

You may search for a master document using character strings and the wildcard symbol (%). For example, to find master documents starting with A, A%.

To clear the search criteria, click **Restore**.

The Select Master Document page appears. The search results are listed.

2. To modify your search, type new search criteria in the Search field and then click **Search**.
3. Click **Select** next to the master document name.
4. Click **Update**.
The View Template page appears.
5. Optionally, to add one or more collateral to the template, click **Collateral**.
 1. In the Collateral area, in the Name column, type a collateral name or other search criteria, and then click **Go**.

You may search for collateral using character strings and the wildcard symbol (%). For example, to find collateral starting with A, A%.

If you want to clear the search criteria, then click **Clear**.

The Select Collateral page appears, listing the search results.
 2. To modify your search, type new search criteria in the Search field and then click **Search**.
 3. Click **Select** next to the collateral name.
 4. Click **Update**.
The View Template page appears.

Assigning a Master Document to a Template

Use this procedure to associate a master document with a template.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

- Create a template.
- Create a master document.

Steps

1. Click the MES Templates tab > Template sub tab.
The Templates page appears.
2. Click a template name.
3. In the side panel, click Master Documents.
The View Template page appears.
4. In the Master Documents area, in the Name column, type a master document name or other search criteria, and then click **Go**.
You may search for a master document using character strings and the wildcard symbol (%). For example, to find master documents starting with A, A%.
To clear the search criteria, click **Restore**.
The Select Master Document page appears. The search results are listed.
5. If you want to modify your search, type new search criteria in the Search field and then click **Search**.
6. Click **Select** next to a master document name, then click **Update**.
You have associated the master document with the template that you selected in step 2.

Assigning Collateral to a Template

Use this procedure to associate a collateral item with a template.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

- Create a template.
- Create collateral.

Responsibility

One-to-One Fulfillment Administrator

Steps

1. Click the MES Templates tab > Template sub tab.

The Templates page appears.

2. Click a template name.

3. In the side panel, click Collateral.

The View Template page appears.

4. In the Collateral area, in the Name column, type a collateral name or other search criteria, and then click **Go**.

You may search for collateral using character strings and the wildcard symbol (%). For example, to find collateral starting with A, A%.

To clear the search criteria, click **Restore**.

The Select Collateral page appears. The search results are listed.

5. To modify your search, type new search criteria in the Search field and then click **Search**.

6. Click **Select** next to a collateral name.

7. Click **Update**.

The View Template page appears.

Updating a Template

Use this procedure to update a template.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Template sub tab.
The Templates page appears.
2. Click a template name link.
The View Template page appears.
3. From the Status list, select the status of the template.
You have the following options:
 - Active: The template is available for fulfillment requests.
 - Inactive: The template is not available for fulfillment requests.
4. Optionally, in the Template Description field, type a description about the template.
This field is for informational purposes only.
5. Click **Update**.

Removing a Master Document from a Template

Use this procedure to remove a master document from a template.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Template sub tab.
The Templates page appears.
2. Click a template name.
3. In the side panel, click **Master Documents**.
The View Template page appears.
4. Select the Remove box for the master document that you want to remove from the template.
5. Click **Update**.
The master document is disassociated from the template and the Master Document list in the View Template page is refreshed.

Removing Collateral from a Template

Use this procedure to disassociate a collateral item from a template.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Template sub tab.

The Templates page appears.

2. Click a template name.
3. In the side panel, click **Collateral**.

The View Template page appears.

4. Select **Remove** next to the collateral name.
5. Click **Update**.

The collateral is disassociated from the template. The Collateral list in the View Template page is refreshed.

Managing Oracle Marketing Encyclopedia System Master Documents

Use the Master Document sub tab to manage documents that are stored in Oracle Marketing Encyclopedia System. The content in the document becomes the body of the fulfillment email or fax print job.

Tasks

You can perform the following tasks:

- Creating an Oracle Marketing Encyclopedia System Master Document Using the Text Editor, page 6-22
- Uploading an Oracle Marketing Encyclopedia System Master Document, page 6-24
- Updating an Oracle Marketing Encyclopedia System Master Document, page 6-25
- Viewing an Oracle Marketing Encyclopedia System Master Document, page 6-26
- Downloading an Oracle Marketing Encyclopedia System Master Document, page 6-26
- Assigning a Query to an Oracle Marketing Encyclopedia System Master Document, page 6-27
- Removing a Query From an Oracle Marketing Encyclopedia System Master Document, page 6-27

Creating an Oracle Marketing Encyclopedia System Master Document Using the Text Editor

Use this procedure to create a master document using the text editor in Oracle

One-to-One Fulfillment. The master document will be saved in Oracle Marketing Encyclopedia System.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Master Document sub tab.
The Master Document page appears.
2. Click **Create and Upload**.
The Master Document Editor page appears.
3. In the Master Document Name field, type a unique name for the master document.
When you save the document, the body of the fulfillment document will be saved with this name (for example, `name.htm`). In addition, the name will appear in list of documents in Oracle One-to-One Fulfillment and the business application. You cannot change this field after the document is saved.
4. If you want to compose the master document using plain text, click the **Change to Plain Text Mode** link.
5. If you want to compose the master document using formatted text, click the **Change to Rich Text Mode** link.
A formatting tool bar appears.
6. If you want to insert a merge field derived from a query, then do the following:
 1. In the SQL Merge Field field, click **Create Query**.
The SQL Editor page appears.
 2. In the Query Name field, type a unique name for the query.
You cannot change this field after the server is saved.

3. Optionally, in the Query Description field, enter a description of the query.
This field is for informational purposes only.

4. In the SQL Query field, type the SQL statement.

Note: Do not insert a semicolon (;) at the end of the SQL statement.

5. Click **Validate SQL and Save**.

If the syntax is valid, the SQL Editor page closes and the Master Document Editor page refreshes. The text of the query appears in the SQL Query field. The aliases (or columns, if there are no aliases) in the query appear in the SQL merge field list. The query is saved in Oracle Marketing Encyclopedia System and appears in the Query list.

If you want to edit the SQL statement, click the **Edit** link.

Merge fields are the fields available from the query.

6. From the SQL/Merge Field list, select a merge field.

7. Place the cursor in the document area.

8. Click **Insert**.

7. If you want to manually create a merge field, then in the Custom Merge Field, type a merge field from the query and then click **Insert**.

The merge field is inserted into the document.

8. Click **Save**.

The Master Documents page appears. The master document appears in the list and if a query was created when the master document was created, the query appears in the Query list. The document is saved with an htm extension.

Uploading an Oracle Marketing Encyclopedia System Master Document

Use this procedure to upload a document to the Oracle Marketing Encyclopedia System as a master document.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Master Document sub tab.
The Master Document page appears.
2. Click **Upload**.
The Update Master Document page appears.
3. In the File Name field, click the **Browse** button to navigate to the directory that contains the document.

Note that the name of the file is the name of the document that will appear in the Master Documents list. You will not be able to change this name later.
4. Select the master document.
5. Click **Upload**.
The Upload Master Document page refreshes.
6. Optionally, in the Description field, type a description of the Master Document.
7. Optionally, from the Query list, select the query that you want to associate with the master document.
8. Click **Upload**.
The Master Document page appears.

Updating an Oracle Marketing Encyclopedia System Master Document

To update a document that has been uploaded to Oracle Marketing Encyclopedia System, upload the document again. The document will have a unique numerical identifier even if it has the same name as a previously uploaded document. Therefore, the list of Oracle Marketing Encyclopedia System master documents may contain more than one document with the same name.

Viewing an Oracle Marketing Encyclopedia System Master Document

Use this procedure to view an Oracle Marketing Encyclopedia System master document.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Master Document sub tab.
The Master Document page appears.
2. Click a master document name link.
3. If the document is an HTML file, then the file appears in a new browser window.
4. If the document is a Zip file, then a File Download window appears.
You can open or save the Zip file.

Downloading an Oracle Marketing Encyclopedia System Master Document

Use this procedure to download an Oracle Marketing Encyclopedia System master document to a local file system.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Master Document sub tab.
The Master Document page appears.
2. Right-click a master document name link.
You can open or save the file.

Assigning a Query to an Oracle Marketing Encyclopedia System Master Document

Use this procedure to assign a query to an Oracle Marketing Encyclopedia System master document.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Master Document sub tab.
The Master Document page appears.
2. From the Query list, select a query.

Removing a Query From an Oracle Marketing Encyclopedia System Master Document

Use this procedure to remove a query from an Oracle Marketing Encyclopedia System

master document.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the MES Templates tab > Master Document sub tab.
The Master Document page appears.
2. From the Query list, select **No Query**.

Viewing Oracle Marketing Online Collateral

Use the Template tab > Collateral sub tab to view a list of the collateral items (called "deliverables" in Oracle Marketing) that are available in Oracle Marketing Online. Collateral items are attached to Fulfillment email.

Managing Oracle Marketing Encyclopedia System Queries

Use the Query sub tab to manage queries for Oracle Marketing Encyclopedia System Master Documents. These queries supply data for dynamic Oracle Marketing Encyclopedia System master documents.

Tasks

You can perform the following tasks:

- Creating a Query, page 6-29
- Updating a Query, page 6-30
- Viewing Oracle Marketing Encyclopedia System Master Documents Associated with a Query, page 6-30

- Creating a Data Source File from a Query, page 6-31

Creating a Query

Use this procedure to create a query.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Query tab > Query sub tab.
The Queries page appears.
2. Click **Create**.
The Create Query page appears.
3. In the Query Name field, type a unique name for the query.
You cannot change this field after the query is saved.
4. Optionally, in the Description field, enter a description of the query.
This field is for informational purposes only.
5. In the SQL Query field, type the SQL SELECT statement.
6. Click **Validate SQL**.
If the syntax of the statement is valid and there are no bind variables, then the query is saved and Queries page appears.
If there are bind variables, then you will have to supply the bind variable before the query can be saved.

Updating a Query

Use this procedure to update a query.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Query tab > Query sub tab.
The Queries page appears.
2. Click a query name link.
The Create Query page appears.
3. Optionally, in the Description field, enter a description of the query.
This field is for informational purposes only.
4. In the SQL Query field, type the SQL SELECT statement.
5. Click **Validate SQL**.

Viewing Oracle Marketing Encyclopedia System Master Documents Associated With a Query

Use this procedure to view a list of the Oracle Marketing Encyclopedia System Master Documents that use a query.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Query tab > Query sub tab.
The Queries page appears.
2. Click a query name link.
The Create Query page appears. The master documents that use the query appear under the Documents Using Query heading.

Creating a Data Source File from a Query

Use the data source sub tab to download a list of merge fields for a query. The text file can be used by other tools, such as Microsoft Word, to create a master document that can be uploaded to Oracle Marketing Encyclopedia System.

Use this procedure to download a list of merge fields for a query.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Create a query.

Steps

1. Click the Query tab > Datasource sub tab.
The Create Datasource page appears.
2. From the Query Name list, select a query.

3. **Click Create.**
The data source file is created and the Create Datasource page refreshes. The data source file is a text file with the column headings or aliases.
4. Use the Datasource file link to download the data source file.
5. To open the data source file, right-click the **Datasource file** link and choose Open.
The Datasource appears in a separate browser window.
6. To save the data source file, right-click the **Datasource file** link and choose Save Target As.
7. Navigate to the desired save location.
8. Optionally, modify the file name.
The save type should be Text Document.
9. Click **Save**.

Managing Content

This chapter explains the concepts to create cover letter content types and cover letters. Sections in this topic includes:

- Cover Letter Overview, page 6-32
- Understanding Cover Letter Content Types, page 6-33
- Queries, page 6-35
- Creating and Managing Cover Letter Content Types, page 6-36
- Creating Cover Letters, page 6-39

Cover Letter Overview

A cover letter is the document that is received by the customer. It can be defined as an outbound content sent to consumers about an offer. It is used to communicate about an offer, information or a message. Typically, a cover letter is created by marketers. It can contain attachments such as data sheets, white papers, analyst reports, press releases, and so on. The cover letters are fulfilled through email, fax, and print channels.

The content is stored in Oracle Content Manager (OCM) central repository. The advantages of using OCM are that it:

- Ensures that the message is consistent, when executed through all channels of execution.
- Helps organizations to maintain the integrity of the content and keep track of its usage.
- Allows organizations to maintain control over the content by giving users the access permissions on folders, by creating content type groups, and by having approval process to approve the content.
- Restricts users to access the content by having access permissions on folders, creating content type groups, and approval process.

Understanding Cover Letter Content Types

A cover letter content type is like a setup template that you can use while creating a content. It provides best practices that enable you to create standardized content. It contains detailed information such as attributes, data types, and blocks that are required to create a cover letter. To enable this, few seeded content types are available. Only administrators can create content types.

While creating content types, you can:

- Restrict the content type to be made available to only certain groups by mapping them to content type groups such as email, print, and fax.
- Decide to set default attributes for the cover letters.
- Set the attributes so they cannot be updated.
- Specify the style sheet to be mapped to the content type.
- Associate multiple style sheets to a content type. This feature enables users to view the content or attachments using any style sheet that is associated with it. Also, a default style sheet can be mapped to the content type.

Other users can duplicate the seeded content types and modify their structures by updating the attributes.

1. If the content type is used to create a cover letter or has the status In Progress, you cannot update the content type.
2. To create cover letters and content type groups that are relevant to cover letters, you must use the options that are provided with Cover Letters and not the generic Oracle Content Manager UI.

3. To map the content types to different groups, you must use this UI and not the Oracle Content Manager Content Type Groups UI.

Content Type Summary Page

In the Content Type Summary page, you can choose to view all, seeded, or inactive content types. The results are displayed in a tabular format listing the content type details such as the owner's name, its status, and so on.

Cover Letter Content Type Attributes

To create new content types, you can either copy seeded content types or create new ones based on the following attributes:

General

In this region, enter the name of the content type, select the status, and the content type group. Only if the status of the content type is active, you can select it to create cover letters. You can also select a default style sheet for the cover letter.

For more information on style sheets, see *Oracle Content Manager Implementation and Administration Guide*.

Header, Content Block, and Footer

You can use these regions to provide content in text format and images. Typically, the header and footer regions are used to provide corporate address, copyrights, contact information, terms and conditions, company logo, and so on. You can also define a default cover letter and set it so it cannot be updated.

If these regions are enabled, then while creating cover letters, users can select a live and approved version of the content from the repository based on their permissions.

Body

You can enter the HTML text for emails in this region. For non-HTML compliant users, you can display the content in text format. If the content type has a query associated with it, then you can choose to embed the merge fields. You can select one or more merge fields from the associated query to embed in the cover letter. These are written in SQL (Structured Query Language) format. Query items are stored in Oracle Content Manager (OCM) as cover letters of type Query.

Typically, queries are created and modified by the Marketing Administrators. For more information on Queries, see *Oracle Marketing User Guide*.

Attachments

You can send attachments of any file type with emails and faxes. But only .rtf and .pdf file attachments can be printed.

In emails, you can send binary files as attachments. You can specify the attachment details in this region. If this region is enabled, then while creating cover letters users can search the attachments based on cover letter name, description, and that are associated with Products or Offers. The search returns all cover letters that are live and approved from the repository, based on the permissions that you have.

Queries

Queries are Structured Query Language (SQL) statements that generate selected data from a database. A query determines which records meet the defined criteria, and returns the matched records.

Queries are the source for the mail merge fields that can be embedded in the cover letter body. They are used to generate personalized tags for mail merge, which are fulfilled by Oracle One-to-One Fulfillment, at the time of execution. The Queries page lists all the queries. You can select an existing query to view the details or click the Create button to create a new query.

For details on queries, see *Oracle Marketing User Guide*.

Use the following procedure to create and update queries.

Creating and Updating Queries

Prerequisites

None

Steps

1. Login with the One-to-One Fulfillment Administrator responsibility and navigate to Content > Query.
2. Click **Create**.
The Create Query page appears.
3. In the Name field, enter a unique name for the query
4. Optionally, in the Description field, enter a description for the query
5. In the Query SQL field, enter the SQL SELECT statement.

It is not necessary to a semi-colon (;) at the end of the SELECT statement.

6. Click **Create**.

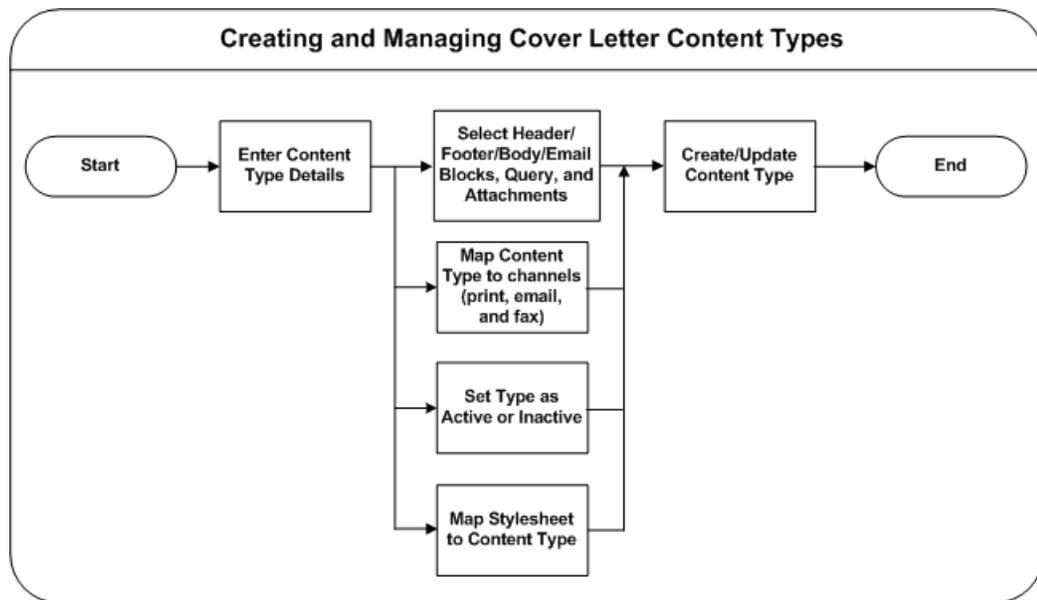
The query is created with a status of In Progress.

7. To update a query, select the query and click **Update**.

- You can update the queries with status In Progress and Submit for Approval.
- You cannot update the queries with status Approved.
- In the Update Query page, update the required fields and click **Update**.

Creating and Managing Cover Letter Content Types

Only administrators can create content types. The following figure depicts the process to create a cover letter content type.



The preceding figure illustrates the following process:

- Map the content type to different groups such as email, fax, and Print.
- Define the requirements for the cover letters, such as header, body, email blocks, queries, and footer.
- Define if the blocks must be displayed and set the default values.
- Define if the content can be updated.

- Create active content types so that they are available to create cover letters.
- Specify the default style sheet for the content type.

The seeded style sheet (provided by OCM) is selected by default.

Note: Oracle recommends that you understand the requirements of your organization before you define the attributes for the content type. After a content type is used to create a cover letter, you cannot update it, you must first duplicate it and update the copy.

Use the following procedure to create and manage cover letter content types.

Prerequisites

None

Steps

1. Login with the One-to-One Fulfillment Administrator responsibility and navigate to Content > Content Type.
2. Click **Create**.
The Create Content Type page appears.
3. In the General region, enter details in all the mandatory fields.
 1. Select the status to be active so that it is available to create cover letters.
 2. Select the groups for which the content type must be available.
 3. Select a default style sheet.
4. To define Header attributes for the content type, select the appropriate Display check boxes.
 1. To default the content in the Header Block and Header Image regions, select a value from the Search list.
 2. Select a value for the Updateable field from the drop-down list. While creating a cover letter, this field indicates if the default value can be updated by the user or not.
 3. Select a value for the Required field from the drop-down list. While creating a cover letter, this field indicates if this field must be a mandatory field.

5. To select a query for the mail merge in the body of the cover letter, select a query from the Search list.
6. To define a content block in the Content Blocks region, select a value from the Search list.
 1. To default the content in the Content Block, select a value from the Search list.
 2. Select a value for the Updateable field from the drop-down list. While creating a cover letter, this field indicates if the default value can be updated by the user or not. This is applicable for all the fields in other regions also.
 3. Select a value for the Required field from the drop-down list. While creating a cover letter, this field indicates if this field must be a mandatory field. This is applicable for all the fields in other regions also.
7. To define Footer attributes for the Content Type, select the appropriate Display check boxes.
 1. To default a content in the Footer Block and Footer Image regions, select a value from the Search list.
 2. Select a value for the Updateable field from the drop-down list.
 3. Select a value for the Required field from the drop-down list.
8. To define an attachment in the Attachments region, select a file from the Search list.
 1. Select a value for the Updateable field from the drop-down list.
 2. Select a value for the Required field from the drop-down list.
9. Optionally, to preview the selected attribute, click the Preview icon.
10. Click **Save**.

The cover letter content type is created.
11. To update a content type:
 1. Select the content type and click **Update**.

The Create Content Type page appears displaying the content type details.
 2. Update the required fields and click **Apply**.

The content type is updated.
12. To duplicate a content type Select the content type and click **Duplicate**.

A duplicate copy of the content type is created.

13. To delete a content type Select the content type and click **Delete**.
The content type is deleted.

Creating Cover Letters

To create cover letter contents you must:

1. Select a cover letter content type from the Create list and click the Go button.
2. Provide content details.

Note: The content type, which you select determines the content details that you must provide.

Selecting a Content Type

You must select a content type from the Cover Letter Summary page. In this page, select a content type from the Create list and click the Go button. In this page you also:

- Sort content types based on the Content Type Group.
- List all the cover letters that are created using a particular content type.
- Search cover letters based on the owner, name, location, last update date, version, and description.

Cover letter contents are stored in folders. For more information on folders and permissions, see *Oracle Content Manager Implementation and Administration Guide*.

Use the following procedure to create cover letters.

Prerequisites

A content type must exist.

Steps

1. Login with the One-to-One Fulfillment Administrator responsibility and navigate to Content > Cover Letter.
2. Select a content type from the Create list and click **Go**.
The Create Content page appears.

3. Enter name, description, owner, and location details in the respective fields.
4. In the Header region, select the values for content block and content image from the Search list.
 1. This region is enabled only if it is defined while creating the selected content type. You can select cover letters of type Image or Content Blocks from a list of values. The search result lists all the cover letters that you have access, which are live versions.
 2. If this region is enabled, then you can select images or content blocks for the body header.
 3. While defining the content type, if the fields are defined as non-updatable fields, then you cannot update them. You can only view the defined values.
 4. To add more content blocks and images, click **Add another Row**. The order in which you add the contents are maintained during execution.
5. In the Body region, enter the text for the email and fax body.
 1. You can enter the content for the email in an HTML format. If the email is for non-html clients, you can choose to display the email content in text format by clicking **Edit Text Version**. The content created for Fax groups will display the content in HTML format.
 2. If the content type has any query associated with it, you can insert mail merge fields based on the available fields, either individually or all at once.
 3. Users can insert mail merge fields based on the available fields.
 4. To add the query mail merge fields individually, select the query and click <. Optionally, to add all the queries for mail merge, click the double < symbol.
6. In the Content Blocks region, select a content block from the Search list.

You can add another content block in the Content Block region by clicking the Add Another Row button and selecting a content block from the Search list.
7. In the Footer region, select the values for content block and content image from the Search list.

You can add another content block in the Footer region by clicking the Add Another Row button and selecting a content block from the Search list.
8. In the Attachments region, select the file for the attachment from the Search list. All the files in the repository are displayed.

1. You can search for attachments that are associated with Products or Offers, which are part of the campaign schedule. The search result lists all the cover letters that you have access and which are live versions.
 2. You can also upload attachments from the desktop. By default, the attachments are saved in the same folder as the cover letter. You can select a different folder for which you have the Create Item permission.
 3. You can add another content block by clicking the Add Another Row button and selecting a content block from the Search list.
9. In the Content Renditions region, you can upload .pdf file for email content.
- This region is enabled only for Print groups.
 - You can upload .pdf or .rtf files for as the print renditions for cover letters. The .pdf or .rtf files may contain merge fields which are related to the query associated with the cover letter.
10. Optionally, click **Save for Later**.
- The cover letter is created with the status **In Progress**. This option enables you to create an item, and update it later.
11. Optionally, to preview the content, click **Preview**.
- You can preview a content, only if you have selected the **Save for Later** option.
 - You can also preview the content in text format.
12. Click **Submit for Approval**.
- The cover letter is submitted for approval.

Submitting for Approvals

After submitting the cover letter content for approval, one of the following happens:

- The cover letter is submitted for approval based on the status of the cover letter or any of the attachments.
- If any of the attachments are in Rework status or In Progress status, you cannot submit the cover letter content for approval.
- If all the attachments are approved, then the content goes through OCM Content Approval process. The status of the cover letter content changes to Submitted. You cannot update the content in this status.
- A cover letter is approved only after all its components are approved.

Searching for Cover Letters

In the Cover Letter Summary page, you can narrow the search based on the content type groups, that is, email, fax, and print. You can also search for cover letters based on the parameters such as cover letter name, description, folder location, and last update

The search returns all live versions of the cover letters for which you have read access permissions. The live versions are the latest approved versions of the item. There may be later versions of the item, which are In Progress status. You cannot update such items.

Note: You must not edit the cover letter content types and cover letters in the generic OCM UI as the attributes may not be displayed correctly.

Managing Fulfillment Requests

Use the Status sub tab to view the status of fulfillment requests that have not been completely processed.

Tasks

You can perform the following tasks:

- Viewing the Status of a Fulfillment Request, page 6-42
- Canceling a Fulfillment Request, page 6-43
- Pausing or Resuming a Fulfillment Request, page 1-16

Viewing the Status of a Fulfillment Request

Use the following procedure to view the status of a fulfillment request.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Status tab.

2. Click the Status sub tab.

The Request Status page appears.

3. From the View list, select a view type.

You have the following options:

- Submitted: Requests that have been submitted to the Advanced Queue but not yet picked up by the Fulfillment Server.
- Re-submitted: Original Fulfillment requests that have been resubmitted to the Advanced Queue.
- In Process: Requests being processed.
- Previewed: This option is no longer in use.

4. Click a request identifier link.

The View Status page appears.

Canceling a Fulfillment Request

Use this procedure to cancel a fulfillment request that is in the queue or is being processed.

Note: With the exception of mass requests, when the Fulfillment Server retrieves a fulfillment request from the request queue, the fulfillment request cannot be canceled.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Status tab > Status sub tab.
The Request Status page appears.
2. From the View list, select All, Submitted, Re-Submitted, or In Progress.
3. In the Action column of the request, click **Cancel**.
The page refreshes and the canceled request is no longer available.
To view the canceled request, see *Managing Fulfillment Request History*, page 6-45

Pausing or Resuming a Fulfillment Request

Use this procedure to pause or resume a fulfillment request that is in progress.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Status tab > Status sub tab.
The Request Status page appears.
2. From the View list, select **In Progress**.
3. In the Pause/Resume column, do one of the following:
 - To temporarily stop a request, click **Pause**.
 - To continue a paused request, click **Resume**.
To view any jobs in the request that have already been processed, see *Managing Fulfillment Request History*, page 6-45.

Managing Fulfillment Request History

Use the History sub tab to view the history of fulfillment requests that have been processed, successfully or unsuccessfully, by a Fulfillment Server.

Tasks

You can perform the following tasks:

- Viewing the Details of a Fulfillment Request, page 6-45
- Viewing the Details of a Fulfillment Request, page 6-45
- Viewing Failed Fulfillment Requests, page 6-48
- Viewing a Partially Successful Fulfillment Request, page 6-48
- Correcting a Malformed Email Job, page 6-49
- Viewing Job Details, page 6-50
- Resubmitting a Fulfillment Request, page 6-53
- Resubmitting a Job, page 6-52
- Deleting a History Entry, page 6-53
- Searching for a Fulfillment Request, page 6-54

Viewing the Details of a Fulfillment Request

Use this procedure to view the details of a fulfillment request.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Status tab > History sub tab.
The Request Summary page appears.
2. From the View list, select which status of requests to view.
 - All
 - Cancelled
 - Failed
 - Partially Successful
 - Stalled
 - Successful

The page refreshes and lists all requests of the selected status.

- Request Identifier: Request identification number
- Server Name: Name of the server that processed the request
- Agent: Name of the agent who submitted the request
- Submitted: Date and time that the request was submitted
- Completed: Date and time that the request was completed
- Outcome: Outcome of the request: Successful, Failed, Partially Successful, Cancelled, Stalled, Email Sent, Writing Log
- Parent Request Identifier: Request identification number and link to the original (parent) request
- Request Type: Single or Test
- Channel: Email, Print, Fax, or Multi (such as fax and email)

The Statistics group lists the following details.

- Fax
 - Sent: Number of fax messages sent
 - Not Sent: Number of fax messages not sent

- Print
 - Sent: Number of print items sent to the printer
 - Not Sent: Number of print items not sent to the printer

- Email
 - Sent: Number of email messages sent
 - Opened: Number of email messages opened by the recipient. (The Count is displayed only if the relevant profile, Email Tracking is enabled.)
 - Unopened: Number of email messages not yet opened by the recipient
 - Not Sent: email message prepared but not sent
 - Malformed: Number of email messages with invalid addresses
 - Do not contact: Number of email messages whose recipients are flagged "Do not contact" in the Customer Database
 - Other Failures: Number of email messages that could not be sent because of other reasons
 - Bounced: Number of bounced email messages received for this request
 - Unsubscribed: Number of email recipients who chose to unsubscribe from the email recipients list
 - Resubmitted Malformed: Number of malformed email messages that have been corrected and reprocessed

- The Content Details group lists information about the content:
 - Content ID
 - Content Number, the total number of content items. For example, a cover letter plus an attachment would be listed in two separate rows in the Content Details table. Content Number 1 would be the main content (the HTML or RTF form document) and Content Number 2 would be the static attachment.
 - Content Name, with a link to view the content, such as an HTML or RTF file
 - Content Type, such as a query, HTML rendition, RTF rendition, collateral

or data

- Store, such as Oracle MarketView, Marketing Encyclopedia System, or Oracle *iContent* (Oracle Content Manager)

3. Click the number of a Request Identifier.

The Request Details page appears.

Viewing Failed Fulfillment Requests

Use this procedure to see a list of failed requests.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Select the Status tab > History sub tab.

The Request Summary page appears.

2. From the View list, select **Failed**.

The page refreshes and lists the requests marked as failed in the Outcome column.

3. Click a Request Identifier number.

The Request Details page appears. The Failure Details group shows the Outcome Description.

Viewing a Partially Successful Fulfillment Request

Use this procedure to see a partially successful fulfillment request.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Select the Status tab > History sub tab.
The Request Summary page appears.
2. From the View list, select **Partially Successful**.
The page refreshes and lists the partially successful requests.
3. Click the number of a Request Identifier.
The Request Details page appears.
4. In the Statistics group, click the linked number of the Not Sent items.
The Failures page appears, listing each failed job and the reason for failure.

Correcting a Malformed Email Job

Use this procedure to correct an improperly addressed email request.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Select the Status tab > History sub tab.
The Request Summary page appears.
2. Click the Request Identifier number of the malformed email request.
The Request Details page appears.
3. In the Statistics group, click the number that corresponds to Malformed.
The Resubmit Malformed Email Jobs page appears. The incorrect address is in the Email Address column.
4. In the Correct Email Address field, enter the email address.
5. Click **Resubmit**.
The Request Summary page appears.

A new single Fulfillment request will be submitted for each corrected email that is resubmitted. The original request identifier appears in the Parent Request Identifier of the Request Summary page.

Viewing Job Details

Use this procedure to see job details, such as customer information, content, and failure details.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Submit at least one job.

Steps

1. Select the Status tab > History sub tab.
The Request Summary page appears.

2. Click the number of the Request Identifier.

The Request Details page appears.

3. In the Job Details group, click the number of the Job.

The Job Details page appears, listing the following information groups.

- Customer Information:
 - Customer Identifier
 - Customer Name
 - Email Address: Email address of the recipient
 - Fax Number
 - Printer Address: Name of the printer to which the request was sent
- Content Details:
 - Content ID
 - Content Number, the total number of content items. For example, a cover letter plus an attachment would be listed in two separate rows in the Content Details table. Content Number 1 would be the main content (the HTML or RTF form document) and Content Number 2 would be the static attachment.
 - Content Name, with a link to view the content, such as an HTML or RTF file
 - Content Type, such as a query, HTML rendition, RTF rendition, collateral or data
 - Store, such as Oracle MarketView, Marketing Encyclopedia System, or Oracle iContent (Oracle Content Manager)
- Failure Details (Only applicable if the request was partially successful):
 - Channel: Email, Fax, and Print
 - Address: Email address, fax number, or printer
 - Reason: Cause for the failure

Resubmitting a Job

Use this procedure to resubmit a job that was sent unsuccessfully.

Note: For Failed requests, if the Request Details page does not list Job Details, the request has failed before individual jobs could be specified. In such cases, it is not possible to resubmit jobs because none have been created.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

Submit at least one job.

Steps

1. Select the Status tab > History sub tab.
The Request Summary page appears.
2. Click the Request Identifier number of the malformed email request.
The Request Details page appears.
3. In the Job Details group, click the linked number of the Job.
The Job Details page appears.
4. Click **Resubmit**.
The Resubmit Job page appears.
5. From the Channel list, select Email, Fax, or Print.
6. In the Address field, enter the email address or fax number of the recipient, or enter the printer name.
7. Click **Resubmit**.

Resubmitting a Fulfillment Request

Use this procedure to resubmit a failed fulfillment request.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Status tab > History sub tab.
The Request History page appears.
2. In the Action column, click **Submit**.

Deleting a History Entry

Use this procedure to delete a history entry for a fulfillment request.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Status tab > History sub tab.

The Request Summary page appears.

2. Optionally, from the View list, select which status of requests to view.
 - All
 - Cancelled
 - Failed
 - Partially Successful
 - Stalled
 - Successful

The page refreshes and lists all requests of the selected status.

3. Click **Select** for the history entry that you want to delete.
4. Click **Delete**.

The selected history entry is removed from the list and the Request History page refreshes.

Searching for a Fulfillment Request

Use the Search sub tab to search for a fulfillment request that has been processed by a Fulfillment Server.

Use this procedure to search for a fulfillment request.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps:

1. Click the Status tab > Search sub tab.

The Search page appears.

2. From the list, select the search type:
 - Customer Name
 - Customer Identifier
 - Request Identifier
3. In the field, type the value for which to search.
4. Click **Search**.

The list of search results displays the following information about each fulfillment request.

- The results display a summary of the Customer Identifier, Customer Name, Request Identifier, Job, and Job Outcome. Users can click the History Request Identifier number to view the Request Details, which displays a summary of the request (server name, submitted & completed times, agent who initiated the request, and the outcome code, statistics, content, and job summary).
- Users can click the job number to view the job details, which displays a summary of the job (customer information, content details, and any applicable failure details).

Using the Quick Find Feature

Use the Quick Find feature to find a Fulfillment-related item, such as a server or template.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps:

1. Select any tab in the administration console.
2. From the Quick Find list, select the search type:
 - Group (fulfillment agents)
 - Master Document
 - Query
 - Server (1)
 - Status
 - Template
3. In the Quick Find field, enter the search criterion. The criteria are case sensitive, except for Request Status which must be typed in all capital letters.

The Request Status types are *(type capital letters only)*:

- SUBMITTED
- RESUBMITTED
- PAUSED
- PREVIEWED
- RESUMED
- CANCELLED

You can search for an item by using character strings and the wildcard symbol (%). For example, to find servers starting with A, type "A%".

4. Click **Go**.
The Search Results page appears.
5. To view the details of an item in the search results, click the item link.
The Details page appears.

Using Merge Fields

The Fulfillment Server code uses sets of merge field delimiters to determine whether a

document contains merge fields. It searches for the left delimiter of each pair within the document text. When it finds a left delimiter, it tries to then pair it with a right delimiter. If it does not find the right delimiter, the next pair of delimiters is searched for, and so on, until either a complete merge field is found or the list of merge delimiters is exhausted.

Topics include:

- Default and Wildcard Merge Delimiter Pairs, page 6-57
- Different Representations of the Same Character, page 6-57
- Using Encrypted Merge Fields, page 6-58
- Creating an RTF Form Document with Merge Fields, page 6-58
- Creating a PDF Form Document with Merge Fields, page 6-68

Default and Wildcard Merge Delimiter Pairs

There are three sets of merge delimiters, the first of which is a wildcard pair. This first set can be overridden with the Java property flags 'merge.LeftDelimiterString' and 'merge.RightDelimiterString' (Java properties must be preceded with '-D' when part of the java command). You can also use the profile options JTF_FM_LMERGE_DELIM and JTF_FM_RMERGE_DELIM to select from a list of seeded merge delimiters. The other two sets are static and therefore always a part of the list. The list includes:

Left Delimiter	Right Delimiter	Description
«	»	ASCII alphabetic delimiter codes. These are the defaults for the wildcard pair.
«	»	ASCII numeric delimiter codes.
'\u00AB'	'\u00BB'	Unicode delimiter codes.

Some of these delimiters begin with an ampersand '&' and end with a semi-colon ';'. These are special characters that are recognized by HTML. In the case of '«', it is interpreted by HTML as a character having the decimal numeric value of 171.

Different Representations of the Same Character

The Fulfillment Server allows multiple merge delimiters due to the way these characters

are represented. Unicode characters are represented in hexadecimal values. HTML uses decimal values. So the left-pointing double angle quotation mark (guillemet) can be represented in decimal as "«" or in Hexadecimal as '\u00AB' or even in Octal as '\253'. The Octal representation in Unicode is '\u00AB'. Some log files contain '\253' and '\273'. The Fulfillment Server needs to have each of these representations because it parses characters and character sequences. When you see the character a left-pointing double angle bracket in your Web browser, it might be represented by the character sequence « in the HTML source. The fulfillment server parses this source and therefore parses the sequence '&' followed by '#' followed by '1' and '7' and '1' and ';'. The same is true for any character sequence set with the flags 'merge.LeftDelimiterString' and 'merge.RightDelimiterString'.

The following table lists the different numeric character conversions of the default merge delimiters that are used by the Fulfillment Server.

Character	Decimal	Hexadecimal	Octal
left-pointing double < bracket (guillemet)	171	AB	253
right-pointing double > bracket (guillemet)	187	BB	273

Using Encrypted Merge Fields

Oracle One-to-One Fulfillment has encrypted merge fields. When an encrypted merge field is found, the server encrypts the merge value and then leaves the delimiters in place so that it can later be recognized and decrypted. The default string delimiters are:

- Left delimiter string `"*encrypt("`
- Right delimiter string `")*"`

You can override these strings with the Java property flags 'merge.LeftEncryptDelimiter' and 'merge.RightEncryptDelimiter'. You can also use the profile options JTF_FM_LENCRYPT_DELIM and JTF_FM_RENCRYPT_DELIM which only allow you to select from a list of seeded values.

Creating an RTF Form Document with Merge Fields

Use the following procedure to create Rich Text Format (RTF) form documents, containing merge fields, to use with the Oracle One-to-One Fulfillment native print function.

Login

Not applicable

Responsibility

Not applicable

Prerequisites

None

Steps

1. In Microsoft Word, open a Word document. The document must have all necessary text, merge field references, and images.
2. Put the cursor on the highest Word toolbar, and click to open the mouse menu.
3. Select **Forms**.
The Forms toolbar appears.
4. Highlight the first text that you want to be a merge field.
5. On the Forms toolbar, click **ab|** (Text Form Field).
The text is highlighted.
6. Double click the highlighted text.
The Text Form Field Options box appears.
7. In the Default text field, enter the display name of the form field.
8. Click **Add Help Text**.
The Form Field Help Text box appears.
9. In the text field, enter the name of the merge field. The merge field name must be exactly the same the name or alias that is used in the SQL query and must be enclosed within "<?" and "?>", for example, <? FIRST_NAME ?>. The merge delimiters must always be in upper case.
10. Click **OK**.
The Form Field Help Text box closes.
11. In the Text Form Field Options box, click **OK**.

The Text Form Field Options box closes. The highlighted text form field name appears.

12. For each merge field, repeat steps 4 through 11 above, page 6-59.
13. Save the document as a Rich Text Format file with an rtf file extension.

Creating a Table with a Non-Repeating Row

The following table is an example of a table containing a non-repeating row.

Contact	Email Address	Status of Email	Date Processed
NAME	EMAIL_ADDRESS	STATUS	SENT_DATE

To include non-repeating data within an RTF document, use the following procedure.

Steps

1. Insert a table, selecting the appropriate number of columns and one or two rows.
2. If required, enter the appropriate column header names.
3. Insert a text form field into the first cell of the second row (assuming there are column headers in the first row). To insert merge fields, see steps 4 through 11 above, page 6-59.
4. Repeat until there is a text form field in each cell of the row.

The following table is an example of how the table containing a non repeating row would appear in the printed document.

Contact	Email Address	Status	Date Processed
CHERYL LAWRENCE	Cheryl.Lawrence@vi sion.com	SENT	21-JAN-2003

Creating a Table with Repeating Rows

A table with repeating rows of data enables multiple rows of data to be returned by a SQL query and presented in a table. The following table is an example of a table that contains repeating rows.

Contact	Email Address	Status of Email	Date Processed
row repeat NAME	EMAIL_ADDRESS	STATUS	SENT_DATE row repeat

Steps

1. Insert a table, selecting the appropriate number of columns and one or two rows.
2. If required, enter the appropriate column header names.
3. Insert a text form field into the first cell of the second row. To insert merge fields, see steps 4 through 11 above, page 6-59.
4. Repeat until there is a text form field in each cell of the row.
5. Add a new form field at the start of the second row (assuming there are column headers in the first row).
6. On the Forms toolbar, click **ab|** (Text Form Field).
The text is highlighted.
7. Double click the highlighted text.
The Text Form Field Options box appears.
8. In the Default text field, enter the display name of the form field, for example "row repeat".
9. Click Add Help Text.
The Form Field Help Text box appears.
10. In the text field, use the for-each:<group name> command (surrounded by the <? and ?> delimiters) to indicate the start of the repeating group (in this case, a row), for example, <?for-each:row?> .
11. Click **OK**.
The Form Field Help Text box closes.
12. In the Text Form Field Options box, click **OK**.
The Text Form Field Options box closes. In the Word document, the highlighted text form field name appears.

13. Add a new form field at the end of the second row (assuming there are column headers in the first row).
14. On the Forms toolbar, click **ab|** (Text Form Field).
The text is highlighted.
15. Double click the highlighted text.
The Text Form Field Options box appears.
In the Default text field, enter the same display name for the form field that was used at the start of the row, for example "row repeat"
16. Click **Add Help Text**.
The Form Field Help Text box appears.
17. In the text field, use the end for-each command (surrounded by the `<? and ?>` delimiters) to indicate the end of the repeating group (in this case, a row), for example, `<?end for-each?>`.
18. Click **OK**.
The Form Field Help Text box closes.
19. In the Text Form Field Options box, click **OK**.
The Text Form Field Options box closes. The highlighted text form field name appears.
The following table is an example of how the table containing repeating rows would appear in the printed document.

Contact	Email Address	Status	Date Processed
CHERYL LAWRENCE	Cheryl.Lawrence@vision.com	SENT	21-JAN-2003
CHERYL LAWRENCE	Cheryl.Lawrence@vision.com	SENT	29-JAN-2003
BRYAN WEBB	BRYAN.WEBB@vision.com	SENT	21-jan-2003
MICHAEL Z ROBERTSON	MICHAEL.ROBERTSON@vision.com	NOT SENT	21-jan-2003

Contact	Email Address	Status	Date Processed
MICHAEL Z ROBERTSON	MICHAEL.ROBERT SON@vision.com	SENT	28-jan-2003

Creating a Repeating Table

A repeating table can have multiple tables of data returned by a SQL query. The following table is an example of repeating rows.

mytable group

Contact	Email Address	Status	Date
NAME	EMAIL_ADDRESS	STATUS	SENT_DATE

mytable group

Steps

1. Insert a table, selecting the appropriate number of columns and one or two rows.
2. If required, enter the appropriate column header names.
3. Insert a text form field into the first cell of the second row, by following steps 4 through 11 above, page 6-59.
4. Repeat until there is a text form field in each cell of the row.
5. Add a new form field just above the table .
6. On the Forms toolbar, click **ab|** (Text Form Field).
The text is highlighted.
7. Double click the highlighted text.
The Text Form Field Options box appears.
8. In the Default text field, enter the display name of the form field, for example "mytable group".
9. Click **Add Help Text**.

The Form Field Help Text box appears.

10. In the text field, use the `for-each:<group name>` command (surrounded by the `<? and ?>` delimiters) to indicate the start of the repeating group (in this case, a table), for example, `<?for-each:mytable?>`.

11. Click **OK**.

The Form Field Help Text box closes.

12. In the Text Form Field Options box, click **OK**.

The Text Form Field Options box closes. In the Word document, the highlighted text form field name appears.

13. Add a new form field just after the table.

14. On the Forms toolbar, click **ab|** (Text Form Field).

The text is highlighted.

15. Double click the highlighted text.

The Text Form Field Options box appears.

16. In the Default text field, enter the same display name for the form field that was used at the start of the row, for example "mytable group".

17. Click **Add Help Text**.

The Form Field Help Text box appears.

18. In the text field, use the `end for-each` command (surrounded by the `<? and ?>` delimiters) to indicate the end of the repeating group (in this case, a table), for example, `<?end for-each?>`.

19. Click **OK**.

The Form Field Help Text box closes.

20. In the Text Form Field Options box, click **OK**.

The Text Form Field Options box closes. In the Word document, the highlighted text form field name appears.

The following example shows how repeating tables would appear in a printed document.

Contact	Email Address	Status	Date
---------	---------------	--------	------

Contact	Email Address	Status	Date
CHERYL LAWRENCE	Cheryl.Lawrence@vision.com	SENT	21-JAN-2003

Contact	Email Address	Status	Date
CHERYL LAWRENCE	Cheryl.Lawrence@vision.com	SENT	29-JAN-2003

Contact	Email Address	Status	Date
BRYAN WEBB	BRYAN.WEBB@vision.com	SENT	21-jan-2003

Contact	Email Address	Status	Date
MICHAEL Z ROBERTSON	MICHAEL.ROBERTSON@vision.com	NOT SENT	21-jan-2003

Contact	Email Address	Status	Date
MICHAEL Z ROBERTSON	MICHAEL.ROBERTSON@vision.com	SENT	28-jan-2003

The following examples are the repeating and non-repeating data scenarios that are supported by the RTF print feature:

- If "for-each" and "end for-each" appears in normal text, then the text between the tags/form fields will be repeated.
- If "for-each" and "end for-each" appears at a table's or multiple tables' start and end points, then the included table(s) will be repeated.
- If "for-each" and "end for-each" appears in a single table cell, then the text between the tags/form fields in the table cell will be repeated.

- If "for-each" and "end for-each" appears in two different cells but on the same row, then the single row will be repeated.
- If "for-each" and "end for-each" appears in two different table rows, then the rows between the tags/form fields will be repeated.

Conditions

The following examples show conditions with tables or text to enable information to be displayed *only* if the condition is true.

```
if references is false
```

Reference Document	Document Type	Date Required
row repeat DOC_NAME	DOC_TYPE	DATE_REQD row repeat

```
end if
```

```
Text Reference is False
if references is false
```

```
REQD_REF_TEXT
```

```
end if
```

Use the following procedure to display a table if a particular condition is true.

1. Insert a table, selecting the appropriate number of columns and rows.
2. If required, enter the appropriate column header names.
3. Insert a text form field into the first cell of the second row. To insert merge fields, see steps 4 through 11 above, page 6-59. Repeat until there is a text form field in each cell of the row.
4. Add a new form field just above the table or text.
5. On the Forms toolbar, click **ab|** (Text Form Field).
The text is highlighted.
6. Double click the highlighted text.

The Text Form Field Options box appears.

1. In the Default text field, enter the display name of the form field, for example "if

references is false".

2. Click Add Help Text.

The Form Field Help Text box appears.

3. In the text field, use following condition command format to evaluate when the condition is true: <?if:<merge field><condition>'<value>'?>.

For example, <?if:reference='false'?>.

4. Click OK.

The Form Field Help Text box closes.

5. In the Text Form Field Options box, click OK.

The Text Form Field Options box closes. In the Word document, the highlighted text form field name appears.

6. Add a new form field just after the table or text.

7. On the Forms toolbar, click **ab (Text Form Field).**

The text is highlighted.

8. Double click the highlighted text.

The Text Form Field Options box appears.

9. In the Default text field, enter the same display name for the form field that was used at the start of the row, for example "end if".

10. Click Add Help Text.

The Form Field Help Text box appears.

11. In the text field, use the end if command (surrounded by the <? and ?> delimiters) to indicate the end of the conditional text or table, for example, <?end if?>.

12. Click OK.

The Form Field Help Text box closes.

13. In the Text Form Field Options box, click OK.

The Text Form Field Options box closes. In the Word document, the highlighted text form field name appears.

The following examples show condition commands used around tables or text.

The following example shows how the table appears if the condition is true.

Reference Document	Document Type	Date Required
United Kingdom Student Visa	Copy	5/18/04
United States Passport	Copy	5/18/04
High School Certificate	Original	3/27/04
Degree Certificate	Original	3/27/04

The following example shows how the text appears if the condition is true.

Your application for the Bachelor of Arts in Fine Arts, Full Time study at the Belmont Campus requires that your Transcripts be submitted by June 1st, 2004. Please submit this item directly to the Admissions Office.

Creating a PDF Form Document with Merge Fields

Use the following procedure to create merge fields in an Adobe Acrobat PDF form document. Oracle One-to-One Fulfillment supports only Adobe Acrobat Version 5.

Login

Not applicable

Responsibility

Not applicable

Prerequisites

None

Steps

1. Do one of the following:
 - If the document is already a PDF file, then proceed to step 4.
 - If the document is a Microsoft Word file, then proceed to step 2.
2. In Microsoft Word, open the document that you want to use as a Fulfillment template. The document must not contain merge fields.

3. In the Word menu, select **Acrobat** or **Adobe PDF >Convert to PDF**.
The Acrobat progress indicator shows the status of the conversion of the Word file to a PDF file.
4. In Adobe Acrobat, open the PDF file.
5. On the Acrobat toolbar, click the Form Tool.
6. Place the cursor where you want create the first merge field, then click and drag the Form Tool to the size that you want the merge field to be.
The Field Properties box opens.
7. In the Name field, enter the name of the merge field. The name must be exactly the same as the form field name or alias in the SQL query, such as FIRST_NAME. The merge delimiters must always be in upper case.
8. In the Appearance tab, select the text options for this field.
9. In the Options tab, select the text alignment.
10. Click **OK**.
11. For each merge field that you want to add, repeat steps 3 through 10.
12. Save the file.

Setting Up High Quality Printing

Use the following procedures to set up high quality Oracle printing.

You can do the following tasks:

- Configuring the Printer, page 6-3
- Setting Up a Pasta Print Server, page 6-3
- Verifying the Printer Configuration, page 6-3

Configuring the Printer

Use the following procedure to configure the printer for high quality Oracle printing.

Login

Not applicable

Responsibility

Not applicable

Prerequisites

None

Steps

1. Verify that you can print from the Fulfillment Server host. Create a test document such as a text file called `hello.txt` that contains simple text.
2. Identify the printer. You may need to ask the network administrator or system administrator the proper name or destination.
3. Use the proper executable to print the file `hello.txt`.
 - For UNIX or Linux, use the `lp` command.
Example usage: `lp -d loc2-prt1-rm108 hello.txt.`
 - For Windows, use the `print` command.
Example usage: `PRINT /D:loc2-prt1-rm108 hello.txt.`
4. Verify that the document printed successfully. The High Quality Printing Solution depends on printing by using the operating system command.
5. Download and install the third party tool XPDF. Depending on your operating system, you may need to download source and build an executable, although most operating systems already have images built.
6. Save the configuration.
7. Test the configuration. Using the same printer name that you used for the `lp` or `print` command, print a PDF file using `FNDPSTAX`. The `-pn` parameter is where you specify the printer name.

Setting Up a Pasta Print Server

Use the following procedure to set up a Pasta print server.

Login

HTML Login URL

Responsibility

One-to-One Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab, then a server name link, then click Print Servers.
2. Click **Create New Print Server**.
Print servers are primarily used for RightFax Printing, but are also used in High Quality Printing as a way to identify which printers are Pasta Printers.
3. In the Print Server Name field, enter a generic name such as Pasta Print Server.
4. From the Server Type list, select **Pasta**. For Pasta printers, no Connection String is necessary.
5. From the main Print Servers table, select the Pasta Printer.
6. On the Detail page, select **Create Printer** to add your Pasta printer to this print server grouping.
7. In the Printer Name field, enter a label or a user-friendly name that users will recognize. For example, if this printer is the printer installed in room 108, enter Room 108 Pasta Printer.
8. Optionally, enter a Description.
9. In the Network Name field, enter the same name that you used to print using lp or print and FNDPSTAX. This name is passed to the call to Pasta.
10. Click **Apply**.
11. If you plan to configure more than one Pasta printer, test each one using the previous steps before adding it to the Pasta Print Server that you created.

Verifying the Printer Configuration

After you have successfully tested and configured the printer, run a Fulfillment request. First create and upload the RTF template. When you have identified the document, submit a normal print request. For the printer address parameter, use the Network

Name field from the Administration Console setup.

If you have installed the necessary 11.5.10 Fulfillment prerequisites, you will get a high quality printed version of the document that you sent to the printer. If you have not entered your printer Network Name in the Administration Console as a printer connected to a Pasta Print Server, the Fulfillment Server will behave as though the printer address is a RightFax printer. The request will fail if RightFax does not have a printer with the same name as the Network Name.

All documents submitted to a Pasta printer must be either PDF or RTF. If you have attachments such as text or HTML, the print request will fail because the Pasta printer will not accept these document types.

Running Concurrent Programs

JTO: Purge Fulfillment Requests

You run the concurrent program JTO: Purge Fulfillment Requests to purge request history and email statistics data for old requests. This program takes Age - the number of days since the request was last updated - as a parameter, and purges data for such requests.

Note: After this request is run, all request history and email statistics data (count of sent, opened, bounced emails) for the qualifying requests is deleted

Interaction History Bulk Processor

You run the concurrent program Interaction History Bulk Processor for the interaction history to be logged for Mass emails.

To run this program, log in with the CRM Administrator responsibility.

Note: This step is *not* required if the Multithreaded Fulfillment option is disabled.

Fulfillment Server Commands

Fulfillment Server Startup Script Parameters

Topics include:

- [Fulfillment Server Commands, page A-1](#)
- [Required Flags, page A-2](#)
- [Optional Flags, page A-3](#)
- [Debug Flags, page A-6](#)
- [Additional Java Flags, page A-8](#)
- [Language Character Parsing, page A-9](#)

Fulfillment Server Commands

This section describes the commands used to operate a Fulfillment Server when it is running. If the `engine.CommandPromptEnabled` parameter is set to true, then you can enter the command at the same command prompt from which you started the Fulfillment Server.

The following table lists and describes the Fulfillment Server commands.

Command	Description
flush logs	Force the log buffers to write to disk.
help	Display all of the valid commands for the Fulfillment Server.

Command	Description
kill	Stop the server. To restart the server, run the server startup script.
list error <n>	List a specific entry in the error log. The parameter n can be a single number, a range of numbers, or the word count to list the number of entries in the log.
list events <n>	List a specific entry in the event log. The parameter n can be a single number, a range, or the word count to list the number of entries in the log.
mstat	Print statistics about the Fulfillment Server monitors.
pstat	Print statistics about the Fulfillment Server processes.
restart	Stop and automatically restart the Fulfillment Server.
start	Start a stopped Fulfillment Server.
stat	Print statistics about the Fulfillment Server.
stop	Pause Fulfillment Server processing. To resume processing, use the start command.
tstat	Print statistics about the java thread activity in the Fulfillment Server.

Required Flags

The flags listed in the following table are the only flags required to run a Fulfillment Server. The server will not start without them. These flags may be subject to change in future releases depending on changing system requirements.

Flag	Description
engine.AOLJ.config	Specifies the full-path connection of the dbc file that the server uses to get connections to the database.
engine.MaxBatchThreads	Specifies the number of java threads to be used by the server for Multithreaded Fulfillment.

Flag	Description
engine.ServerID	Identifies the Fulfillment Server as referenced in the database. With the server id, the Fulfillment Server can get all configuration from the database without the need for optional flags.
engine.ServerInstanceID	Numeric identifier for the server process. Auto config specifies a unique value for this flag.
engine.TurnOffLowPriorityMonitor	Specifies if Bounceback processing Monitor and the Interaction History Monitor should be started on the current server node.

Optional Flags

The following flags are optional. Certain flags, such as the logging flags, can degrade the performance of the server. Other flags can improve the performance of the server.

Note: Fulfillment Server flags are added or deprecated from time to time, so that this list may change.

Flag	Description
email.PrefersMultipartAlternative	Specifies that the server is to allow a text-only alternative for the DATA or QUERY content that is sent by email. Default is 'false'.
email.TextOnly	Specifies that the server is to send out text-only email instead of the HTML-based email that it sends out for its DATA or QUERY contents. Default is 'false'.
engine.CommandPort	Specifies the numeric port id, on the servers' hardware, that the server will listen to for receipt of remote commands. The only other way to send a command to the server is by the command prompt from which it was started (see engine.CommandPromptEnabled). Default is to not have command port defined.

Flag	Description
engine.CommandPromptEnabled	Specifies that server is to be able to receive commands by the command prompt from which it was started. This flag is useless when running the process in background on UNIX or as a service on Windows NT. For those types of servers, the flag engine.CommandPrompt should be used. The default value for this flag is 'true'.
engine.ErrorLog	Specifies the name of the log used by the server to log errors. The default is 'Errors.log'. Also see 'engine.LogPath' and 'engine.LogLevel'.
engine.EventLog	Specifies the name of the log used by the server to log events. The default is 'Events.log'. Also see 'engine.LogPath'.
engine.FaxEnabler	Indicates the class that the Fulfillment Server is to use for sending faxes. This class must implement the 'oracle.apps.jtf.fm.engine.disp.FaxEnabler' interface to be used as a FaxEnabler. Also see 'engine.PrintEnabler'. No Default value.
engine.LogLevel	Indicates the level of detail that will be written to the error log (see engine.ErrorLog). For each number from 1-n the server will write an extra layer of chained exception traces to the errors log. The exceptions are chained together so that the source of the errors as they occur can be traced.
engine.LogPath	Specifies the directory path where Fulfillment Server logging occurs. The default if this flag is not set is to use 'engine.ServerPath' or 'logs' under the directory from which the server is started if that is also not set.
engine.LogRequests	A Boolean flag that indicates that the original XML-based request pulled from and Advanced Queue by the server is to be written to disk. The requests will be saved as an XML file in the form of #####.xml where '#####' is the request id. The requests will be saved in a directory called 'requests' under the log path. Also see 'engine.LogPath' and 'engine.LogDispatches'. Default is 'false'.

Flag	Description
engine.PrintEnabler	Indicates the class that the Fulfillment Server is to use for sending print jobs to the printer. This class must implement the 'oracle.apps.jtf.fm.engine.disp.PrintEnabler' interface to be used as a PrintEnabler. Also see 'engine.FaxEnabler'.
engine.RefreshRate	Indicates the amount of time, in milliseconds, between processor refreshes. This is when the servers' processor checks the status of all sub-processes to make sure they are functioning correctly.
engine.ResultSetWarningLimit	Indicates how large of a ResultSet can be returned from a database query, before a warning will be logged to the Fulfillment Servers error log. This is to warn the server administrator that in case of an OutOfMemoryError, it might have been caused by the size of the Result Set. *This is not so much a problem in the newer versions of the Fulfillment Server. Also see 'log.ShowWarnings'. Default is '100'.
db.ExitOnFailure	Specifies that the Fulfillment Server is to shutdown when the database becomes unavailable. Default is 'false'.
history.NoContents	Indicates the Fulfillment Server is not to write Content Level History. This is detailed information about the processing and delivery of Fulfillment Content. This flag should be used when content history is not needed but a performance boost is. Also see 'history.NoInteractions'. Default is 'false'
history.NoInteractions	Indicates the Fulfillment Server is not to write Interaction History. This is detailed information about the interaction between the agent and parties addressed in the Fulfillment request. This flag should be used when Interaction History is not needed but a performance boost is. Also see 'history.NoContents'. Default is 'false'.

Flag	Description
pkg.ident	Specifies the Fulfillment Server should write class file versions of the Fulfillment Server to the events log. *This only works when the oracle.apps.jtf.fm.FulfillmentServer class is used to start the Fulfillment Server instead of oracle.apps.jtf.fm.engine.Processor. Default is 'false'.
pkg.showmissing	Works with the 'pkg.ident' flag and specifies the Fulfillment Server note missing class files in the events log. This is useful when wrapping an older version of the Fulfillment Server with the FulfillmentServer class. *This only works when the oracle.apps.jtf.fm.FulfillmentServer class is used to start the Fulfillment Server instead of oracle.apps.jtf.fm.engine.Processor. Also see 'pkg.ident'. Default is 'false'.
log.ShowWarnings	Indicates that warnings are to be logged to the error log. By default, warnings do not show up in the error log. If this flag is not set, then the first time a warning appears, a, one time only, message shows up in the events log signifying that a warning had been issued. Default is 'false'.
smtp.port	Overrides the default port '25' for email sent to an SMTP email server.
smtp.server	Overrides the default SMTP server that is set using the Fulfillment Administration Console.

Debug Flags

The Fulfillment Server can be debugged, at runtime, through the use of a debug flag in the Fulfillment Server startup script or on the command line. The debug information is written to the Fulfillment Server events log, with debug=log, or to Central Logging in the database. In some cases, through use of third-party debugging (for example, Java Mail API), debug information is written to the console.

Note: Oracle advises that you do not try to use debug flags to improve server performance. Using any of these flags reduces the performance of the server by the time it takes for the debug sections to run. In most cases, debug information is either written to the Events log or it is

written to the command prompt.

The correct syntax for this flag is

```
-Ddebug=<flag>[,<flag>],
```

For example,

```
-Ddebug=email,request,content, query
```

This comma-separated list allows for individually setting the debug flags listed in the following table.

Flag	Description
full	Turns on all debug flags, including 'log' which disables Central Logging.
off	Turns off all debugging.
content	Causes request content, as downloaded from the database, to be written to local disk on which the server is running. The directory is called 'contents' and can be found under the log directory.
email	Causes details of the delivery of email to be printed to the Events log and to the console. (Not all information printed to the console is recorded in the Events log.)
fax	Debug the delivery of faxes.
history	Debug the writing of Fulfillment Request History, Fulfillment Content History and Interaction History.
log	Debug the creation of local disk log files. This flag disables the use of Central Logging.
monitor	Debug the operations of the Fulfillment Server's internal monitors. All vital internal processes of the server are handled by these monitors.
print	Debug the sending of print jobs.
processor	Debug the functions of the server main thread.
query	Debug queries executed by the server. These are written to the Events log.
request	Debug requests that are processed by the server. Details of the request are written to the Events log.

Flag	Description
header	Debug extended headers. Each is written as part of the email header for email sent by the server. Additional details of the request are written to the header such as request id, party_id, and so on.
jdbc	Debug JDBC calls made by the server. JDBC is how Java communicates with the database.
fault	Debug the functioning of the server's fault monitor. This provides details of request fault detection and handling.
mass	Debug the processing of Mass Requests.
failure	Debug failures that occur as a normal part of processing a request. The failure is written to the Error log to record a stack trace from where the failure originated. This skews the actual error count because an exception is created to track the stack trace of the failure.
merge	Debug the parsing of merge fields and insertion of bind variables within dynamic request content (Master Documents and email subject field).
queue	Debug the reading and writing to Advanced Queues (AQ). The XML-based requests and responses are written to the Events log.
bind	Debug the use of bind data associated with content of a fulfillment request.
labels	Debug PDF labels requests.

Additional Java Flags

The standard flags of the Java programming language are useful in using the Fulfillment Server to maximum benefit. Some of these flags may not be available for all versions of Java that are used to run the Fulfillment Server. You can view a list of Java flags that work with your version of Java by typing the full path to the java or jre command, as you would use to start the Fulfillment Server.

Flag	Description
-------------	--------------------

Flag	Description
ms	Specifies the amount of memory to allocate to the Fulfillment Server on startup. Use it as <code>-Dms##[k m]</code> where <code>##</code> is the amount of memory, <code>k</code> is kilobytes and <code>m</code> is megabytes. Specifying a larger value for <code>ms</code> at startup can produce some performance gains depending on how memory intensive the server is for the types of requests it is processing. The default value is 1 MB.
mx	Specifies the maximum amount of memory to allocate to the Fulfillment Server. The server starts with the value for 'ms' and works up to the value of <code>mx</code> as needed. Use it as <code>-Dmx##[k m]</code> where <code>##</code> is the amount of memory, <code>k</code> is kilobytes, and <code>m</code> is megabytes. Use this flag if the server is encountering an <code>OutOfMemoryError</code> . The default value is 16 MB.

Language Character Parsing

The Fulfillment Server sometimes incorrectly parses certain natural language characters in email when using default environment settings and Java runtime options. The following configuration settings in the file `FM_start.sh` work for some of the languages that have encoding issues with the default settings.

- Chinese


```
export LANG=zh_TW.big5
-Dengine.CharEncoding=BIG5
```
- French


```
LANG=fr_FR.iso88591
-Dengine.CharEncoding=8859_1
```
- Korean (Hangul)


```
-Dfile.encoding=EUC-KR
```
- Japanese


```
export LANG=ja_JP.PCK
-Dengine.CharEncoding=SHIFT_JIS
```
- Russian (Cyrillic)


```
export LANG=ru_RU.KOI8R
or
export LANG=ru_RU.iso88591
```


This appendix covers the following topics:

- Introduction
- Priority Levels
- Interaction History Logging
- Extended Headers
- Extended Header XML Format
- Start_Request
- Get_Content_Xml
- One-to-One Fulfillment Request Types
- Example <item> XML Element
- Messages and Notifications
- Business Events
- Common Messages
- Get_Content_XML Messages
- Example Scripts

Introduction

Oracle One-to-One Fulfillment is an automated way to send information to customers. It provides the ability to immediately satisfy a customer's requests for information, literature, and other correspondence or collateral. Customer service representatives handle a variety of requests ranging from product and service inquiries, pricing questions, billing inquiries and general customer care issues. Many of these requests result in distribution of literature, collateral, forms of application, letters or correspondence to the customer.

Oracle One-to-One Fulfillment provides the ability for interaction center administrators, mobile field representatives, marketing managers, customer care representatives and other service agents to respond to different customer needs quickly and easily by using email, fax or print channels.

PROCEDURE NAME	DESCRIPTION
Start_Request	Generates the history request identifier (x_hist_id), used as the tracking identifier for the fulfillment request.
Get_Content_XML	Forms the <item> element of the XML request. You can call this API as many times as necessary.
Submit_Request	Initiates the fulfillment request process.

Priority Levels

The following table lists the priority levels defined as global constants in package JTF_FM_REQUEST_GRP. The default is G_PRIORITY_REGULAR. Unused priority numbers are reserved for future use.

LEVEL	NUMBER
G_PRIORITY_HIGHEST	1
G_PRIORITY_SPECIALITY_FLAG	6
G_PRIORITY_REGULAR	7

Interaction History Logging

One-to-One Fulfillment requests information can be logged in Interaction History Tables in the following ways:

- Interaction History logging for Single Requests. Valid party_id should be passed to log interaction history.
- For any other type of Requests except "TEST" requests, p_per_user_history should be set to 'Y' to log interaction history. There is no interaction history logging for "TEST" requests.

Extended Headers

Extended Header information can be passed to the One-to-One Fulfillment Server using the (p_)extended_header parameter of the API. This is an optional parameter. If passed, the API expects this parameter in the XML format described in Extended Header XML Format, page B-3. If any other format is used, the API will throw an error and the request will not be submitted.

Extended Header XML Format

```
<extended_header media_type="EMAIL">
<header_name>email_from_address</header_name>
<header_value>John.Doe@oracle.com</header_value>
</extended_header>
```

One-to-One Fulfillment Server recognizes the following sets of extended header parameters.

- media_type="EMAIL", "FAX" or "PRINTER"
- email_from_address
- email_reply_to_address
- email_cc_address
- email_bcc_address
- sender_display_name

If any parameter other than those listed above are passed, the Fulfillment Server will put it into the header information.

Currently the media_type defaults to "EMAIL". If media_type is not set, then the Fulfillment Server will assume it to be EMAIL.

Sample Extended Header XML

```
<extended_header media_type="EMAIL">
<header_name>email_from_address</header_name>
<header_value>John.Doe@oracle.com</header_value>
<header_name>email_cc_address</header_name>
<header_value>Jane.Doe@oracle.com</header_value>
<header_name>email_bcc_address</header_name>
<header_value>xyz@oracle.com</header_value>
<header_name>sender_display_name</header_name>
<header_value>Jason.Doe</header_value>
```

</extended_header>

Start_Request

Call the Start_Request first to generate the history request identifier (x_request_id) that is used as the tracking identifier for the fulfillment request.

Procedure Specification

```
PROCEDURE Start_Request
( p_api_version IN NUMBER,
  p_init_msg_list IN VARCHAR2 := FND_API.G_FALSE,
  p_commit IN VARCHAR2 := FND_API.G_FALSE,
  p_validation_level IN NUMBER := FND_API.G_VALID_LEVEL_FULL,
  x_return_status OUT VARCHAR2,
  x_msg_count OUT NUMBER,
  x_msg_data OUT VARCHAR2,
  x_request_id OUT NUMBER
);
```

Parameter Descriptions

The following table describes the IN parameters associated with the Start Request.

Parameter	Data Type	Required	Descriptions and Validations
p_api_version	NUMBER	Yes	This must match the version number of the API. An unexpected error is returned if the calling program version number is incompatible with the current API version number.
p_init_msg_list	VARCHAR2	Yes	Default = FND_API.G_FALSE. If set to true, then the API makes a call to fnd_msg_pub.initialize the message stack. If set to false, then the calling program must initialize the message stack. This action is required to be performed only once, even when more than one API is called.

Parameter	Data Type	Required	Descriptions and Validations
p_commit	VARCHAR2(1)	No	Default = FND_API.G_FALSE. If set to true, then the API commits before returning to the calling program. If set to false, then it is the responsibility of the calling program to commit the transaction.
p_validation_level	NUMBER	No	This value determines whether or not validation should occur. This variable is checked against FND_API.G_VALID_LEVEL_NONE, and if it is greater than the latter, validation occurs.

Parameter Descriptions

The following table lists the OUT parameters associated with Request.

Parameter	Data Type	Description
x_return_status	VARCHAR2	Indicates the return status of the API. The values returned are one of the following: FND_API.G_RET_STS_SUCCESS:Success: Indicates that the API call was successful FND_API.G_RET_STS_ERROR:Expected Error: There is a validation error, or missing data error. FND_API.G_RET_STS_UNEXP_ERROR:Unexpected Error: The calling program can not correct the error.
x_msg_count	NUMBER	Holds the number of messages in the message list.
x_msg_data	VARCHAR2	Holds the encoded message if x_msg_xount is equal to one.
x_request_id	NUMBER	The fulfillment request identifier. This can be used to track the status of the request.

Get_Content_Xml

Oracle does not recommend and does not support the creation of XML fulfillment

requests by any other means than through these published APIs. Oracle provides this information on how the XML request is formed solely to facilitate understanding of the fulfillment request process.

Use the `Get_Content_XML` API to generate one, or more, XML `<item>` elements for a fulfillment request. The content is formed based on the parameters passed to the API, and the API returns the XML `<item>` as parameter `x_content_xml`.

The `Get_Content_XML` API performs the following tasks:

- Verifies that the `p_content_id`, `p_media_type`, `p_content_type` and `p_request_id` parameters were passed to it. If not supplied, the API exits with an unexpected error condition.
- Forms the XML element using the parameters passed to it.
- Passes the XML element back to the calling application using the `x_content_xml` output parameter.

The `Get_Content_XML` API generates only one XML parameter for each API call. If you want to submit multiple content items for a single fulfillment request, then you must do the following:

- Append the `x_content_xml` content from the previous calls to the current one
- Submit the resulting concatenated `x_content_xml` string.

The `Submit_Request` API takes this parameter and merges it with the XML request that it forms, submitting the complete request to the Request queue.

Content Source Parameter

New Parameter added in 11.5.9 when One-to-One Fulfillment started supporting both as Marketing Encyclopedia Systems (MES) and Oracle Content Management (OCM).

The Default Value assumed for this parameter is "mes". If `GET_CONTENT_XML` API is called without this parameter, the repository is assumed to be Marketing Encyclopedia Systems (MES).

From Release 11.5.9 onward, this API supports Oracle Content Management. The other valid value is "ocm".

From 11.5.10 release onward, this API supports the renditions available in Oracle Content Management.

One-to-One Fulfillment Request Types

Use one of the following content types (`p_content_type`) to create a valid fulfillment request.

- QUERY

- COLLATERAL
- ATTACHMENT
- DATA

If any other values are passed to it, the Get_Content_XML API exits with an unexpected error.

The following table lists example values for the p_content_id parameters used with query and collateral fulfillment requests.

Sample Parameter Values for Query and Collateral Requests

- p_content_id
- 7008
- 7009
- 7111
- 7234
- 8101
- 9101

Example <item> XML Element

The Get_Content_XML API forms the XML in the fulfillment request.

In Release 11.5.10, Oracle One-to-One Fulfillment supports content from Oracle Content Manager and Management Encyclopedia System. To facilitate this support, the GET_CONTENT_XML has been overloaded. You can use the MES repository without changing any API calls. But, if the new content repository OCM is to be used, then you need to call the overloaded GET_CONTENT_XML with p_content_source = 'ocm' and p_version = version of the document.

The following code is an example of XML that is generated as part of an attachment fulfillment request, with an email media type. The content_id parameter is required for this API. The XML is passed back to the calling application as output parameter x_content_xml.

```
<items><item>
<media_type>
<email> abc@oracle.com</email>
</media_type>
<item_content quantity="0" user_note="THIS IS A FULFILLMENT TEST"
source="mes">
```

```

<files>
<file id="117423" body="yes"/>
</files>
</item_content>
</item>
</items>

```

Note: Oracle does not recommend, nor does it support, the creation of XML fulfillment requests by any other means than through the APIs published in this document. Oracle provides this information on how the XML request is formed *only* to facilitate understanding of the fulfillment request process.

Get_Content_Xml - Overloaded Procedure Specification

```

PROCEDURE Get_Content_XML
( p_api_version IN NUMBER,
  p_init_msg_list IN VARCHAR2 := FND_API.G_FALSE,
  p_commit IN VARCHAR2 := FND_API.G_FALSE,
  p_validation_level IN NUMBER := FND_API.G_VALID_LEVEL_FULL,
  x_return_status OUT VARCHAR2,
  x_msg_count OUT NUMBER,
  x_msg_data OUT VARCHAR2,
  p_content_id IN NUMBER,
  p_content_nm IN VARCHAR2 := FND_API.G_MISS_CHAR, -- deprecated
  p_document_type IN VARCHAR2 := FND_API.G_MISS_CHAR, --deprecated
  p_quantity IN NUMBER := 1,
  p_media_type IN VARCHAR2,
  p_printer IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_email IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_fax IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_file_path IN VARCHAR2 := FND_API.G_MISS_CHAR, --deprecated
  p_user_note IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_content_type IN VARCHAR2,
  p_bind_var IN G_VARCHAR_TBL_TYPE := L_VARCHAR_TBL,
  p_bind_val IN G_VARCHAR_TBL_TYPE := L_VARCHAR_TBL,
  p_bind_var_type IN G_VARCHAR_TBL_TYPE := L_VARCHAR_TBL,
  p_request_id IN NUMBER,
  x_content_xml OUT VARCHAR2 ,
  p_content_source IN VARCHAR2,

```

```

p_version IN NUMBER
);

```

Get_Content_Xml Specification

```

PROCEDURE Get_Content_XML
(
  p_api_version IN NUMBER,
  p_init_msg_list IN VARCHAR2 := FND_API.G_FALSE,
  p_commit IN VARCHAR2 := FND_API.G_FALSE,
  p_validation_level IN NUMBER := FND_API.G_VALID_LEVEL_FULL,
  x_return_status OUT VARCHAR2,
  x_msg_count OUT NUMBER,
  x_msg_data OUT VARCHAR2,
  p_content_id IN NUMBER,
  p_content_nm IN VARCHAR2 := FND_API.G_MISS_CHAR, -- deprecated
  p_document_type IN VARCHAR2 := FND_API.G_MISS_CHAR, --deprecated
  p_quantity IN NUMBER := 1,
  p_media_type IN VARCHAR2,
  p_printer IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_email IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_fax IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_file_path IN VARCHAR2 := FND_API.G_MISS_CHAR, --deprecated
  p_user_note IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_content_type IN VARCHAR2,
  p_bind_var IN G_VARCHAR_TBL_TYPE := L_VARCHAR_TBL,
  p_bind_val IN G_VARCHAR_TBL_TYPE := L_VARCHAR_TBL,
  p_bind_var_type IN G_VARCHAR_TBL_TYPE := L_VARCHAR_TBL,
  p_request_id IN NUMBER,
  x_content_xml OUT VARCHAR2)

```

Parameter Descriptions

The following table describes the Get_Content_XML IN parameters.

Get_Content_XML IN Parameters

Parameter	Data Type	Required	Descriptions and Validations
------------------	------------------	-----------------	-------------------------------------

Parameter	Data Type	Required	Descriptions and Validations
p_api_version	NUMBER	Yes	Must match the version number of the API. An unexpected error is returned if the calling program version number is incompatible with the current API version number (provided in the documentation).
p_init_msg_list	VARCHAR2	Yes	Default = FND_API.G_FALSE. If set to true, then the API makes a call to fnd_msg_pub.initialize the message stack. If set to false, then the calling program must initialize the message stack. This action is required to be performed only once, even in cases when more than one API is called.
p_commit	VARCHAR2(1)	No	Default = FND_API.G_FALSE. If set to true, then the API commits before returning to the calling program. If set to false, then it is the responsibility of the calling program to commit the transaction.
p_validation_level	NUMBER	No	This value determines whether or not validation should occur. This variable is checked against FND_API.G_VALID_LEVEL_NONE, and if it is greater than the latter, then validation occurs.
p_content_id	NUMBER	Yes	A unique numeric identifier that identifies the content that is to be sent out. If not provided in the current API call, then an error message is generated and the API exits.
p_content_nm	VARCHAR2	No	Deprecated, currently unused.
p_document_type	VARCHAR2	No	Deprecated, currently unused.
p_quantity	NUMBER	Yes	Quantity of the content to be sent (set always to 1).

Parameter	Data Type	Required	Descriptions and Validations
p_media_type	VARCHAR2	Yes	Media type of the content generated by the Get_Content_XML API: FAX EMAIL PRINTER. If not provided in the current API call, then an error message is generated and the API exits. The same content can also be sent through all three media types or a combination of two media types. The values can be comma separated, for example, 'FAX,EMAIL,PRINTER' or 'EMAIL,FAX' or 'EMAIL,PRINTER' or any similar combinations.
p_printer	VARCHAR2	No	Supplied printer address, if p_media_type is PRINTER. This parameter can accept the printer name/alias values.
p_email	VARCHAR2	No	Supplied email address, if p_media_type is EMAIL.
p_fax	VARCHAR2	No	Supplied fax address, if p_media_type is FAX
p_file_path	VARCHAR2	No	Deprecated, currently unused.
p_user_note	VARCHAR2	No	Any note that the agent wants to attach.
p_content_type	VARCHAR2	Yes	Valid content types: <ul style="list-style-type: none"> • QUERY • COLLATERAL • ATTACHMENT • DATA If not provided in the current API call, then an error message is generated and the API exits.
p_bind_var	G_VARCHAR _TBL_TYPE=L _VARCHAR_T BL	No	Used with QUERY and DATA fulfillment requests. Part of a set of three matching records.

Parameter	Data Type	Required	Descriptions and Validations
p_bind_val	G_VARCHAR _TBL_TYPE=L _VARCHAR_T BL	No	Used with QUERY and DATA fulfillment requests. Part of a set of three matching records.
p_bind_var_type	G_VARCHAR _TBL_TYPE=L _VARCHAR_T BL	No	Used with QUERY and DATA fulfillment requests. Part of a set of three matching records.
p_request_id	NUMBER	Yes	The Start_Request API generates this unique identifier for the fulfillment request. If not provided in the current API call, then an error message is generated and the API exits.
p_content_source	VARCHAR2	No	New parameter added in 11.5.9 release. Supports the content repositories MES and OCM. The default is assumed as MES. OCM will use the Rendition functionality in OCM 11.5.10.
p_version	NUMBER	No	OCM repositories support content versioning. Default is assumed as 1.

Get Content XML OUT Parameters

The following table describes the Get_Content_XML OUT parameters.

Parameter	Data Type	Description
-----------	-----------	-------------

Parameter	Data Type	Description
x_return_status	VARCHAR2	Indicates the return status of the API. The returned values returned are one of the following: <ul style="list-style-type: none"> FND_API.G_RET_STS_SUCCESS. Success: Indicates that the API call was successful. FND_API.G_RET_STS_ERROR. Expected Error: There is a validation error, or missing data error. FND_API.G_RET_STS_UNEXP_ERROR. Unexpected Error: The calling program can not correct the error.
x_msg_count	NUMBER	Holds the number of messages in the message list.
x_msg_data	VARCHAR2	Holds the encoded message if x_msg_count is equal to one.
x_content_xml	VARCHAR2	Output content XML.

Send Request & Submit Request

Send Request is submit request overloaded. Submit Request in turn makes the call to Send Request. The additional parameter is p_preview.

Parameter p_preview

The p_preview parameter provides the ability to preview a fulfillment request before sending it to its destination. You can preview, submit or delete a fulfillment request through the One-to-One Fulfillment Administrator Console.

Previewing allows for a visual inspection of a fulfillment request before you decide to send it or cancel it. When you preview the request in a browser, you can print it through the browser. Printing a previewed request through a browser does not remove the request from the status table.

The value of p_preview determines whether the request is to be dispatched immediately or stored in the database to be inspected visually.

- If p_preview is set to TRUE, then the request is marked as PREVIEW, and written to the Request queue, but it is not sent.
- If p_preview is set to FALSE, then the request is written to the Request queue, and it is sent in the usual fashion.

Note: The fulfillment request is written to the Request queue regardless of the value that is set for the parameter p_preview.

Use the Submit_Request procedure to submit a single new fulfillment request to the fulfillment engine. This API calls the Send_Request API.

Send Request Procedure Specification

```
PROCEDURE Submit_Request
( p_api_version IN NUMBER,
  p_init_msg_list IN VARCHAR2 := FND_API.G_FALSE,
  p_commit IN VARCHAR2 := FND_API.G_FALSE,
  p_validation_level IN NUMBER := FND_API.G_VALID_LEVEL_FULL,
  x_return_status OUT VARCHAR2,
  x_msg_count OUT NUMBER,
  x_msg_data OUT VARCHAR2,
  p_template_id IN NUMBER := FND_API.G_MISS_NUM,
  p_subject IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_party_id IN NUMBER := FND_API.G_MISS_NUM,
  p_party_name IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_user_id IN NUMBER,
  p_priority IN NUMBER := G_PRIORITY_REGULAR,
  p_source_code_id IN NUMBER := FND_API.G_MISS_NUM,
  p_source_code IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_object_type IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_object_id IN NUMBER := FND_API.G_MISS_NUM,
  p_order_id IN NUMBER := FND_API.G_MISS_NUM,
  p_doc_id IN NUMBER := FND_API.G_MISS_NUM,
  p_doc_ref IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_server_id IN NUMBER := FND_API.G_MISS_NUM,
  p_queue_response IN VARCHAR2 := FND_API.G_FALSE,
  p_extended_header IN VARCHAR2 := FND_API.G_MISS_CHAR,
  p_content_xml IN VARCHAR2,
  p_request_id IN NUMBER ,
  p_preview IN VARCHAR2
);
```

Submit Request Procedure Specification

```
PROCEDURE Submit_Request
( p_api_version IN NUMBER,
```

```

p_init_msg_list IN VARCHAR2 := FND_API.G_FALSE,
p_commit IN VARCHAR2 := FND_API.G_FALSE,
p_validation_level IN NUMBER := FND_API.G_VALID_LEVEL_FULL,
x_return_status OUT VARCHAR2,
x_msg_count OUT NUMBER,
x_msg_data OUT VARCHAR2,
p_template_id IN NUMBER := FND_API.G_MISS_NUM,
p_subject IN VARCHAR2 := FND_API.G_MISS_CHAR,
p_party_id IN NUMBER := FND_API.G_MISS_NUM,
p_party_name IN VARCHAR2 := FND_API.G_MISS_CHAR,
p_user_id IN NUMBER,
p_priority IN NUMBER := G_PRIORITY_REGULAR,
p_source_code_id IN NUMBER := FND_API.G_MISS_NUM,
p_source_code IN VARCHAR2 := FND_API.G_MISS_CHAR,
p_object_type IN VARCHAR2 := FND_API.G_MISS_CHAR,
p_object_id IN NUMBER := FND_API.G_MISS_NUM,
p_order_id IN NUMBER := FND_API.G_MISS_NUM,
p_doc_id IN NUMBER := FND_API.G_MISS_NUM,
p_doc_ref IN VARCHAR2 := FND_API.G_MISS_CHAR,
p_server_id IN NUMBER := FND_API.G_MISS_NUM,
p_queue_response IN VARCHAR2 := FND_API.G_FALSE,
p_extended_header IN VARCHAR2 := FND_API.G_MISS_CHAR,
p_content_xml IN VARCHAR2,
p_request_id IN NUMBER
);

```

Send and Submit_Request IN Parameters

The following table lists the IN parameters for Send_Request and Submit_Request.

Send and Submit_Request IN Parameters

Parameter	Data Type	Required	Descriptions and Validations
p_api_version	NUMBER	Yes	This must match the version number of the API. An unexpected error is returned if the calling program version number is incompatible with the current API version number (provided in the documentation)

Parameter	Data Type	Required	Descriptions and Validations
p_init_msg_list	VARCHAR2	Yes	Default = FND_API.G_FALSE. If set to true, then the API makes a call to fnd_msg_pub.initialize the message stack. If set to false, then the calling program must initialize the message stack. This action is required to be performed only once, even in the case where more than one API is called
p_commit	VARCHAR2(1)	No	Default = FND_API.G_FALSE. If set to true, then the API commits before returning to the calling program. If set to false, then it is the calling program's responsibility to commit the transaction.
p_validation_level	NUMBER	No	This value determines whether validation should occur or not. This variable is checked against FND_API.G_VALID_LEVEL_NONE and if it is greater than the latter, validation occurs.
p_template_id	NUMBER	No	The fulfillment request template identifier. NULL if request does not correspond to a predefined template.
p_subject	VARCHAR2	No	Subject specifies the purpose of the request
p_party_id	NUMBER	No	Customer Identifier.
p_party_name	VARCHAR2	No	Customer Name.
p_user_id	NUMBER	Yes	Agent or user Identifier who is making the request.
p_priority	NUMBER	No	Priority for that particular request. If none provided, default is regular
p_source_code_id	NUMBER	No	Campaign or promotion field.
p_source_code	VARCHAR2	No	Campaign or promotion field.

Parameter	Data Type	Required	Descriptions and Validations
p_object_type	VARCHAR2	No	Campaign or promotion field.
p_object_id	NUMBER	No	Campaign or promotion field.
p_order_id	NUMBER	No	Unique identifier of the field.
p_doc_id	NUMBER	No	Document Identifier, if not provided, set to p_request_id
p_doc_ref	VARCHAR2	No	Document Reference. If not provided, set to JFUF.
p_server_id	NUMBER	No	Unique Identifier of the One-to-One Fulfillment Server. It is the primary key of JTF_FM_SERVICE_ALL table. If this is provided, that particular server processes the request.
p_queue_response	VARCHAR2	No	Deprecated and not supported. This field is used internally for other purposes.
p_extended_header	VARCHAR2	No	Used to insert a character string in the form of one or more extended header elements into the XML request.
p_content_xml	VARCHAR2	Yes	The XML content formed by calling the GET_CONTENT_XML API.
p_request_id	NUMBER	Yes	The unique identifier for the fulfillment request generated by the Start_Request API. If not provided in the current API call, an error message is generated and the API exits.
p_preview	VARCHAR2	Yes	Boolean Value: Used to specify whether or not this is a preview request: TRUE: Preview this document. FALSE: Do not preview this document, instead send it to the fulfillment request queue.

Submit Request OUT Parameters

The following table describes the Submit_Request OUT parameters.

Parameter	Data Type	Description
x_return_status	VARCHAR2	Indicates the return status of the API. Returns one of the following values: <ul style="list-style-type: none">FND_API.G_RET_STS_SUCCESS. Success: Indicates that the API call was successfulFND_API.G_RET_STS_ERROR. Expected Error: There is a validation error, or missing data error.FND_API.G_RET_STS_UNEXP_ERROR. Unexpected Error: The calling program cannot correct the error.
x_msg_count	NUMBER	Holds the number of messages in the message list.
x_msg_data	VARCHAR2	Holds the encoded message if x_msg_count is equal to one.

Messages and Notifications

The APIs contained in package JTF_FM_REQUEST_GRP generate messages and notifications as necessary. In addition, some messages are common to all of the Fulfillment APIs. These messages are detailed in the following sections.

The messages and notifications generated by each API are described in the following sections.

- "Common Messages", page B-19
- "Get_Content_XML Messages", page B-19

Note: Not all status notifications are required to provide a number identifier. However, in many cases a number identifier is provided.

Business Events

When using the Multithreaded Fulfillment option, the business event `oracle.apps.jtf.fm.int.RequestPostProcessing` is raised at the end of the pre-processing stage of a request. By subscribing to this event, implementers have the ability to add

custom validation for email request processing.

Common Messages

The following table lists the messages and notifications that are common to many of the APIs contained in package JTF_FM_REQUEST_GRP.

Common Status Messages

NUMBER	TYPE	NAME	TEXT
210925	E	JTF_FM_API_QUEUE_NOT_FOUND	Could not find queue names for Server ID &ARG1
210927	E	JTF_FM_API_REQUEST_NOTFOUND	Could not find fulfillment request for Request ID & ARG1 in database
210932	E	JTF_FM_API_MISSING_REQUEST_ID	Value for Request ID cannot be NULL.
210939	E	JTF_FM_API_DEBUG_MESSAGE	1 to 1 Fulfillment API: &ARG1
210940	S	JTF_FM_API_SUCCESS_MESSAGE	The &ARG1 API has completed successfully.

Get_Content_XML Messages

The following table lists the messages and notifications that are used by the Get_Content_XML API.

Status Messages for Get_Content_XML

NUMBER	TYPE	NAME	TEXT
210922	E	JTF_FM_API_INVALID_MEDIATYPE	Invalid media type &ARG1 passed for Content ID &ARG2. Allowed media types are EMAIL, FAX, PRINTER

NUMBER	TYPE	NAME	TEXT
210923	E	JTF_FM_API_MULTIPLE_MEDIATYPE	Multiple media types were passed for Content ID = &ARG1. Only one media type is allowed for each content.
210924	E	JTF_FM_API_INVALID_CONTENT	Invalid content type &ARG1 passed for Content ID &ARG2. Allowed content types are QUERY, COLLATERAL, ATTACHMENT or DATA.
210929	E	JTF_FM_API_MISSING_CONTENT_ID	Value for Content ID cannot be NULL.
210930	E	JTF_FM_API_MISSING_MEDIA_TYPE	Value for Media Type cannot be NULL.
210931	E	JTF_FM_API_MISSING_CONTENTTYPE	Value for Content Type cannot be NULL.
210974	E	JTF_FM_DISP_CONTENT_NOT_FOUND	Cannot find document for Content ID &ARG1.
	E	JTF_FM_API_CONTENT_INVALID	The content_id &ARG1 is not a valid content_id, or does not exist.
	E	JTF_FM_API_INVALID_LISTTYPE	Invalid list type &ARG1 passed. Allowed list types are VIEW or ADDRESS
	E	JTF_FM_API_MISSING_LIST_TYPE	Value for List cannot be NULL
	E	JTF_FM_API_INVALID_REQTYPE	Invalid request type specified. Only allowed values are "E,P or T". E for "Electronic Request", P for "Physical Collateral" and T for "Test Requests".

NUMBER	TYPE	NAME	TEXT
E		JTF_FM_API_CF_INVALID_MEDIA	Invalid Media Type specified. Only allowed Media types are any one of the following. "EPF, EFP, FEP, FPE, PEF, PFE, EP, EF, E, PE, PF, P, FE, FP, F". E represents "EMAIL", P represents "PRINTER", and F represents "FAX"

Example Scripts

The following scripts are examples that may not work if used in the exact form shown here and placed into another environment.

Sample Script - Start_Request, Get_Content_Xml, Submit_Request

```

DECLARE
l_api_versionNUMBER := 1.0;
l_commitVARCHAR2(5) := FND_API.G_TRUE;
l_content_idVARCHAR2(30);
l_media_typeVARCHAR2(30);
l_request_typeVARCHAR2(20);
l_user_noteVARCHAR2(1000);
l_document_typeVARCHAR2(150);
l_bind_var      JTF_FM_REQUEST_GRP.G_VARCHAR_TBL_TYPE ;
l_bind_var_type JTF_FM_REQUEST_GRP.G_VARCHAR_TBL_TYPE ;
l_bind_val      JTF_FM_REQUEST_GRP.G_VARCHAR_TBL_TYPE ;
l_template_id  NUMBER;
l_party_id NUMBER;
l_user_idNUMBER;
l_server_id NUMBER;
l_request_idNUMBER;
l_subjectVARCHAR2(100);
l_msg_count  NUMBER;
l_msg_data  VARCHAR2(1000);
l_return_status VARCHAR2(1000);
l_content_xml1 VARCHAR2(1000);
l_content_xml VARCHAR2(10000);

```

```

l_content_nm VARCHAR2(100);
l_email VARCHAR2(100);
l_printer VARCHAR2(100);
l_file_path VARCHAR2(100);
l_fax VARCHAR2(100);

BEGIN

-- Start the fulfillment request. The output request_id must be passed
-- to all subsequent calls made for this request.
JTF_FM_REQUEST_GRP.Start_Request
(
p_api_version => l_api_version,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data,
x_request_id => l_request_id
);

DBMS_OUTPUT.PUT_LINE('Request_ID: '||to_char(l_request_id));

-- Initialize the parameters for Master Document.
-- this does not need anything.it just sends the email .
l_content_id := &p_contentid;--item id
l_media_type := &p_mediaType;
l_document_type := &docType;--zip or htm or html
l_request_type := &p_requestType;
l_user_note := 'Please dont hesitate to call at 1-800-INFODSK.';
l_email := &p_email;
l_printer := '\\ORADEV8\DEVSTANLEY';
l_file_path := NULL;
l_fax := '9999999999';

-- This call gets the XML string for the content(Master Document) with
-- the parameters as defined above
JTF_FM_REQUEST_GRP.Get_Content_XML
(
p_api_version => l_api_version,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data,
p_content_id => l_content_id,
p_content_nm => l_content_nm,
p_document_type => l_document_type,

```

```

p_media_type=> l_media_type,
p_printer => l_printer,
p_email => l_email,
p_file_path => l_file_path,
p_fax => l_fax,
p_user_note => l_user_note, p_content_type => l_request_type,
p_bind_var => l_bind_var,
p_bind_val => l_bind_val,
p_bind_var_type => l_bind_var_type,
p_request_id => l_request_id,
x_content_xml => l_content_xml1);
-- The output XML string is stored in a local variable
l_content_xml := l_content_xml1;
-- Check the return status
DBMS_OUTPUT.PUT_LINE('Return Status: '||l_return_status);
DBMS_OUTPUT.PUT_LINE('Message_Count: '||l_msg_count);
DBMS_OUTPUT.PUT_LINE('Message Data: '||l_msg_data);
-- Initialize Parameters for submitting the fulfillment request
l_user_id := &p_userId;--this is a parameter that can be filled during
run
-- l_server_id := null;
l_subject := &p_subject;--subject could be environment and what test it
is
-- Submit the fulfillment request
JTF_FM_REQUEST_GRP.Submit_Request
(
p_api_version => l_api_version,
p_commit => l_commit,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data,
p_subject => l_subject,
p_party_id => l_party_id,
p_user_id => l_user_id,
p_queue_response => FND_API.G_TRUE,
p_content_xml => l_content_xml,
p_request_id => l_request_id
);
DBMS_OUTPUT.PUT_LINE('Return Status: '||l_return_status);
DBMS_OUTPUT.PUT_LINE('Message_Count: '||l_msg_count);

```

```
DBMS_OUTPUT.PUT_LINE('Message Data: '||l_msg_data);  
DBMS_OUTPUT.PUT_LINE('Message Data: '||l_user_id);  
END;
```

Profile Options

Fulfillment Profile Options

The following table lists the Oracle One-to-One Fulfillment profile options.

Oracle One-to-One Fulfillment Profile Options

Name	Category	Seeded Description	Description	Example
JTO: Debug: Enable UI Debug	Debug	When true, write debug output directly to the generated JSP page.	Print debug statements on JSP pages.	<blank>
JTO: Email: Tracking URL	Setup	URL to embed in email messages used to register when a recipient opens a received email. See the Implementation Guide for this value.	-	<pre><img src="http://< host>:<port>/ OA_HTML/jtffm ImageTracker. jsp?PAGE.OBJ. requestId=&PA GE.OBJECT.obj Id=&file_name =jtfultrpx.gif " height="0" width="0" alt=""></pre>

JTO: Email: Unsubscribe email footer for TEXT emails	Setup	This profile stores the text and the URL that will be added to the end of text format email body; this portion points the user to the Unsubscribe page in the application.	-	-
JTO: Email: Unsubscribe URL	Setup	The unsubscribe text at the end of the email message and the URL link to allow recipients to add their email addresses to a Do Not Contact list.	-	"To unsubscribe, please click here
JTO: Integration: OSO Create Page JSP	Integration	The JSP filename of the page invoked from the Fulfillment Status page Create button.	Destination of the Create page button.	<blank>
JTO: Integration: OSO Done Page JSP	Integration	The JSP filename of the page to return to after submitting a request.	Destination of the Create page button.	jtffmast.jsp
JTO: Interaction History: Result ID Failure	Interaction History	Interaction History result ID assigned to failed requests.	Set the Interaction History result ID of failed dispatches of the Fulfillment Server.	9
JTO: Interaction History: Outcome ID	Interaction History	Interaction History outcome ID assigned to requests.	Set the Outcome code of Interactions logged by the Fulfillment Server.	10

JTO: Interaction History: Result ID Success	Interaction History	Interaction History result ID assigned to successful requests.	Set the Interaction History result id of successful dispatches of the Fulfillment Server.	8
JTO: Merge Delimiter: Left	Setup	Merge field left delimiter.	Designate the left delimiter of merge fields.	<< or {{
JTO: Merge Delimiter: Left Encrypted	Setup	Used to designate the left delimiter of encrypted merge fields.	Designate the left delimiter of encrypted merge fields.	<< or {{
JTO: Merge Delimiter: Right	Setup	Merge field right delimiter.	Designate the right delimiter of merge fields.	>> or }}
JTO: Merge Delimiter: Right: Encrypted	Setup	Designate the right delimiter of encrypted merge fields.	Designate the right delimiter of encrypted merge fields.	>> or }}
JTO: Security: IMAP Password Encryption Key	Setup	The value to encrypt the bounceback account password.	Key used to encrypt and decrypt bounceback password.	changeme
JTO: Security: Merge Field Encryption Key	Setup	The value to encrypt and decrypt merge fields.	Key used to encrypt and decrypt encrypted merge fields.	changeme

JTO: Server: Batch Size	Setup	Used in the multithreaded design for the server, this profile specifies the number of request lines that are picked up by the server in one database call during the request fulfillment process. Server needs to be restarted for the change in this profile to take effect.	-	-
JTO: Server: Default Server ID	Setup	The default server to use when the user is not associated with any server.	Used by the APIs. The default server is assumed as the Rapid Install server (serverId - 5000). Groups can set up their own default server in this profile.	5000
JTO: Server: Error Log Filename	Setup	The server error log filename.	Default Fulfillment Server Error Log name.	Errors.log
JTO: Server: Event Log Filename	Setup	Events log file name.	Default Fulfillment Server Event Log name.	Events.log
JTO: Server: Log Level	Debug	Level of logging.	Level of logging used by the Fulfillment Server.	2

JTO: Server: Preload Contact Preferences	Setup	Used in the multithreaded design for the server, this profile specifies if the server should check the TCA contact preference for the TCA party in the request lines.	-	-
JTO: Server: Record success for Mass request at Batch level only	Setup	Used in the multithreaded design for the server, this profile specifies if the server should make a database update for every request line. Such updates slow down the request completion time. Please note that with this profile set to "Y", the database update is only done for failed lines. This means that if for some reason the fulfillment server Java process is killed the server will have no recollection of the emails sent successfully. When the server is brought up subsequently there is a chance that the server will send some emails again.	-	-
JTO: Server: Refresh Idle Milliseconds	Setup	The time, in milliseconds, during which the Fulfillment Server rests between self-diagnostic inspections.	Time, in milliseconds, during which the Fulfillment Server rests between self-diagnostic inspections.	20000

JTO: Server: Temp Directory	Setup	Fulfillment temp directory.	Default location for creating Fulfillment Server temporary files.	tmp
JTO: Server: Use Multithreaded Fulfillment	Setup	Specifies if the fulfillment server should use the multithreaded design for mass email requests. This profile option does not impact the processing for single & test requests. Also this design is not used for requests of type other than the email media type. Server needs to be restarted for the change in this profile to take effect.	-	-

Glossary

Advanced Queue

A mechanism within the Oracle database that is used by the Fulfillment APIs and Fulfillment Server to queue requests. Fulfillment APIs put requests into the queue. The Fulfillment Server pulls requests out of the queue.

bounceback

The process of returning email to the sending server because of an incorrect address (hard bounceback) or because the recipient's inbox is full (soft bounceback).

content item (cover letter)

An instance of a content type.

content type

The definition of the structure of a cover letter that can be used in email, fax and print.

database connectivity file

A text file that contains the connection parameters for the Oracle Applications database. It is created automatically when Oracle Applications is installed.

data source file

A text file that contains tab-separated merge fields that have been obtained by a query.

dequeuing

A process by which the Fulfillment Server retrieves fulfillment requests from a queue.

enqueueing

A process by which a business application places fulfillment requests into a queue.

email channel

Requests sent by email.

fax channel

Requests sent by fax.

Fulfillment Administration Console

The HTML interface for administrators to define and maintain the servers and the associated email, fax and print servers. Administrator can define groups of agents and associate them with the defined servers. Oracle One-to-One Fulfillment supports two content repositories, Marketing Encyclopedia System (MES) and Oracle Content Manager (OCM), and provides administration HTML pages to create and upload content into each repository.

fulfillment

The process of compiling and distributing information to customers.

fulfillment request

The customer request that is processed. The request can be made up of a simple email, merged email, email with attachments, and so on. A request starts when the Fulfillment APIs receive data. The request ends when the Fulfillment Server passes information to the Java mail APIs or to the Print or Fax Enabler.

Fulfillment Server

The multi-threaded Java server that processes Fulfillment requests by de-queuing them from an Advanced Queue.

glue code

Code that connects two components.

Java

A technology that is both a programming language and a platform.

Java API

The Java application programming interface. A large collection of ready-made software components that provides many capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces known as packages.

javac

The Java programming language compiler.

Java language

An object programming language in which programs are both compiled and interpreted. Using the compiler, a program is translated into an intermediate language

called Java bytecodes, which are platform-independent codes that are interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java bytecode instruction on the computer. Compilation happens only once, but interpretation occurs each time the program runs.

Java platform

A software-only environment that runs on top of other hardware-based platforms. The Java platform has two components: Java Virtual Machine (Java VM) and the Java Application Programming Interface (Java API).

Java Virtual Machine

The base for the Java platform. It is ported onto various hardware-based platforms.

malformed email

Email addresses that are incorrectly formed (such as a missing "@"). The resulting email cannot be sent to the customer.

mass request type

Large volume One-to-One requests. For mass requests, the calling application passes a SQL query which when executed derives the list of email addresses and fax numbers.

master document

An HTML file that contains the main content in a fulfillment item.

Oracle Marketing Encyclopedia System

A file repository that stores master documents and queries that are used to specify fulfillment content.

multi channel

Requests sent by more than one channel.

OCM

See Oracle Content Manager.

(One-to-One) request

Requests initiated by a calling application (such as Oracle Collections) to send one or more master documents or pieces of collateral to a customer.

(One-to-One) job

Requests that are comprised of one or more jobs (sub-requests) which represent the individual email or faxes, and so forth, that are sent to a specific customer.

opened email

Emails sent by Oracle One-to-One Fulfillment that have been opened by the recipient.

Oracle Content Manager

A document repository for fulfillment content.

physical request

One-to-One physical collateral requests for Collateral kits. These Collateral kits are separated into individual collateral and an order created which has multiple order lines (one for each collateral) and multiple shipment lines if the request has multiple recipients.

print channel

Requests sent to a printer and then mailed to a customer.

query

A SQL (structured query language) statement that generates selected data from a database. A query determines which records meet the defined criteria, and which fields of those records are returned.

queue

A repository for messages. Queues are stored in queue tables. Each queue table is a database table and contains one or more queues.

sent email

Emails sent by Oracle One-to-One Fulfillment.

single request type

One-to-One requests sent to a single customer.

template

A grouping of material that is useful but not required for distributing master documents and collateral. An Oracle Marketing Encyclopedia System template is a package of content that is compiled by the Fulfillment Server. The fulfillment request identifies the template being used to compile the content.

unopened email

Emails sent by Oracle One-to-One Fulfillment that have not yet been opened by the recipient.

Unsubscribe option

An option for email sent by Oracle One-to-One Fulfillment that contain an

"Unsubscribe" link, and in which the recipient (customer) has selected to not be contacted, either by email or not at all.

versioning

Updating a content item by modifying an existing version or creating a new one. Each version can have the status of In Progress, Submitted or Approved. In Oracle Content Manager, multiple approved versions can exist, but only one version can be set as the live version.

Index

Symbols

-D flags, 5-10

A

account statements, 1-1
Administration Console, 6-71
Adobe Acrobat, 6-68
Advanced Queuing, 1-7, 1-7
agent groups, 1-9
aliases, 1-13
amsContactPreferencesMain.jsp, 1-16
angle quotes, 5-14
API
 call for faxes, 4-2
 function of in fulfillment requests, 1-1
 in fulfillment requests, 1-7
 interface for Java, 4-2
 Java Mail, 1-5
 procedures, 1-7
 static, 4-4
approved version, 6-34
APPS.HZ_CONTACT_PREFERENCES, 5-11
apps.zip
 and log directory path, 5-9
associating printers with a printer server, 6-10

B

background mode, 5-12
batch email, 1-16
batch request, 1-9
bind variable, 1-13

blank screen on monitor, 5-3
bounceback, 1-7, 3-17

C

canceling requests, 6-43, 6-44
Captaris RightFax, 1-9, 2-2, 4-1
collateral, 6-28
 Oracle Marketing Online items, 6-15
 sales and marketing, 1-2
 upload, 1-14
commands in background mode, 5-12
conditional dependencies, 2-7
constructor, 4-2, 4-4
contact management, 2-6
Contact Preferences, 1-15
content, types of, 1-7
Content Renditions region, 6-41
content types, 5-9
Content Type Summary Page, 6-34
cover letter, 6-32
 attachments, 6-35
 body, 6-34
Cover Letter Content
 create, 6-39
cover letter content type
 Content Block, 6-34
 Footer, 6-34
 Header, 6-34
Cover Letter Content Types, 6-33
 Creating and Managing, 6-36
cover letters, 1-17
 and merge field errors, 5-14

- in master documents, 1-10
- in Oracle TeleSales, 2-6
- Create Query page, 6-31
- creating a print server, 6-9
- creating a user account, 3-6, 3-11

D

- database
 - column names in queries, 1-13
 - in Advanced Queuing, 1-7
 - polling, 1-8
 - table, 1-7
- data object, 1-7
- data security, 1-11
- data source file, 1-14
- Datasource sub tab, 6-31
- deactivating an assigned responsibility, 3-7, 3-12
- debug
 - server, 5-23
- debug flags, 5-5
- debug logging, 5-22
- define
 - attachment, 6-38
 - content block, 6-38
 - Footer attributes, 6-38
 - General region, 6-37
 - Header attributes, 6-37
- deleting history entries, 6-53
- delimiter flags, 5-14
- deliverables, 6-28
- dependencies
 - conditional, 2-7
 - mandatory, 2-2
- dequeuing, 1-7
- directory
 - for error messages, 5-9
- Documents Using Query, 6-31
- DOS
 - batch file, 3-22
 - commands, 5-13
 - startup script, 3-22
- double angle quotation mark, 6-57
- downloading master documents, 6-26
- duplicate
 - cover letter content type, 6-33
- dynamic content, 1-11

- dynamic master document, 1-11

E

- email, 6-32
 - and Oracle Collections, 2-5
 - and Oracle Lease Management, 2-5
 - and Oracle Marketing, 2-5
 - and Oracle Scripting, 2-6
 - and Oracle Student Systems, 2-6
 - and Oracle TeleSales, 2-7
 - and Oracle TeleService, 2-7
 - and queries, 1-12
 - as output channel, 1-7
 - as output device, 1-1, 1-9, 2-2
 - contact preferences, 2-7
 - fulfillment requests, 1-9
 - language characters, 5-8
 - protocols, 1-7
 - servers, 1-9
 - tracking, 1-9, 1-16
 - Unsubscribe option, 1-15
- Email Servers page, 3-15
- engine.CommandPort, 3-21
- enqueueing, 1-7
- error message directory, 5-9
- errors
 - common, 5-2
 - email, 5-7
 - Fulfillment Server, 5-3

F

- FAQ, 5-8
- fax, 6-32
 - API call, 4-2
 - as output channel, 1-7
 - as output device, 1-1, 1-9, 2-2
 - requests, 5-24
- fax integration, 4-1
- Fax Servers page, 3-18
- firewall, 1-10
- flags
 - D, 5-10
 - debug, 5-5
 - delimiter, 5-14
 - effect of changes to, 5-7
 - in startup script, 5-8

FM Server Script, 5-10
FNDPSTAX, 6-71
form documents, 1-17, 6-58
frequently asked questions, 5-8
fulfilled, 6-32
fulfillment
 definition, 1-1
 requests, 1-7
Fulfillment Server
 and errors, 5-24
 and print fulfillment, 4-4
 configuration for Unsubscribe, 1-15
 email errors, 5-7
 function of, 1-8
 history of requests, 1-14
 in fulfillment requests, 1-7
 parameters, A-1
 requirements, 2-1
 startup scripts, 1-9
 stopping and restarting, 5-7
Fulfillment Server error messages, 5-3

G

global variable, 1-13
grouping documents, 1-8
guillemet, 6-57, 6-58

H

hard bounce back, 1-7
Hexadecimal, 6-57
hidden image, 1-10
high quality print, 1-17
High-Quality Printing, 6-9
history entry, 6-53
History sub tab, 1-14, 6-45
HTML
 code in master documents, 1-11
 document, 1-8, 1-8
HTTP, 1-7

I

IDAP, 1-7
image tag, 1-10
initializing processor error, 5-13
integrating fax software, 4-1

Interaction History, 3-15

J

Java
 API Interface, 4-2
 exception, 5-9
 fax call, 4-2
 Mail APIs, 1-5
 print solution, 4-4
 property flags, 6-57, 6-58
 server, 1-8
java.lang.IllegalArgumentException, 5-10
java.lang.NullPointerException, 5-3
Java Server Page, 1-10
Java Virtual Machine
 resolving character errors in email, 5-8
JTF_FM_ADMIN, 3-7
JTF_FM_MAX_PROCESSES, 5-2
JTF_SYSTEM_ADMIN_ROLE, 3-8
JTFFMCTL.SH, 5-2

L

language characters, 5-8
lease contracts, 1-10
lease terms, 1-1
Linux, 1-17
log files
 directory path, 5-9

M

mail merge, 6-35
 personalized tags, 6-35
mandatory dependencies, 2-2
master document, 1-7
 and queries, 6-27
 creating and saving, 1-11
 downloading, 6-26
 in queries, 1-14
 types of, 1-10
 updating, 6-25
 uploading, 6-24
 viewing, 6-26
 viewing list of queries, 6-30
merge delimiters, 5-8
merge field, 6-34, 6-40

- merge fields
 - and data source file, 1-13
 - and master documents, 1-11
 - downloading list of, 6-31
 - errors, 5-14
 - in Adobe Acrobat PDF form document, 6-68
 - in RTF documents, 6-58
 - table rows, 5-15
- Microsoft Notepad, 1-11, 3-21
- Microsoft Word, 1-11, 6-59
- Microsoft WordPad, 1-11
- monitor, remote, 3-22
- monitor displays blank screen, 5-3
- multi-threaded, 1-8

N

- native print function, 6-58
- natural language, 5-8
- Network Name, 6-71
- new content items folder, 1-12
- notification letters, 1-1

O

- Octal, 6-57
- One-to-One Fulfillment Administrator responsibility, 6-39
- oracle.apps.jtf.fm.engine.FaxEnabler, 4-2
- oracle.apps.jtf.fm.engine.logging.logger, 5-9
- oracle.apps.jtf.fm.engine.PrintEnabler, 4-4
- Oracle Collections, 2-5
- Oracle Content Manager
 - profile options, 1-11
- Oracle CRM Connector, 4-2
- Oracle Interaction History, 1-8
- Oracle Lease Management, 2-5
- Oracle Marketing, 2-5
- Oracle Marketing Encyclopedia System
 - and updating documents, 6-25
 - assigning query to, 6-27
 - creating queries for, 6-29
 - fulfillment requests, 1-7
 - queries stored in, 6-28
 - uploading master document to, 6-24
- Oracle Marketing Online, 1-15, 1-16, 6-28
- Oracle Marketing Online items, 6-28
- Oracle One-to-One Fulfillment, 6-35

- Oracle Pasta, 1-17
- Oracle Scripting, 1-14, 2-6
- Oracle solution, 1-17
- Oracle Student Systems, 2-6
- Oracle Telesales, 5-9
- Oracle TeleSales, 1-14, 2-6
- Oracle TeleService, 2-7
- outbound content, 6-32
- outbound correspondence, 1-1
- out of memory error, 5-2
- output channel, 1-7
- output devices, 1-1, 1-9, 2-2
- Overview
 - Cover Letter, 6-32

P

- parameters
 - effect of changes to, 5-7
 - Fulfillment Server, A-1
- party_id, 1-16
- Pasta printer, 6-71
- Pasta Print Server, 6-71
- past due notices, 1-10
- PDF, 1-17, 1-18
 - collateral, 1-3
 - merge fields in, 6-68
- physical collateral, 1-1
- PL/SQL, 5-22
- port, 1-9, 3-21
- print
 - as output channel, 1-7
 - fulfillment requests, 4-4
 - request, 5-24
- print channel, 6-32
- printers, associating with print server, 6-10
- print renditions, 6-41
- print server, creating, 6-9
- print servers
 - as output device, 1-1, 1-9, 2-2
- product information, 1-1
- profile options, 1-11, 2-10

Q

- queries
 - and master document, 6-27
 - creating for Oracle Marketing Encyclopedia

- System, 6-29
 - functions of and syntax for, 1-11, 1-12
 - list of, 6-30
 - updating, 6-30
- Queries, 6-35
- query, 6-38
- Query sub tab, 6-28
- question marks in email, 5-8
- queues and queuing, 1-7
- Quick Find, 6-55

R

- remote command script, 1-9, 3-21
- Remote Monitor, 3-22
- request
 - canceling, 6-43, 6-44
 - outcomes, 1-14
 - print, 4-4
 - processing, 5-13
 - queue, 1-7
 - resubmitting, 6-53
 - search for, 6-54
 - stalled, 1-15
 - status of, 6-42
 - viewing details of, 6-45
- request_id, 1-16
- Request Failure Result, 3-15
- Request Outcome, 3-15
- Request Success Result, 3-15
- responsibilities, 2-9, 3-7
 - deactivating, 3-7, 3-12
- restarting stalled requests, 1-15
- resubmitting requests, 6-53
- rfwebcon.dll, 4-2
- Rich Text Format, 1-17, 6-58
- RightFax, 1-18
 - integration, 5-24
- RightFax Integration Module, 4-1
- RTF, 1-17, 1-18
- RTF document upload failure, 5-9
- RTF template, 6-71

S

- sales and marketing collateral, 1-2
- sales offers, 1-10
- screen on monitor goes blank, 5-3

- script
 - fulfillment server, 1-9
 - startup, 3-21, 4-2, 4-4
- search for requests, 6-54
- Searching Cover Letters, 6-41
- semicolon, in SQL statements, 1-13
- server
 - debugging, 5-23
- server, fulfillment, 2-1
- server polling, 1-8
- server process
 - maximum, 5-2
- service request notifications, 1-10
- shutdown time, 6-4
- SMTP, 1-9
- soft bounce back, 1-7
- SQL, 1-8, 1-12, 1-13, 6-35
- SQL query, 6-59, 6-69
- stalled requests, 1-15
- start time, 6-4
- startup script, 5-10
 - and fax solution, 4-2
 - flags, 5-8
 - monitoring remote commands, 3-21
 - print solution, 4-4
- static document, 1-8
- static master document, 1-11
- statistics, 1-9, 1-16
- status of request, 6-42
- Status page, 1-9
- Status sub tab, 1-15
- stopping and restarting the Fulfillment Server, 5-7
- structured query language, 6-35
- stylesheet, 6-33
- Submitting for Approvals, 6-41
- surveys, 1-1
- syntax, SQL query, 1-13
- system requirements, 2-1

T

- table rows, 5-15
- tables, queue, 1-7
- template
 - definition of, 1-10
- tracking email, 1-9, 3-16

U

Unicode, 6-57

UNIX, 1-17

- remote command script, 5-12

- shell script, 3-22

- startup script, 3-21

Unsubscribe, 3-16

Unsubscribe option, 1-15

updating master documents, 6-25

uploading RTF document, 5-9

US ASCII, 5-8

user account, 3-5, 3-6, 3-7, 3-11

V

variables, 1-13

viewing master documents, 6-26

W

Windows, 1-17

X

XML, 1-7

- interface for RightFax, 4-2

XML Publisher, 1-18

XPDF, 1-18

XSL-FO, 1-18