



Oracle One-to-One Fulfillment

Oracle One-to-One Fulfillment provides E-Business Suite business applications with a centralized mechanism for managing fulfillment. Fulfillment is the process of compiling and distributing information to customers. The fulfillment request identifies the content and the server. The Oracle One-to-One Fulfillment server processes the request. It compiles the content and sends the content to an output device. Output devices include email, fax, and print servers.

Oracle One-to-One Fulfillment consists of an Application programming interface (API) and a server. The Oracle One-to-One Fulfillment API is used to initiate a request for fulfillment processing.

The fulfillment request identifies the content and the server. The Oracle One-to-One Fulfillment server processes the request. It compiles the content and sends the content to an output device. Output devices include email, fax, and print servers.

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
- _ Account statements
- _ Lease terms
- _ Product information

Using the Oracle One-to-One Fulfillment Application programming interface (API), applications initiate a request for fulfillment processing. The business application:

- _ Generates a numerical identifier for the request
- _ Specifies the components of the content
- _ Specifies the output channel
- _ Specifies the fulfillment server
- _ Places the fulfillment request in the fulfillment request queue
- _ Creates a request status record
- _ Creates a request history record

The Oracle One-to-One Fulfillment server monitors the fulfillment request queue. It retrieves fulfillment requests from the queue and processes the requests. The server:

- _ Verifies the contact preferences of the recipient(s)
- _ Compiles the content
- _ Sends the content to an output device (email, fax, or print server)
- _ Updates the request status and history
- _ Creates a customer interaction record in Oracle Interaction History

Components of Fulfillment:

FULFILLMENT REQUESTS

A fulfillment request is the data object that is submitted for fulfillment processing. The E-Business Suite business 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 fulfillment server. The resulting data object is placed in the fulfillment request queue. Advanced Queuing (AQ) provides the fulfillment request queue.

EACH BUSINESS APPLICATION HAS A UNIQUE INTERFACE FOR MAKING A FULFILLMENT REQUEST ON BEHALF OF ONE OR MORE PARTIES.

FULFILLMENT SERVER

Fulfillment server is a multi-threaded Java server that processes fulfillment requests by dequeuing the request from an Advanced Queue. Business applications place fulfillment request in the queue using Fulfillment APIs.

You can specify the time period during which the server polls the fulfillment request queues in the database. It does not physically start or stop the server. If 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 create a interaction record in Oracle Interaction History. You can select which outcome reason and result your want to associate with a successful or failed fulfillment request.

Fulfillment Output Channels

The fulfillment server compiles the content and sends the content to an output device.

Output devices include email, fax, and print servers.

For email fulfillment requests, Oracle One-to-One Fulfillment supports any email server that uses SMTP (Simple Mail Transfer Protocol) to transmit e-mail messages.

For fax and print fulfillment requests, Oracle One-to-One Fulfillment supports Captaris RightFax.

One or more email servers can be associated with a Fulfillment Server, but only the email server flagged as the default will be used when processing email fulfillment requests.

If a site has multiple email servers, it is useful to associate more than one email server with each fulfillment server. In the event of an outage on one email server, the fulfillment server can easily be switched to use another email server, simply by changing the default.

Queries

A query is a SQL statement that generates selected data from a database. A query serves two purposes.

The first is to determine which records meet the defined criteria.

The second is to determine which fields of those records are returned.

The returned data is merged with a template to create a customized email for each recipient.

A single query may be associated with more than one template provided that the merge field names match the database column names or aliases and there is a one to one correspondence between the number of merge fields and the number of columns queried in the SQL statement.

Use the following syntax when creating an SQL query:

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

SQL statements typically end with a semicolon (;). However, in this case, the query engine appends the semicolon itself. 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. Alias 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 bits of 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 in from the user interface.

For example:

```
SELECT person_last_name lname,
party_id CustNumb,creation_date,phone_number
FROM jtf_contact_points_v
WHERE person_first_name = :fname;
```

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.

Creating a Master Document:

Use this procedure to create a Master Document in Oracle One-to-One Fulfillment. The Master Document can also be uploaded from the screen

Login

HTML Login URL

Responsibility

Fulfillment Administrator

Ex: Use the user fmadmin/welcome on findv115 and login to the application.

Navigate to: Template -> Master Document. And click on Upload to upload the master document. The UI will look as follows.

Item Id	Master Document Name	Query	Created By	Creation Date	Last Modified By	Last Modified Date
11658	PaymentSchedulescreen.htm	Query for Payment Schedules	FMADMIN	2003-10-17 15:37:01.0	FMADMIN	2003-10-17 15:37:04.0
11640	OKL.CureandRepurchaseRequest.htm	Cure and Repurchase Query	FMADMIN	2003-10-08 14:07:10.0	FMADMIN	2003-10-08 14:07:10.0
11639	OKL.RepurchaseRequest.htm	Repurchase Query	FMADMIN	2003-10-08 14:05:35.0	FMADMIN	2003-10-08 14:05:35.0
11638	OKL.CureRequestTemplate_v1.htm	Repurchase Query	FMADMIN	2003-10-08 13:45:51.0	FMADMIN	2003-10-08 13:45:53.0
11618	OKL.CureRequestTemplate_v1.htm	Cure Query	FMADMIN	2003-10-07 10:32:04.0	FMADMIN	2003-10-07 10:32:06.0

A sample Master Document will be as follows for the Payment Schedules Example:

Note that this master document is nothing but a HTML page with the Place Holders for the data that will be queried using a Query

Line Type	Asset Number	Stream Type	Amount	Currency Code
<line_type>	<asset_number>	<stream_type>	<amount>	<currency_code>

After uploading the master document we need to create the query and associate the query to a Master Document.

The UI for creating the Query will be as follows:

*Indicates required field

Validate SQL and save

The Query for the example that we are discussing will be as follows:

```
select contract_number,  
       to_char(sysdate,'DD-MON-YY') request_date,  
       Po_type,  
       Po_number,  
       Po_item_type,  
       po_amount,  
       po_currency_code  
from po_xx_po_summary_uv  
where po_id=:p_po_id
```

In this Query p_contract_id is a bind variable and the query uses the value in this bind variable at run time to fetch data from the Database that will be plugged into the master document and emailed to the user

Fulfillment Request History

The History subtab provides access to the history of fulfillment requests that have been processed, successfully or unsuccessfully, by a fulfillment server. You can view the details of a fulfillment request, re-submit a fulfillment request, forward a fulfillment request, and delete a history entry from the history list.

Fulfillment Request Status

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

Administration Tasks:

FULFILLMENT SERVERS

Use the Server subtab to configure fulfillment servers and output servers.

Creating a Fulfillment Server

Use this procedure to create and configure a fulfillment server.

Login

HTML Login URL

Responsibility

Fulfillment Administrator

Prerequisites

_ Install and implement a fulfillment output server.

Steps

1. Select the Server tab.

2. Select the Server subtab.

The Servers page appears.

3. Click **Create**.

The Server Details page appears.

4. Enter the details about the server.

a. 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.

b. Optionally, from the Start Time lists, select the hour and minute for the start time.

The start time is 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.

c. Optionally, from the Shutdown Time lists, select the hour and minute for the shutdown time.

The shutdown time is 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.

Steps

1. Click the Server tab.
2. Click the Server subtab.
The Servers page appears.
3. Click a fulfillment server name hyperlink.
The Server Details page appears.
4. From the Request Outcome list, select the outcome that you want to associate with a fulfillment request that has been retrieved from the request queue by the fulfillment server.
5. 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.
6. 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.
7. Click **Continue**.
The outcome and results are saved.

ENABLING THE UNSUBSCRIBE URL IN E-MAIL FULFILLMENT REQUESTS

Use this procedure to enable the fulfillment server to insert a hypertext link into the e-mail content that will allow recipients to access their contact preferences in Oracle Marketing.

Login

HTML Login URL

Responsibility

Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab.
2. Click the Server subtab.
The Servers page appears.
3. Click a fulfillment server name hyperlink.
The Server Details page appears.
4. In the side panel, click **Email Servers**.

The Email Servers page appears.

5. Select **Unsubscribe**.
6. Click **Update**.

ENABLING TRACKING FOR E-MAIL FULFILLMENT REQUESTS

Use this procedure to enable the fulfillment server to track e-mail statistic for batch and mass e-mail requests.

Login

HTML Login URL

Responsibility

Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab.
2. Click the Server subtab.
The Servers page appears.
3. Click a fulfillment server name hyperlink.
The Server Details page appears.
4. In the side panel, click **Email Servers**.
The Email Servers page appears.
5. Select **Track Email**.
6. Click **Update**.

ASSIGNING AN E-MAIL SERVER TO A FULFILLMENT SERVER

Use this procedure to assign an e-mail server to a fulfillment server.

Login

HTML Login URL

Responsibility

Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab.
2. Click the Server subtab.
The Servers page appears.
3. Click a fulfillment server name hyperlink.
The Server Details page appears.
4. In the side panel, click **Email Servers**.
The Email Servers page appears.
5. In the Summary of Email Server area, click **Add Email Server**.
The Create Email Server page appears.
6. In the Email Server Name field, type the name of the e-mail server.
You cannot change this field after the definition is saved.
7. In the Outgoing Mail Server field, type the name of the SMTP server.
This is the host name of the SMTP server.
8. In the Port Number field, type the port number that the fulfillment server will use to communicate with the e-mail server.
9. Optionally, in the Description field, type a description of the e-mail server
This field is for informational purposes only.
10. Click **Save**.
The Email Servers page appears.

REMOVING AN E-MAIL SERVER FROM A FULFILLMENT SERVER

Use this procedure to remove an e-mail server from a fulfillment server.

Login

HTML Login URL

Responsibility

Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Server tab.
2. Click the Server subtab.
The Servers page appears.
3. Click a fulfillment server name hyperlink.
The Server Details page appears.
4. In the side panel, click **Email Servers**.
The Email Servers page appears.
5. In the Summary of Email Server area, select the Remove box for the e-mail server that you want to remove.
6. Click **Update**.
The e-mail server is removed from the fulfillment server and the Email Servers page is refreshed.

FULFILLMENT REQUEST STATUS

Use the Status subtab to view the status of fulfillment requests that have been placed in the request queue, but not yet retrieved by the fulfillment server.

VIEWING THE STATUS OF A FULFILLMENT REQUEST

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

Login

HTML Login URL

Responsibility

Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Status tab.
2. Click the Status subtab.
The Request Status page appears.
3. From the View list, select a view type.
You have the following options:
 - _ Submitted
 - _ Re-submitted
 - _ In Process
 - _ Preview
4. Click a request identifier hyperlink.
The View Status page appears.
If the request is a single e-mail, the View Status page displays:
If the request type is batch or mass e-mail, the View Status page displays:

CANCELING A FULFILLMENT REQUEST

Use this procedure to cancel a fulfillment request in the queue.

Login

HTML Login URL

Note: Once the fulfillment server retrieves a fulfillment request from the request queue, the fulfillment request cannot be canceled.

Responsibility

Fulfillment Administrator

Prerequisites

None

Steps

1. Click the Status tab.
2. Click the Status subtab.
The Request Status page appears.
3. From the View list, select **submitted** or **Re-Submitted**.
4. In the Action column, click **Cancel**.

Script for starting up the Fulfillment Server in an instance:

Run the following command to start up the Fulfillment server, which picks up the request form the Fulfillment tables and sends it out to the Email Server.

```
java -ms32m -mx64m -Dengine.RequestDebug=true -Dengine.ProcessorDebug=true -  
Dengine.HistoryDebug=true -Dengine.MailDebug=true -Dlog.ShowWarnings -  
Dengine.LogLevel=6 -Dengine.LogRequests=true -Dengine.LogDispatches=true -  
Dengine.OutputDir=/home/rvaduri/work/fulfillment/findv115 -  
Dengine.CommandPromptEnabled=true -Dengine.ServerID=10001 -  
Dengine.AOLJ.config=/nfs/lafz/fin_top/findba/findv115/fnd/11.5.0/secure/ap102fam_findv115.dbc  
oracle.apps.jtf.fm.engine.processor.Processor
```