

Gateway Services User Guide  
**Oracle Banking Trade Finance**  
Release 14.7.2.0.0

**Part No. F90547-01**

November 2023

Gateway Services User Guide  
Oracle Financial Services Software  
Limited Oracle Park

Off Western Express  
Highway Goregaon (East)  
Mumbai, Maharashtra 400  
063 India  
Worldwide Inquiries:  
Phone: +91 22 6718 3000  
Fax: +91 22 6718 3001  
<https://www.oracle.com/industries/financial->

[services/index.html](#) Copyright © 2023, Oracle and/or

its affiliates. All rights reserved.

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

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

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

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

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

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

---

# Contents

<b>1. Preface .....</b>	<b>1-1</b>
1.1 Introduction .....	1-1
1.2 Audience .....	1-1
1.3 Documentation Accessibility .....	1-1
1.4 Acronyms and Abbreviations .....	1-1
1.5 Organization.....	1-1
1.6 Related Documents.....	1-2
1.7 Glossary of Icons .....	1-2
<b>2. Gateway Functions - An Overview.....</b>	<b>2-1</b>
2.1 Introduction .....	2-1
2.2 Inbound Application Integration .....	2-1
2.2.1 EJB Based Synchronous Deployment Pattern .....	2-1
2.2.2 Web Services Based Synchronous Deployment Pattern.....	2-2
2.2.3 HTTP Servlet Based Synchronous Deployment Pattern .....	2-2
2.2.4 MDB Based Asynchronous Deployment Pattern .....	2-2
2.3 Outbound Application Integration.....	2-3
2.4 Responsibilities of Integration Gateway.....	2-3
2.5 Deployment of Oracle Banking Trade Finance Integration Gateway.....	2-3
2.6 Deployment Patterns for Application Integration .....	2-4
<b>3. External System Maintenance.....</b>	<b>3-1</b>
3.1 Introduction .....	3-1
3.2 External System .....	3-1
3.2.1 Defining an External System.....	3-1
3.2.2 Specifying External System Details .....	3-2
3.2.3 Indicating Correlation Pattern.....	3-2
3.2.4 Specifying Message Exchange Pattern .....	3-2
3.2.5 Specifying Queue Details.....	3-3
3.2.6 Specifying External System Queue Details .....	3-3
3.2.7 Viewing External System Details .....	3-3
3.2.8 Accessing Services and Operations .....	3-4
3.3 Access Rights to an External System.....	3-5
3.3.1 Defining Access Rights to an External System.....	3-5
3.3.2 Viewing External System Function Details.....	3-6
3.3.3 Defining Access Rights to a User ID .....	3-7
3.3.4 Maintaining Upload Source Details.....	3-7
3.3.5 Specifying Upload Source Preferences .....	3-8
3.3.6 Maintaining Function ID Preferences .....	3-9
3.4 Gateway Password Generation Logic For External System Authentication .....	3-10
3.4.1 Password and Message ID as input.....	3-11
3.5 Incoming Message Browser.....	3-11
3.5.1 Viewing Incoming Message Details .....	3-13
3.6 Duplicate Message Browser .....	3-13
3.6.1 Querying Duplicate Message Browser.....	3-15
3.7 Outgoing Message Browser.....	3-16
3.7.1 Querying Outgoing Message Browser .....	3-19
3.8 Defining Notification Messages.....	3-19
3.8.1 Viewing Installed Notification Details .....	3-21

3.9	Defining the Notifications Enroutes .....	3-22
3.9.1	<i>Viewing Summary screen</i> .....	3-23
3.10	Amendment Maintenance .....	3-24
3.10.1	<i>Maintaining Gateway Amendment Details</i> .....	3-24
3.11	Gateway Maintenance for Approot Deployment .....	3-26
3.11.1	<i>Maintaining Gateway for Approot Deployment</i> .....	3-26
<b>4.</b>	<b>Message Formats</b> .....	<b>4-1</b>
4.1	Introduction .....	4-1
4.2	Oracle Banking Trade Finance Envelope .....	4-2
4.2.1	<i>FCUBS_HEADER</i> .....	4-3
4.2.2	<i>FCUBS_BODY</i> .....	4-5
4.3	Oracle Banking Trade Finance NOTIFICATION .....	4-6
4.3.1	<i>OBTF NOTIFICATION HEADER</i> .....	4-6
<b>5.</b>	<b>Annexure – List of Messages</b> .....	<b>5-1</b>
5.1	Introduction .....	5-1
5.2	Inbound Application Integration – Services .....	5-2
5.2.1	<i>OBTFBCService</i> .....	5-2
5.2.2	<i>OBTFLCService</i> .....	5-9
5.2.3	<i>OBTFCustomerService</i> .....	5-11
5.2.4	<i>OBTFIBService</i> .....	5-14
5.2.5	<i>OBTFLCMaintService</i> .....	5-20
5.2.6	<i>OBTFILService</i> .....	5-22
5.2.7	<i>OBTFMLService</i> .....	5-24
5.2.8	<i>OBTFMSService</i> .....	5-25
<b>6.</b>	<b>Function ID Glossary</b> .....	<b>6-1</b>

---

# 1. Preface

## 1.1 Introduction

This manual is designed to help you quickly get acquainted with the Integration Gateway of Oracle Banking Trade Finance.

It provides an overview to the module, and provides information on using the Integration Gateway module of Oracle Banking Trade Finance. This user manual is a supplement to the Common Core - Core Entities and Services User Manual and contains only specific functionalities and information related to Oracle Banking Trade Finance Gateway Services. Hence, this document should be read in conjunction with the Common Core - Core Entities and Services User Manual from the perspective of completeness in flow and understanding.

You can further obtain information specific to a particular field by placing the cursor on the relevant field and striking <F1> on the keyboard.

## 1.2 Audience

This manual is intended for the following User/User Roles:

Role	Function
IT department members responsible for integration	Input functions for maintenance related to the gateway
IT managers	Authorization functions

## 1.3 Documentation Accessibility

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

## 1.4 Acronyms and Abbreviations

Abbreviation	Description
OBTF	Oracle Banking Trade Finance

## 1.5 Organization

The manual is organized in the following manner:

Chapter	Description
Chapter 1	<i>About this Manual</i> gives information on the intended audience. It also lists the various chapters covered in this User Manual.
Chapter 2	<i>Gateway Functions - An Overview</i> explains the features of this module.





<b>Chapter 3</b>	<i>External System Maintenance</i> explains the maintenance of external systems and the various processes involved.
<b>Chapter 4</b>	<i>Message Formats</i> explains the different message formats.
<b>Chapter 5</b>	<i>Annexure – List of Messages</i> lists all the messages that you encounter while working with Oracle Banking Trade Finance Gateway services.
<b>Chapter 6</b>	<i>Planned Outage Support Server</i> describes the process of having a support server to substitute the host server during downtime.
<b>Chapter 7</b>	<i>Function ID Glossary</i> has alphabetical listing of Function/Screen ID's used in the module with page references for quick navigation.

## 1.6 Related Documents

- Core Entities User Manual
- Procedures User Manual
- Settlements User Manual
- Profit and Charges User Manual
- Common Core - Core Entities and Services

## 1.7 Glossary of Icons

This User Manual may refer to all or some of the following icons:

Icons	Function
	Exit
	Add row
	Delete row
	Option List

---

## 2. Gateway Functions - An Overview

### 2.1 Introduction

Integration of different applications and solutions is a key area in today's systems. A variety of specialized applications deployed on disparate platforms and using different infrastructure need to be able to communicate and integrate seamlessly with Oracle Banking Trade Finance in order to exchange data. The Oracle Banking Trade Finance Integration Gateway (referred to as 'Gateway' in the rest of the document) will cater to these integration needs.

The integration needs supported by the Gateway can be broadly categorized from the perspective of the Gateway as follows:

- Inbound application integration – used when any external system needs to add, modify or query information within Oracle Banking Trade Finance.
- Outbound application integration – used when any external system needs to be notified of the various events that occur within Oracle Banking Trade Finance.

### 2.2 Inbound Application Integration

Oracle Banking Trade Finance Inbound Application Gateway provides XML based interfaces thus enhancing the need to communicate and integrate with the external systems. The data exchanged between Oracle Banking Trade Finance and the external systems will be in the form of XML messages. These XML messages are defined in OBTF in the form of XML Schema Documents (XSD) and are referred to as 'OBTF formats'

*For more information on OBTF formats refer the Message Formats chapter in this User Manual.*

OBTF Inbound Application Integration Gateway uses the Synchronous and Asynchronous Deployment Pattern for addressing the integration needs.

The Synchronous Deployment Pattern is classified into the following:

- Oracle Banking Trade Finance EJB Based Synchronous Inbound Application Integration Deployment Pattern
- Oracle Banking Trade Finance Web Services Based Synchronous Inbound Application Integration Deployment Pattern
- Oracle Banking Trade Finance HTTP Servlet Based Synchronous Inbound Application Integration Deployment Pattern

Asynchronous Deployment Pattern is:

- Oracle Banking Trade Finance MDB Based Asynchronous Inbound Application Integration Deployment Pattern

#### 2.2.1 EJB Based Synchronous Deployment Pattern

The Enterprise Java Beans (EJB) deployment pattern will be used in integration scenarios where the external system connecting to Oracle Banking Trade Finance is 'EJB literate', i.e., the external system is capable of interacting with Oracle Banking Trade Finance based upon the EJB interface. In this deployment pattern, the external system will use the RMI/IIOP protocol to communicate with the Oracle Banking Trade Finance EJB.

In this deployment pattern the EJB displayed by Oracle Banking Trade Finance will be a

stateless session bean. The actual request will be in the form of an XML message. After the necessary processing is done in Oracle Banking Trade Finance based on the request, the response is returned to the external system as an XML message. The transaction control for the processing will stay with the Oracle Banking Trade Finance EJB.

### **2.2.2 Web Services Based Synchronous Deployment Pattern**

The web services deployment pattern will be used in integration scenarios where the external system connecting to Oracle Banking Trade Finance wants to connect using standards-based, inter-operable web services.

This deployment pattern is especially applicable to systems which meet the following broad guidelines:

- Systems that are not 'EJB literate', i.e., such systems are not capable of establishing connections with Oracle Banking Trade Finance based upon the EJB interface; and/or
- Systems that prefer to use a standards-based approach

In this deployment pattern, the external system will use the SOAP (Simple Object Access Protocol) messages to communicate to the Oracle Banking Trade Finance web services.

The services displayed by Oracle Banking Trade Finance are of a 'message based' style, i.e., the actual request will be in the form of an XML message, but the request will be a 'payload' within the SOAP message. After the necessary processing is done in Oracle Banking Trade Finance based on the request, the response is returned to the external system as an XML message which will be a 'payload' within the response SOAP message. The transaction control for the processing will stay with the Oracle Banking Trade Finance.

### **2.2.3 HTTP Servlet Based Synchronous Deployment Pattern**

The HTTP servlet deployment pattern will be used in integration scenarios where the external system connecting to Oracle Banking Trade Finance wants to connect to Oracle Banking Trade Finance using simple HTTP messages.

This is especially applicable to systems such as the following:

- Systems that are not 'EJB literate', i.e., are not capable establishing a connections with Oracle Banking Trade Finance based upon the EJB interface; and/or
- Systems that prefer to use a simple http message based approach without wanting to use SOAP as the standard

In this deployment pattern, the external system will make an HTTP request to the Oracle Banking Trade Finance servlet.

For this deployment pattern, Oracle Banking Trade Finance will display a single servlet. The actual request will be in the form of an XML message. This XML message is embedded into the body of the HTTP request sent to the Oracle Banking Trade Finance servlet. After the necessary processing is done in Oracle Banking Trade Finance based on the request, the response is returned to the external system as an XML message which is once again embedded within the body of the response HTTP message. The transaction control for the processing will stay with the Oracle Banking Trade Finance.

### **2.2.4 MDB Based Asynchronous Deployment Pattern**

The MDB deployment pattern is used in integration scenarios where the external system connecting to Oracle Banking Trade Finance wants to connect to Oracle Banking Trade Finance using JMS queues.



This is especially applicable to systems such as the following:

- Systems that prefer to use JMS queues based approach without wanting to wait for the reply

Here external system sends messages in XML format to request queue on which an MDB is listening. When a message arrives on the queue, it is picked up for processing. After the necessary processing is done in Oracle Banking Trade Finance, based on the request, the response is sent to the response queue as an XML message

## **2.3 Outbound Application Integration**

The Outbound Application Integration is also called the Oracle Banking Trade Finance Notify Application Integration layer. This application layer sends out notification messages to the external system whenever events occur in Oracle Banking Trade Finance.

The notification messages generated by OBTF on the occurrence of these events will be XML messages. These XML messages are defined in OBTF in the form of XML Schema Documents (XSD) and are referred to as 'OBTF formats'

*For more information on OBTF formats refer the Message Formats chapter in this module.*

## **2.4 Responsibilities of Integration Gateway**

The primary responsibilities of Oracle Banking Trade Finance Integration Gateway include the following:

- Authentication
- Duplicate recognition
- Validation
- Routing
- Logging of messages

## **2.5 Deployment of Oracle Banking Trade Finance Integration Gate- way**

Message communication - incoming or outgoing from/to an external system in Oracle Banking Trade Finance will happen only through an Oracle Banking Trade Finance Integration Gateway. Hence, it becomes the first point of contact or last point of contact with the database in message flow. The Oracle Banking Trade Finance Integration Gateway can be deployed to support both the distributed and single schema deployments of Oracle Banking Trade Finance:

- Distributed deployment of OBTF – In this situation the database components of the Gateway are deployed as two or more schemas
  - The messaging schema as part of SMS schema in the SMS and/or HO instance
  - The business schema(s) in the various branch schemas in the branch instance(s)
- Single schema deployment of OBTF – In this situation the database components of the Gateway (messaging and business) are both deployed as part of the single Oracle Banking Trade Finance schema.

## 2.6 Deployment Patterns for Application Integration

Business Integration Needs	Nature of Integration	Oracle Banking Trade Finance Deployment Pattern	Remarks
Inbound Transactions into Oracle Banking Trade Finance	Synchronous	FLEXCUBE OBTF EJB	Recommended
		FLEXCUBE OBTF HTTP Servlet	This can be used if the external system cannot communicate to Oracle Banking Trade Finance using EJB.
		FLEXCUBE OBTF Web Services	This can be used if the external system chooses to communicate only through Web Services.
	Asynchronous	FLEXCUBE OBTF MDB	This can be used if the external system chooses to communicate only through JMS queues
Inbound Queries into Oracle Banking Trade Finance	Synchronous	FLEXCUBE OBTF EJB	Recommended
		FLEXCUBE OBTF In Servlet	This can be used if the external system cannot communicate to Oracle Banking Trade Finance using EJB.
		FLEXCUBE OBTF Web Services	This can be used if the external system chooses to communicate only through Web Services.
	Asynchronous	FLEXCUBE OBTF MDB	This can be used if the external system chooses to communicate only through JMS queues
Handoffs from Oracle Banking Trade Finance	Asynchronous	FLEXCUBE OBTF Notify	Recommended

---

## 3. External System Maintenance

### 3.1 Introduction

This chapter contains the following sections:

- [Section 3.2, "External System"](#)
- [Section 3.3, "Access Rights to an External System"](#)
- [Section 3.4, "Gateway Password Generation Logic For External System Authentication"](#)
- [Section 3.5, "Incoming Message Browser"](#)
- [Section 3.6, "Duplicate Message Browser"](#)
- [Section 3.7, "Outgoing Message Browser"](#)
- [Section 3.8, "Defining Notification Messages"](#)
- [Section 3.9, "Defining the Notifications Enroutes"](#)
- [Section 3.10, "Amendment Maintenance"](#)
- [Section 3.11, "Gateway Maintenance for Approval Deployment"](#)

### 3.2 External System

This section contains the following topics:

- [Section 3.2.1, "Defining an External System"](#)
- [Section 3.2.2, "Specifying External System Details"](#)
- [Section 3.2.3, "Indicating Correlation Pattern "](#)
- [Section 3.2.4, "Specifying Message Exchange Pattern"](#)
- [Section 3.2.5, "Specifying Queue Details"](#)
- [Section 3.2.6, "Specifying External System Queue Details"](#)
- [Section 3.2.7, "Viewing External System Details"](#)
- [Section 3.2.8, "Accessing Services and Operations "](#)

#### 3.2.1 Defining an External System

You need to define an external system that will communicate with the Oracle Banking Trade Finance Integration Gateway.

You can define an external system using the **External System Maintenance** page

You can invoke this screen by typing 'GWDETSYS' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

The various details required by the **External System Maintenance** page are described below.

### 3.2.2 **Specifying External System Details**

#### **External System**

Specify a name for the external system. This should be the same as the Source in an incoming message.

#### **Description**

Specify a brief description for the External System.

### 3.2.3 **Indicating Correlation Pattern**

#### **Request**

You can define a way in which the external system should correlate its request message with the response message. You can choose Message ID of a request message as the Correlation ID in the response message. Alternatively, you can choose Correlation ID of a request message and maintain it as the Correlation ID of the corresponding response message.

### 3.2.4 **Specifying Message Exchange Pattern**

#### **Request Message**

You can choose the Request message to be 'Full Screen' or 'Input Only'. If you select 'Full Screen' as the request message, the response message will also display 'Full Screen'.

#### **Response Message**

You can choose the Response message to be 'Full Screen' or 'Record Identification Msg'.

**Note**

You can select 'Record Identification Msg' as the response message only if you select 'Input Only' in the request message.

### 3.2.5 **Specifying Queue Details**

**Default Response Queue**

You can define a response queue for each of the In Queue's through which the External System will communicate with Oracle Banking Trade Finance. Define a valid queue name as the Default Response Queue.

**Dead Letter Queue**

If the messages received are non-readable, such messages are directed to Dead Letter Queue defined for the external system.

**Note**

If the Dead Letter Queue is not defined, such messages will be redirected to a queue with the name of the request queue appended with '\_E'.

**XSD Validation Required**

Check this box to indicate if the request message should be validated against its corresponding XSD.

**Register Response Queue Message ID**

Check this box to indicate if the message ID provided by the Response Queue should be logged when a response message is posted into the queue.

### 3.2.6 **Specifying External System Queue Details**

The 'External System Queues' list contains the 'In Queue' and 'Response Queue' lists. To add a record to the 'External System Queues' list click add icon. To delete a record from the list, select the record using its check box and then click delete icon.

**In Queue**

Specify the name of the queue from which the messages were received. The name of the queue will help identify the external system.

**Note**

This is required only if an incoming message does not display the source of the message. An In Queue is mapped to only one External System.

You can map multiple queues to a source. System will allow a source to post messages to multiple queues.

**Response Queue**

You can define Response Queue for every In Queue. This is required only when the External System fails to display the queue name on posting a request message into the In Queue.

### 3.2.7 **Viewing External System Details**

The details of previously defined external Systems can be viewed using the 'External System - Summary' screen as shown below. You can invoke this screen by typing 'GWSETSYS' in

the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

This summary screen can be used to search for external systems which match the data specified for any of the following criteria:

- Authorization Status

The screenshot shows the 'External System Summary' window. At the top, there are buttons for 'Search', 'Advanced Search', 'Reset', and 'Clear All'. A 'Records per page' dropdown is set to 15. Below these is a 'Search (Case Sensitive)' section with five search criteria: 'Authorization Status', 'Record Status', 'External System', 'Default Response Queue', and 'Dead Letter Queue'. Each criterion has a corresponding input field with a search icon. Below the search criteria is a 'Search Results' section with a 'Lock Columns' dropdown set to 0. A table header is visible with columns: 'Authorization Status', 'Record Status', 'External System', 'Default Response Queue', and 'Dead Letter Queue'. The table body shows 'No data to display.' At the bottom, there is a pagination bar showing 'Page 1 Of 1' and navigation icons. An 'Exit' button is located in the bottom right corner.

- External System
- Dead Letter Queue
- Record Status
- Default Response Queue

The 'Result' list shows the external systems which match your query. The search functions available are:

#### **Advanced**

Click **Advanced** to specify queries with logical operators such as AND, OR and NOT.

#### **Reset**

Click **Reset** to empty the values in the criteria fields, so that you may begin a new search.

#### **Query**

After specifying your search criteria click **Query** to view the list of results which match your search criteria.

#### **Refresh**

Click **Refresh** to refresh the list of results.

### **3.2.8 Accessing Services and Operations**

In a message it is mandatory to maintain a list of Service Names and Operation Codes. This information is called Gateway Operations.

A combination of every such Service Name and Operation Code is mapped to a combination of Function ID and Action. Every screen in Oracle Banking Trade Finance is linked with a function ID. This information is called Gateway Functions.

You can gain access to an external system using the Gateway Functions. The Function IDs mapped in Gateway Functions should be valid Function IDs maintained in Oracle Banking Trade Finance. Hence, for every new Service or Operation being introduced, it is important that you provide data in Gateway Operations and Gateway Functions.

Refer the Message Formats chapter for more information on Services and Operations.

## 3.3 **Access Rights to an External System**

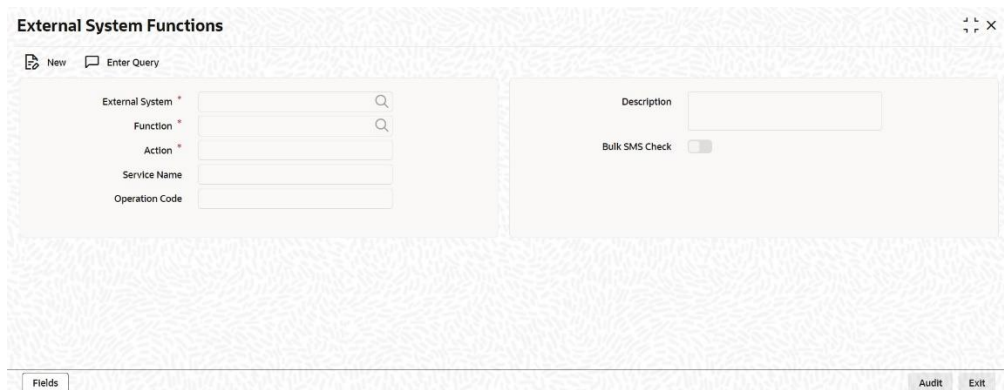
This section contains the following topics:

- [Section 3.3.1, "Defining Access Rights to an External System"](#)
- [Section 3.3.2, "Viewing External System Function Details"](#)
- [Section 3.3.3, "Defining Access Rights to a User ID"](#)
- [Section 3.3.4, "Maintaining Upload Source Details"](#)
- [Section 3.3.5, "Specifying Upload Source Preferences"](#)
- [Section 3.3.6, "Maintaining Function ID Preferences"](#)

### 3.3.1 **Defining Access Rights to an External System**

You can define access rights to an external system using the 'External System Functions – Detailed' screen.

You can invoke this screen by typing 'GWDETFUN' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.



In the above screen, you need to specify the following details:

## External System

Select an external system for which you wish to provide access rights. The adjoining option list displays all the external systems you have maintained in the 'External Systems – Detailed' screen.

## Function ID

Select a Function ID from the list of values by clicking the adjoining option list. The function ids are invoked from Gateway Functions.

## Action

Select an action for the external system from the option list provided.

## Service Name

This displays the service name based on the Function ID and Action you select.

## Operation Code

This displays the Operation Code based on the Function ID and Action you select.

### 3.3.2 Viewing External System Function Details

You can view the access rights details which have already been defined using the 'External System Functions - Summary' screen as shown below. You can invoke this screen by typing 'GWSETFUN' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

External System Function Summary

Search Advanced Search Reset Clear All Records per page 15

Search (Case Sensitive)

Authorization Status Record Status Function Action External System

Search Results Lock Columns 0

Authorization Status	Record Status	Function	Action	External System
No data to display.				

Page 1 Of 1 < >

Exit

This summary screen can be used to search for external system functions which match any of the following criteria:

- Authorization Status
- External System
- Record Status
- Action
- Function

The 'Result' list shows the external system functions which match your query. The search functions available are:



### Advanced

Click **Advanced** to specify queries with logical operators such as AND, OR and NOT.

### Reset

Click **Reset** to empty the values in the criteria fields, so that you may begin a new search.

### Query

After specifying your search criteria click **Query** to view the list of results which match your search criteria.

### Refresh

Click **Refresh** to refresh the list of results.

## 3.3.3 Defining Access Rights to a User ID

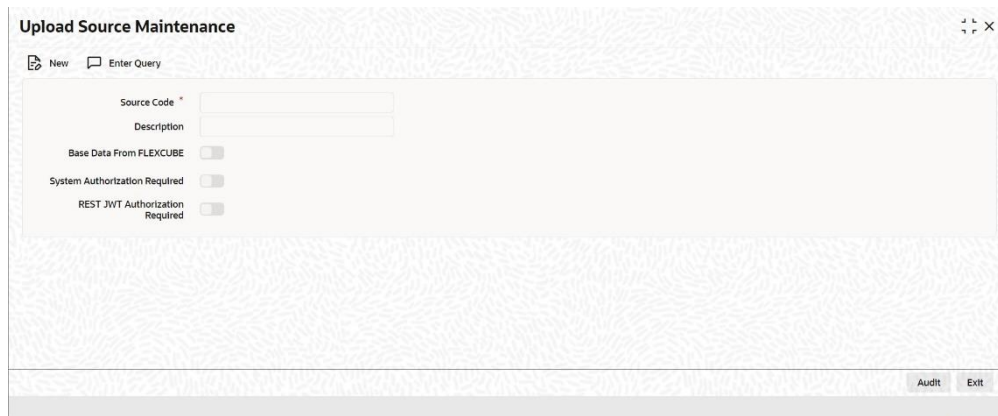
For a User ID to access a request message, you can individually map each Function ID to a User ID.

### Note

You can provide access to functions only if the Function IDs provided in Gateway Functions are valid.

## 3.3.4 Maintaining Upload Source Details

Oracle Banking Trade Finance facilitates upload of data from an external source. The details of the source from which data has to be uploaded need to be maintained in Oracle Banking Trade Finance using the 'Upload Source Maintenance' screen. You can invoke the 'Upload Source Maintenance' screen by typing 'CODSORCE' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.



The following details need to be captured here:

### Source Code

Specify a code for the source from which data has to be uploaded to Oracle Banking Trade Finance.

## Description

Give a small description for the source code specified.

## Base Data From FLEXCUBE

Check this box to indicate if base data has to be uploaded from Oracle Banking Trade Finance.

### 3.3.5 Specifying Upload Source Preferences

You can set preferences for upload of data from an external source in the 'Upload Source Preferences Maintenance' screen. You can invoke the 'Upload Source Preferences Maintenance' screen by typing 'CODUPLDM' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

The following details are captured here:

#### Source Code

Select Source Code from the option list. Depending on the source code you select here data is uploaded from that source into Oracle Banking Trade Finance.

#### Module Code

You can choose to upload data from a source directly onto a module in Oracle Banking Trade Finance. Indicate the module into which you would like to upload data from a given source.

#### On Override

Oracle Banking Trade Finance generates override messages in case it encounters any discrepancies during data upload. You can choose to do any of the following:

- Ignore – Select this option to ignore such error messages and continue with the upload process.
- Override – Select this option to put the record on hold for user intervention later.
- Reject – Select this option to reject the record.

#### On Exception

In case a serious error occurs during data upload, Oracle Banking Trade Finance generates an error message. You can choose to put the record with the error on hold. In such a case, choose

'Move To Queue' from the list of options available. If you would like to reject the record altogether, choose 'Reject'.

### Post Upload Status

If you would like to automatically authorize the data that is uploaded into Oracle Banking Trade Finance choose the 'Authorize' option here.

If you would like the record to be put on hold choose this option in this field.

If you would like the record to be unauthorized, choose the 'Unauthorized' option in this field. The record will not be authorized automatically on upload. You will have to manually authorize the data.

### Purge Days (Calendar)

Specify the days maintained for purging of the data uploaded.

### Allow Deferred Processing

Check this option to defer processing of amendment and cancellation uploads.

### Allow EOD with Deferred

Check this option to proceed even if the records exist in the deferred processing log.

If it is unchecked, then the EOD process halts until the deferred process log is cleared.

### Allow Delete

Check this option to delete the process log.

## 3.3.6 Maintaining Function ID Preferences

Click 'Function ID Preferences' button in the 'Upload Source Preferences Maintenance' screen.

The screenshot displays a web-based interface for maintaining function ID preferences. It features a table with the following headers: 'Function', 'Status', 'On Queue Exception', 'On Override', and 'Proceed with EOD'. The table body is empty, with a message 'No data to display.' below the headers. A pagination bar at the bottom of the table indicates 'Page 1 (0 of 0 items)' and includes navigation icons. The interface is enclosed in a window with a close button (X) in the top right corner. At the bottom right of the window, there are 'Exit' and 'Save' buttons.

You can enter the following details:

### Function Id

Specify the function ID. The adjoining option list displays all the function IDs maintained in the system. Select the appropriate one.

**Status**

Select the status of the record from the drop-down list and the available options are:

- Authorized
- Unauthorized
- Put On hold

**On Exception**

Select the record status in case of any exception while saving from the drop-down list and the available options are:

- Put On Hold
- Reject

**On Override**

Select the record status in case of any overriding a saved record from the drop-down list and the available options are:

- Put On Hold
- Reject
- Ignore

**Proceed With EOD**

Check this box, if you wish to proceed with EOD.

**Deleted Allowed**

Check this box, if deleting is allowed.

**Reverse Allowed**

Check this box, if reversal is allowed.

**Amend Allowed**

Check this box, if amendment is allowed.

**Purge Days (Calendar)**

Specify the calendar purge days

**Allow Deferred Processing**

Check this box, if deferred processing is allowed.

**Allow EOD with Deferred**

Check this box, if EOD with deferred is allowed.

### **3.4 Gateway Password Generation Logic For External System Authentication**

This section describes the Encryption information for gateway authentication.

If Password authentication is required for an external system, then check the 'System Authentication Required' check box while maintaining the External System details in CODSORCE.

Once this check box is checked Oracle Banking Trade Finance validates for Encrypted password as part of every request sent by the External System. The password encryption logic as below:

### 3.4.1 Password and Message ID as input

Message ID, present as part of the header in Request XML, is considered as hash. External System generates an unique Message ID, which is functional mandatory field in the header. Create a Message Digest with SHA-512 algorithm.

The hash created from the previous step and the password in clear text together is encrypted in DESede encryption method. Apply Base64 encoding to encrypted value and send to the Oracle Banking Trade Finance gateway.

## 3.5 Incoming Message Browser

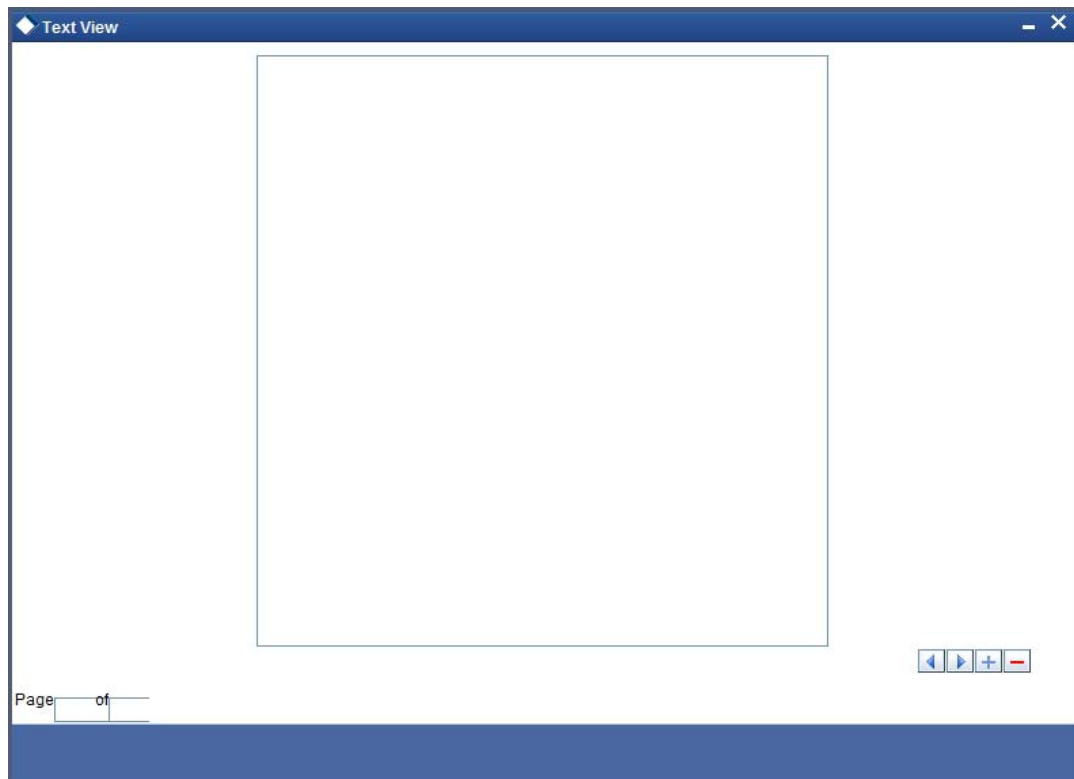
The messages received from the external system will be displayed in the Incoming Message Browser.

You can invoke the 'Incoming Message Browser' screen by typing 'GWDINBRW' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

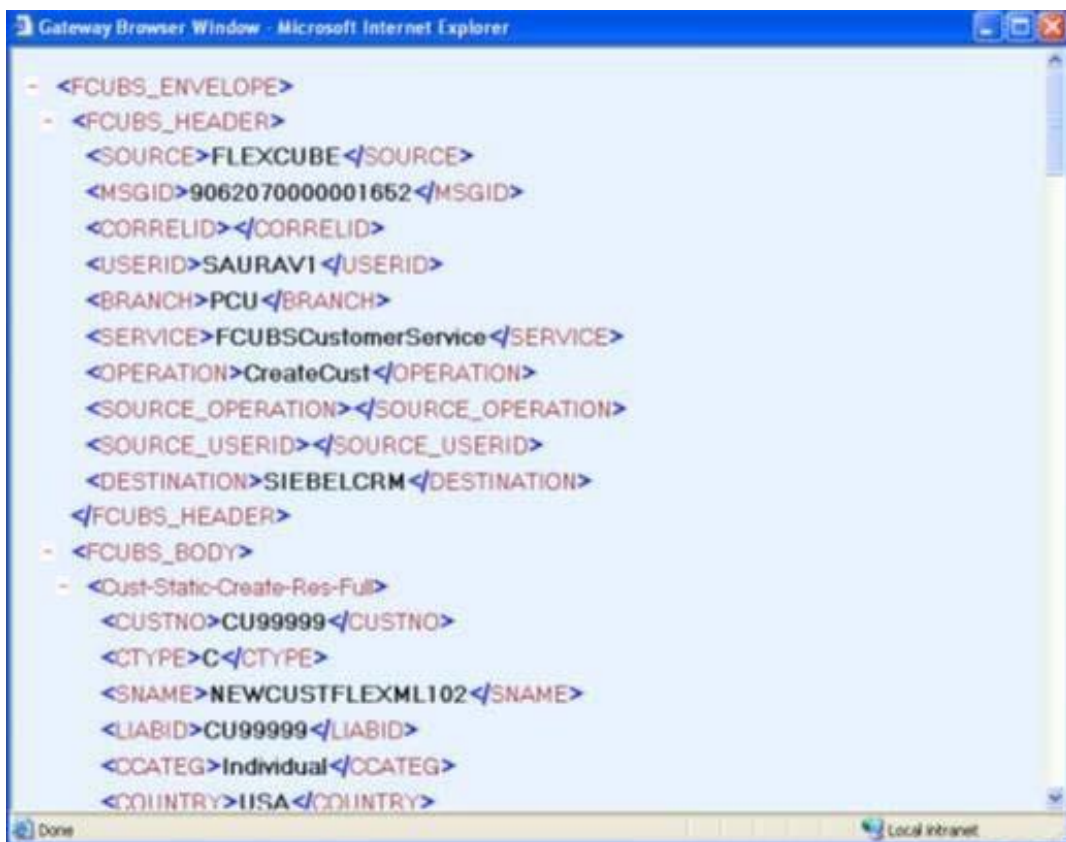
The screenshot shows the 'Incoming Message Browser - Detail' window. It features a search bar at the top left with a magnifying glass icon and the text 'Enter Query'. The main area is split into two columns. The left column, titled 'Message Reference', contains input fields for 'Message ID', 'Message Status', 'Operation Code', 'FLEXCUBE Reference', 'Item User ID', 'Queue Name', and 'Request Queue Message ID'. The right column, titled 'External System', contains input fields for 'Correlation ID', 'Service Name', 'Branch', 'User ID', 'Branch Date', and 'Server Date Stamp'. Below these columns is a 'Request Reason' field. At the bottom of the window, there are two buttons: 'XML View' and 'Text View'. The window has a standard title bar with the text 'Incoming Message Browser - Detail' and a close button in the top right corner.

In the 'Incoming Message Browser' screen you can view the details of the messages received from the external systems. You can also view the messages in the XML format or the Text format.

Click 'Text View' button to view the incoming messages in text format as shown below:



Click 'XML View' button to view the 'Gateway Browser Window' screen which displays the messages in XML format.



### 3.5.1 Viewing Incoming Message Details

The summary of all messages received from the external system can be viewed using the 'Incoming Message Browser - Summary' screen as shown below. You can invoke this screen by typing 'GWSINBRW' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

This summary screen can be used to search for incoming messages which match the criteria (Message Reference Number, External System, Service Name etc) you specify. The 'Result' list shows the messages which match your query. The search functions available are:

#### **Advanced**

Click **Advanced** to specify queries with logical operators such as AND, OR and NOT.

#### **Reset**

Click **Reset** to empty the values in the criteria fields, so that you may begin a new search.

#### **Search**

Specify your search criteria to view the list of results which match your search criteria.

## 3.6 Duplicate Message Browser

If Oracle Banking Trade Finance Integration Gateway identifies a duplicate message it will be sent to the Duplicate Messages Browser. You can invoke the 'Duplicate Message Browser - Summary'

screen by typing 'GWSPBROW' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

Duplicate Message Browser Summary

Search Advanced Search Reset

Case Sensitive

Message Reference Branch

External System Message Id

Correlation Id Request Queue Message Id

Service Name Operation Code

User Id Their User Id

Server Date Stamp Status

Records per page 15 First Previous 1 Of 1 Next Last Go Lock Columns 0

Message Reference	Branch	External System	Message Id	Correlation Id	Request Queue Message Id	Service Name	Operation Code	User Id	Their User Id	Media	Serve
-------------------	--------	-----------------	------------	----------------	--------------------------	--------------	----------------	---------	---------------	-------	-------

Exit

Click 'Text View' button in the 'Duplicate Message Browser - Summary' screen to view the duplicate messages in the text format.

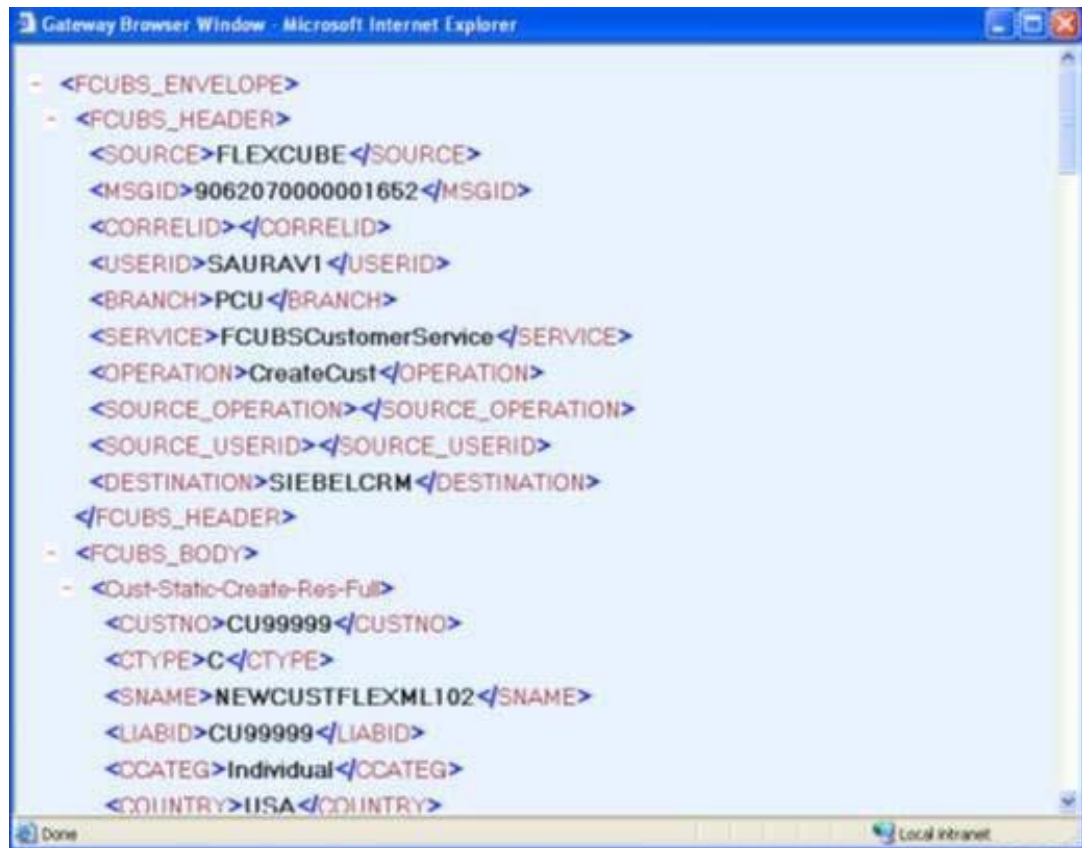
Duplicate Message Browser - Text View

Page of

< > + -



Click 'XML View' button in the 'Duplicate Message Browser - Summary' screen to view the duplicate messages in the following format:



### 3.6.1 Querying Duplicate Message Browser

You can query the list of duplicate messages using the 'Duplicate Message Browser - Summary' screen with functions for search. You can invoke this screen by typing

'GWSPBROW' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

This query screen can be used to search for duplicate messages which match the criteria (Message Reference Number, External System, Service Name etc) you specify. The 'Result' list shows the messages which match your query. The search functions available are:

#### Advanced

Click **Advanced** to specify queries with logical operators such as AND, OR and NOT.

#### Reset

Click **Reset** to empty the values in the criteria fields, so that you may begin a new search.

#### Query

After specifying your search criteria click **Query** to view the list of results which match your search criteria.

#### Refresh

Click **Refresh** to refresh the list of results.

## 3.7 Outgoing Message Browser

Once the incoming messages have been processed, a response message will be sent to the external systems along with the status of the processed messages. The response messages will be displayed in the 'Outgoing Message Browser'. You can invoke the 'Outgoing Message

Browser' screen by typing 'GWDOTBRW' in the field at the top right corner of the Application tool bar and clicking on the adjoining arrow button.

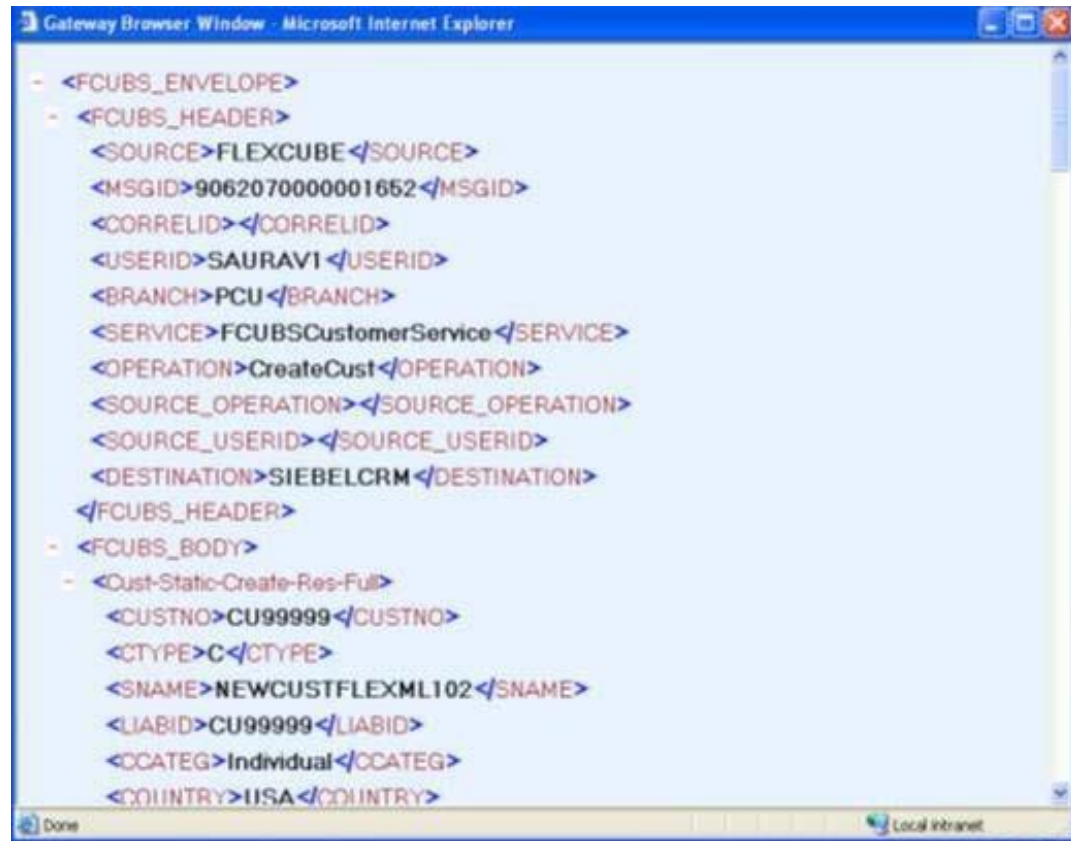
The 'Outgoing Message Browser' window features a search bar at the top with a magnifying glass icon and a close button. Below the search bar, there are two main sections for filtering messages. The left section, titled 'Message Reference', includes fields for Message Id, Message Status, Operation Code, FLEXCUBE Reference, Their User Id, Response Queue Message Id, and Queue Name. The right section, titled 'External System', includes fields for Correlation Id, Service Name, Branch, User Id, Server Date Stamp, Branch Date, and Related Message Reference. At the bottom of the filter sections, there is a 'Repeat Reason' field and two buttons: 'Text View' and 'XML View'. An 'Exit' button is located in the bottom right corner of the window.

In the 'Outgoing Message Browser' screen you can view the details of the messages sent to the external systems. You can also view the messages in the XML format or the Text format.

Click 'Text View' button to view the response message in text format as shown below:

The 'Text View' window displays a large, empty rectangular area for viewing the message content. At the bottom left, there is a 'Page' field followed by 'of' and another empty field. At the bottom right, there are four navigation buttons: a left arrow, a right arrow, a plus sign, and a minus sign. The window has a blue title bar with the text 'Text View' and standard window control buttons (minimize, maximize, close).

Click 'XML View' button to view the response messages in XML format as shown below:



### 3.7.1 Querying Outgoing Message Browser

You can query the list of outgoing messages using the 'Outgoing Message Browser - Summary' screen. You can invoke this screen by typing 'GWSOTBRW' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

This query screen can be used to search for outgoing messages which match the criteria (Message Reference Number, External System, Service Name etc) you specify. The 'Result' list shows the messages which match your query. The search functions available are:

#### **Advanced**

Click **Advanced** to specify queries with logical operators such as AND, OR and NOT.

#### **Reset**

Click **Reset** to empty the values in the criteria fields, so that you may begin a new search.

#### **Search**

Specify your search criteria to view the list of results which match your search criteria.

## 3.8 Defining Notification Messages

The outgoing message browser also displays the notification messages. Every time the database layer receives a notification message, it updates the outgoing message browser to display the same.

At the branch level, you can maintain notification codes for each branch. You can do this using the 'Notifications Installed' screen.

You can invoke the 'Notifications Installed – Detailed' screen by typing 'GWDNTFIN' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

The screenshot shows the 'Notifications Installed' application window. The title bar is blue with a diamond icon and the text 'Notifications Installed'. Below the title bar is a menu bar with 'New' and 'Enter Query' options. The main area contains four input fields: 'Branch Code \*', 'Notification Code \*', 'Branch Name', and 'Description'. Each field has a small icon to its right. At the bottom, there is a 'Fields' section with a list of fields: 'Input By Date Time', 'Authorized By Date Time', 'Modification Number', 'Authorized', and 'Open'. An 'Exit' button is located in the bottom right corner.

In this screen, you need to specify the following details:

#### **Branch Code**

Select a branch code for which you wish to assign a notification code. Once you select the branch code, the corresponding description is displayed.

#### **Notification Code**

From the list of values, select a notification code you wish to link with the branch. This will be used to generate notifications for the entire branch. Once you select the notification code, the corresponding description is displayed.

In addition to maintaining a notification code for a given branch, you can also specify a topic or a queue to which the notification messages should be sent.

You can define a topic/queue for a given branch and notification code in the 'Notifications Enroute' screen.

### 3.8.1 Viewing Installed Notification Details

You can view and query the list of notification messages using the 'Notifications Installed - Summary' screen. You can invoke this screen by typing 'GWSNTFIN' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

Notifications Installed Summary

Search Advanced Search Reset

Case Sensitive

Authorization Status  Record Status

Branch Code  Notification Code

Records per page: 15 First Previous 1 Of 1 Next Last Go Lock Columns: 0

Authorization Status	Record Status	Branch Code	Notification Code
----------------------	---------------	-------------	-------------------

Authorization Status: A - Authorized U - Unauthorized

Record Status: C - Closed O - Open

Exit

This summary screen can be used to search for notification messages which match the criteria (Message Reference Number, External System, Service Name etc) you specify. The 'Result' list shows the messages which match your query. The search functions available are:

#### **Advanced**

Click **Advanced** to specify queries with logical operators such as AND, OR and NOT.

#### **Reset**

Click **Reset** to empty the values in the criteria fields, so that you may begin a new search.

#### **Search**

Specify your search criteria to view the list of results which match your search criteria.

## 3.9 Defining the Notifications Enroutes

You can invoke the 'Notification Enroutes - Detailed' screen by typing 'GWDNTFEN' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

The screenshot shows the 'Trade Notification Enroute Maintenance' application window. It features a title bar with standard window controls. Below the title bar, there are two buttons: 'New' and 'Enter Query'. The main content area is divided into two panels. The left panel contains three input fields: 'Branch Code \*', 'Description', and 'Destination Name \*'. The right panel contains two input fields: 'Notification Code \*' and 'Notification Description'. At the bottom of the screen, there are three buttons: 'Fields', 'Audit', and 'Exit'.

In 'Notification Enroutes - Detailed' screen, you define the following details:

### **Branch Code**

Specify a branch for which you wish to define a topic. If you want to use a single topic for all branches and for a given notification code, you can use a wildcard '\*\*\*'. Once you select a Branch Code, the corresponding description is displayed.

### **Notification Code**

Select a notification code from the list of notification codes provided. You can indicate a wildcard '\*\*\*' if a single topic needs to be used for all notifications in a branch.

### **Destination Name**

Specify the destination name.

Once you have captured the details, save the maintenance.



### 3.9.1 Viewing Summary screen

You can view and query the list of notifications enroute using the 'Notifications Enroutes - Summary' screen. You can invoke this screen by typing 'GWSNTFEN' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

**Trade Notification Enroute Summary**

Search Advanced Search Reset Clear All Records per page 15

Search (Case Sensitive)

Authorization Status Record Status Branch Code  
Destination Name Notification Code

Search Results Lock Columns 0

Authorization Status	Record Status	Branch Code	Destination Name	Notification Code
No data to display.				

Page 1 Of 1 < >

Exit

This summary screen can be used to search for notifications enroute which match the criteria (Message Reference Number, External System, Service Name etc) you specify. The 'Result' list shows the messages which match your query. The search functions available are:

#### **Advanced**

Click **Advanced** to specify queries with logical operators such as AND, OR and NOT.

#### **Reset**

Click **Reset** to empty the values in the criteria fields, so that you may begin a new search.

#### **Search**

Specify your search criteria to view the list of results which match your search criteria.

## 3.10 Amendment Maintenance

This section contains the following topics:

- [Section 3.10.1, "Maintaining Gateway Amendment Details"](#)

### 3.10.1 Maintaining Gateway Amendment Details

To recall, you need to identify the fields that can be amended by an external system, say, a Siebel CRM application.

Every amendment request coming from such a system has the following data:

- **Service Name:** – This is a broad level grouping of similar operations within a module in Oracle Banking Trade Finance. The service names are published by Oracle Banking Trade Finance. As an example, you can consider OBTFBCService. This service is exposed by the OBTF Interface Gateway to create BC contract.
- **Operation Name:-** This is the name of the operation that the external system wishes to perform within the service. These operations names are published by Oracle Banking Trade Finance. As an example, you can consider QueryContract, which is for Querying BC contract.
- **External Operation Name:-** This is the specific area of operation that an external system is performing on its side within the broad context of the Oracle Banking Trade Finance's amendment. In an external system, if the BC contract is created, this has a unique name by which it is identified within Oracle Banking Trade Finance.

It is through the 'Gateway Amendment Maintenance' screen that you maintain a set of amendable fields, which can amend in Oracle Banking Trade Finance whenever a request for the same will be send from an external system. Based on this maintenance, the amendment request is addressed by Oracle Banking Trade Finance. Invoke this screen by typing 'GWDAMDMT' in the field at

the top right corner of the Application tool bar and clicking the adjoining arrow button. The screen is as shown below:

Gateway Amendment Maintenance

New Enter Query

External System \*  
Origin System \*  
Source Operation \*  
Service Name  
Operation Code

Amendable Nodes

Node Name *	New Allowed	Delete Allowed	All Records
No data to display.			

Page 1 (0 of 0 items) |< < 1 > >|

Amendable Fields

Field Name *
No data to display.

Page 1 (0 of 0 items) |< < 1 > >|

Audit Exit

#### External System

Select the relevant external system. Based on the maintenance here, only the fields that are

selected as amendable can be modified if a request comes from the chosen external system.

**Note**

The maintenance pertaining to external systems is factory shipped for your bank.

**Origin System**

Specify the origin system for which the amendment details are applicable.

**Source Operation**

The free format text (without spaces) which identifies the amendment you are doing. This is mandatory.

**Note**

Source Operation will be defaulted as (FUNCTIONID)\_MODIFY, if the Source Operation is not sent from an external system and the function ID will be derived from Service and Operation combination.

**Service Name**

The relevant service name, this is a broad level grouping of similar operations within a module in Oracle Banking Trade Finance. The service names are published by Oracle Banking Trade Finance.

**Note**

The maintenance pertaining to service names is factory shipped for your bank

**Operation Code**

The relevant operation code. This is the operation that the external system wishes to perform within the selected service. The operation names are published by Oracle Banking Trade Finance. As an example, you can take 'ModifyContract', which is for modification of a BC contract. Each operation under different service names is identified by a unique code.

**Note**

The maintenance pertaining to operation codes is factory shipped for your bank.

**Amendable Nodes**

**Node Name**

Specify the node name. Alternatively, you can select the node name from the option list. The list displays all valid nodes maintained in the system.

**New Allowed**

Check this box if 'New Allowed' is applicable.

**Delete Allowed**

Check this box if 'Delete Allowed' is applicable.

**All Records**

Check this box if all records are applicable.

## **3.11 Gateway Maintenance for Approot Deployment**

This section contains the following topics:

- [Section 3.11.1, "Maintaining Gateway for Approot Deployment"](#)

### **3.11.1 Maintaining Gateway for Approot Deployment**

The following function IDs are used for the gateway maintenance for Approot deployment.

The function IDs detail and their descriptions are listed below:

<b>Function ID</b>	<b>Summary Function ID</b>	<b>Description</b>
CODAPSRC	COSAPSRC	Upload Source Maintenance
CODAPUPL	COSAPUPL	Upload Source Preferences Maintenance
GWDAPEFN	GWSAPEFN	Gateway Amendment Maintenance
GWDAPAMD	GWSAPAMD	External System Functions
GWDAPESY	GWSAPESY	External System Maintenance

---

## 4. Message Formats

### 4.1 Introduction

Data exchanged between Oracle Banking Trade Finance and the external systems will be in the form of XML messages. These XML messages are defined in OBTF in the form of XML Schema Documents (XSD) and are referred to as 'OBTF formats'. An XML Schema is uniquely identified by its Namespace and its Root Element (root node). The root node for XSDs of gateway messages will be generated as per the following pattern:

- <OPERATION>\_<MESSAGE EXCHANGE PATTERN>\_REQ
- <OPERATION>\_<MESSAGE EXCHANGE PATTERN>\_RES
- Where
- <OPERATION> refers to an Oracle Banking Trade Finance operation like QueryContract.
- <MESSAGE\_EXCHANGE\_PATTERN> refers to the request & response message patterns exchanged. The Message Pattern is of the following types
  - FS (Full Screen) - This indicates that the payload will contain all the elements on the corresponding Oracle Banking Trade Finance screen.
  - IO (Input Only) - This indicates that the payload will contain the only input elements on the corresponding Oracle Banking Trade Finance screen.
  - PK (Primary Key): This indicates that the payload will contain only the Primary Key elements of the corresponding Oracle Banking Trade Finance screen.

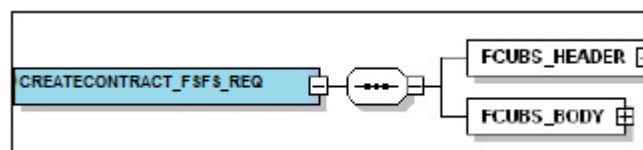
<MESSAGE\_EXCHANGE\_PATTERN> is a combination of message patterns. The <MESSAGE\_EXCHANGE\_PATTERN> available in OBTF is shown below:

- FSFS - Full Screen Request and Full Screen Response
- IOPK - Input Only Request and Primary Key Response
- IOFS - Input Only Request and Full Screen Response

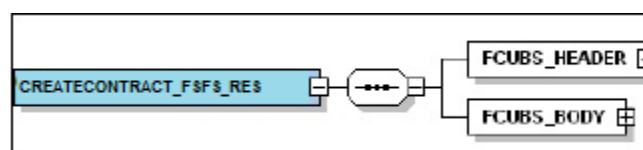
#### **Non-Query Formats**

For example, consider the Operation 'CREATECONTRACT'. The root node of messages under this operation will be as under:

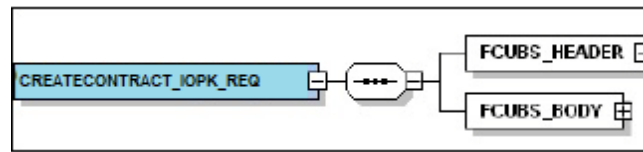
- CREATECONTRACT\_FSFS\_REQ



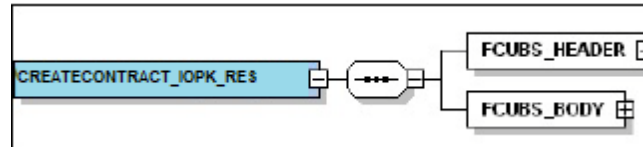
- CREATECONTRACT\_FSFS\_RES



- CREATECONTRACT\_IOPK\_REQ



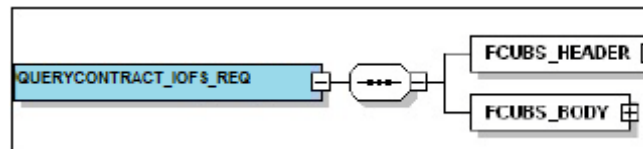
- CREATECONTRACT\_IOPK\_RES



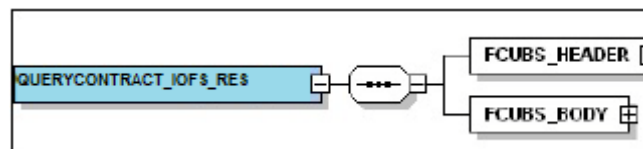
### **Query Formats**

For example, consider the Operation 'QUERYCONTRACT'. The root node of messages under this operation will be as under:

- QUERYCONTRACT\_IOFS\_REQ



- QUERYCONTRACT\_IOFS\_RES



This chapter contains the following sections

- [Section 4.2, "Oracle Banking Trade Finance Envelope"](#)
- [Section 4.3, "Oracle Banking Trade Finance NOTIFICATION"](#):

## **4.2 Oracle Banking Trade Finance Envelope**

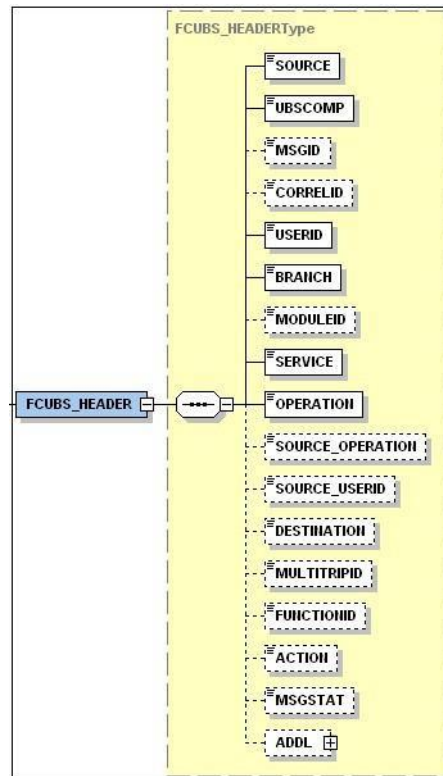
This section contains the following topics:

- [Section 4.2.1, "FCUBS\\_HEADER"](#)
- [Section 4.2.2, "FCUBS\\_BODY"](#)

A standard gateway message in the Oracle Banking Trade Finance Envelope contains two main components namely:

- **FCUBS\_HEADER** – This is a node that forms the header of the OBTF Envelope. The tags under the header portion of a message will identify a partner system, a service, an operation, user, branch, etc. These tags are constant across all messages.

- **FCUBS\_BODY** – This node contains the actual payload of a message. The contents of this node will vary for each message.



#### 4.2.1 **FCUBS\_HEADER**

The tags under FCUBS\_HEADER have been described below:

##### **SOURCE**

This indicates the name of the External system that is the source of the message.

##### **UBSCOMP**

This indicates the Oracle Banking Trade Finance component of the message.

##### **MSGID**

This unique ID identifies each message – incoming or outgoing in Oracle Banking Trade Finance. Every message will have a distinct message ID.

**CORRELID**

This is the id using which any system which has sent a request to OBTF can correlate to the response. In the External system maintenance, the Correlation Pattern can be configured for each external system. It can be maintained that either the 'MSGID' or the 'CORRELID' of the request message is returned back as the 'CORRELID' in the response message.

Depending on this maintenance, Oracle Banking Trade Finance will set either the 'MSGID' or the 'CORRELID' of the request message in the response message.

**USERID**

For request messages, this ID is used to submit message requests. Oracle Banking Trade Finance will process this request using this id.

For response messages, the value of this will be 'null'.

**BRANCH**

This indicates the Oracle Banking Trade Finance Branch Code where the request message needs to be processed. If the BRANCH is missing in the header, request message will be transmitted and processed in Head Office branch.

**MODULEID**

This indicates the module ID.

**SERVICE**

This provides details on the various services of Oracle Banking Trade Finance. For every incoming message in Oracle Banking Trade Finance, the service name is mandatory.

**OPERATION**

This indicates the functional operation.

**SOURCE\_OPERATION**

This indicates the functional operation as registered in Oracle Banking Trade Finance.

**SOURCE\_USERID**

This is the User ID with which the request message was invoked from the SOURCE.

**DESTINATION**

For incoming messages, the destination will be Oracle Banking Trade Finance. For response messages, system will populate the SOURCE of the request message as DESTINATION.

**MULTIRIPID**

This is a unique id which indicates overrides.

**FUNCTIONID**

This indicates the Oracle Banking Trade Finance Function ID

**MSGSTAT**

This indicates whether the transaction is a SUCCESS or FAILURE.

**ADDL**

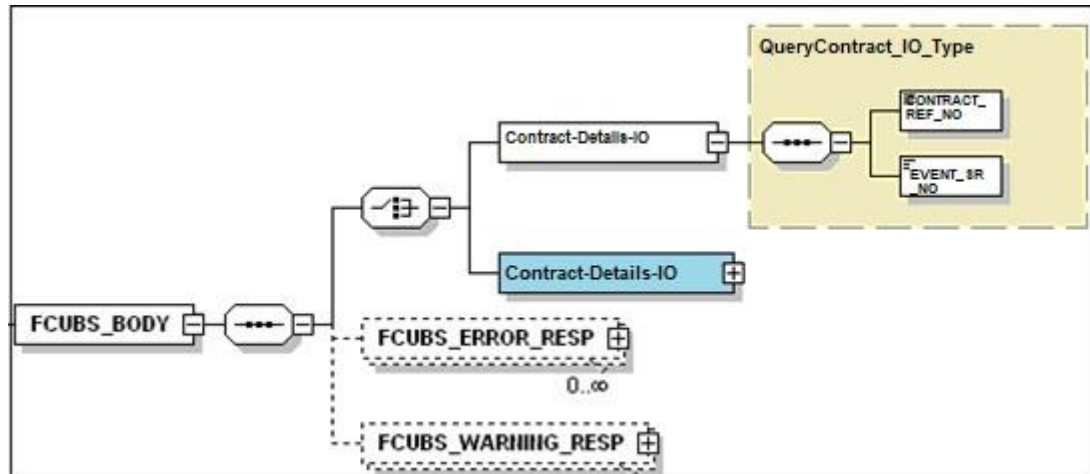
This is used to send additional parameters i.e. parameters not available in Oracle Banking Trade Finance.



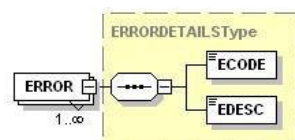
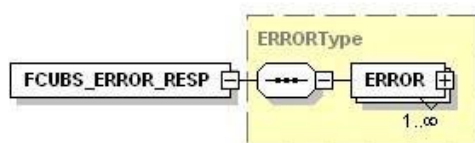
## 4.2.2 FCUBS\_BODY

The FCUBS\_BODY will contain the actual payload to perform the respective transaction. The contents of the payload will vary for each operation.

The following snapshot shows a sample FCUBS\_BODY of QUERYCONTRACT operation.



FCUBS\_BODY will contain additional nodes for error response and warning response. A diagrammatic representation of the Error response is as shown below:



### 4.2.2.1 FCUBS\_ERROR\_RESP

The error response message will be sent from Oracle Banking Trade Finance when errors are raised in a transaction. The error response will have another tag 'ERROR' within it.

#### ERROR

The 'ERROR' node will have tags for error code and error description. The 'ERROR' node will be generated for each error raised by OBTF.

### 4.2.2.2 FCUBS\_WARNING\_RESP

The warning response message will be sent when overrides are raised in a transaction. The Warning response will have another tag 'WARNING' within it.

## WARNING

This node will have tags for warning code and warning description. The 'WARNING' node will be generated for each override raised by OBTF.

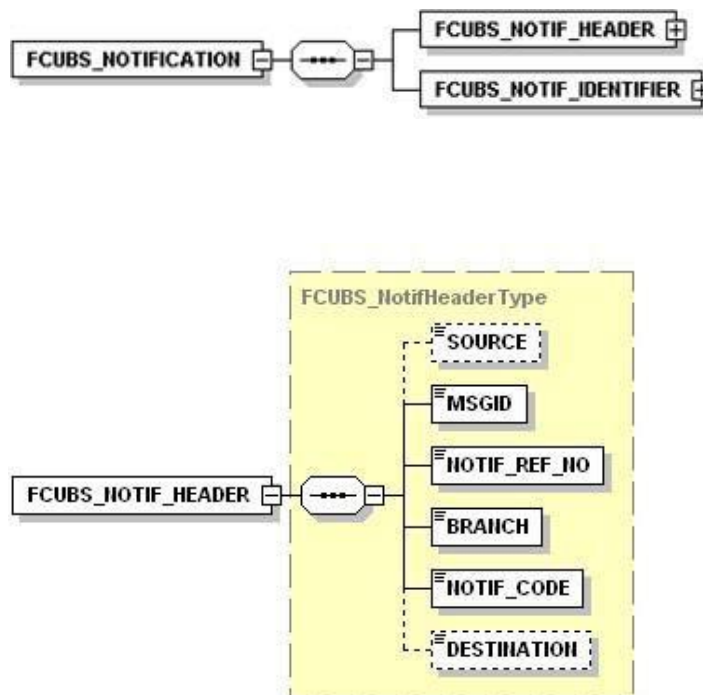
## 4.3 Oracle Banking Trade Finance NOTIFICATION

The notification messages are generated in a standard format. The notification messages will consist of two main components:

OBTF\_NOTIF\_HEADER – This forms the header portion of a notification message. This contains a standard set of tags that can identify a notification. These tags are constant across all notification messages.

OBTF\_NOTIF\_IDENTIFIER – This will identify the maintenance records based on the information provided under this node. The contents of this node will vary for each notification.

A diagrammatic representation of OBTF NOTIFICATION is as shown below:



### 4.3.1 OBTF NOTIFICATION HEADER

The tags under OBTF NOTIFICATION HEADER have been described below:

#### **SOURCE**

This indicates the name of the External system or the source of the message.

#### **MSGID**

This is the unique reference number generated by Oracle Banking Trade Finance.

#### **NOTIF\_REF\_NO.**

This unique reference number identifies each notification message generated in Oracle Banking Trade Finance.

**BRANCH**

This indicates the branch in which notification has been triggered.

**NOTIF\_CODE**

This indicates the code for the notification that has been triggered.

**DESTINATION**

For incoming messages, the DESTINATION should be Oracle Banking Trade Finance. For response messages, system will populate the SOURCE of the request message as DESTINATION.

*Refer 'Service-Documentation' available under 'Gateway' for details about each message.*

---

## 5. Annexure – List of Messages

### 5.1 Introduction

This chapter lists out the various messages that fold under the purview of the following Gateway services:

- Inbound application integration – used when any external system needs to add, modify or query information within Oracle Banking Trade Finance
- Outbound application integration – used when any external system needs to be notified of the various events that occur within Oracle Banking Trade Finance.

### 5.2 Inbound Application Integration – Services

Oracle Banking Trade Finance Inbound Application Gateway provides XML based 'Services'. These services consist of one or more 'Operations'. These operations can be invoked by any external system to perform the relevant operation in Oracle Banking Trade Finance.

This chapter contains the following sections:

- [Section 5.2.1, "OBTFBCService"](#)
- [Section 5.2.2, "OBTFCLService"](#)
- [Section 5.2.3, "OBTFCustomerService"](#)
- [Section 5.2.4, "OBTFIBService"](#)
- [Section 5.2.5, "OBTFCLMaintService"](#)
- [Section 5.2.6, "OBTFILService"](#)
- [Section 5.2.7, "OBTFMLService"](#)
- [Section 5.2.8, "OBTFMSService"](#)

#### 5.2.1 OBTFBCService

All Bills and Collections operations are logically grouped under a service called 'OBTFBCService'.

Oracle Banking Trade Finance supports the following operations under this service:

- QueryContract
- CreateContract
- ModifyContract
- LiquidateContract
- AuthorizeContract
- CloseContract
- DeleteContract
- ReverseContract
- RolloverContract
- CreateBCCContractSim
- ModifyBCCContractSim
- LiquidateBCCContractSim
- QueryBCCContractSim
- AuthorizeBCCContractSim

- CloseBCCContractSim
- DeleteBCCContractSim
- ReverseBCCContractSim
- RolloverBCCContractSim

#### 5.2.1.1 **QueryContract**

This is the operation code for querying the BC contract in Oracle Banking Trade Finance. It involves the following messages:

- QueryContract-Req-IO - This is the request message to query BC contract in Oracle Banking Trade Finance. An external system which intends to query BC contract has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for querying the BC contract.
- QueryContract-Res-Full - This is the response message which is sent back to the external system when a BC contract is successfully queried in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

#### 5.2.1.2 **CreateContract**

This is the operation code for creating new BC contract in Oracle Banking Trade Finance. It involves the following messages:

- CreateContract-Req-Full - This is the request message to create a new BC contract in Oracle Banking Trade Finance. An external system which intends to create new BC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- CreateContract-Req-IO - This is the request message to create a new BC contract in Oracle Banking Trade Finance. An external system which intends to create new BC contract has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for creating the BC contract.
- CreateContract-Res-Full - This is the response message which is sent back to the external system when a BC contract is successfully created in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- CreateContract-Res-PK - This is the response message which is sent back to the external system when a BC contract is successfully created in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the BC contract creation in the Oracle Banking Trade Finance data store.

#### 5.2.1.3 **ModifyContract**

This is the operation code for modifying the BC contract in Oracle Banking Trade Finance. It involves the following messages:

- ModifyContract-Req-Full - This is the request message to modifying the BC contract in Oracle Banking Trade Finance. An external system which intends modify the BC contract has to send the request in the format specified. The format of the message is called 'Full- screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- ModifyContract-Req-IO - This is the request message to modifying the BC contract in Oracle Banking Trade Finance. An external system which intends modify the BC contract has to send the request in the specified format. The format of the message is called 'Input- only', which has only those fields that are actually required for modifying the contract.
- ModifyContract-Res-Full - This is the response message which is sent back to the external system when a BC contract is successfully modified in Oracle Banking Trade

Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

- **ModifyContract-Res-PK-** This is the response message which is sent back to the external system when a BC contract is successfully modified in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the BC contract modification in the Oracle Banking Trade Finance data store.

#### **5.2.1.4 LiquidateContract**

This is the operation code for liquidating the BC contract in Oracle Banking Trade Finance. It involves the following messages:

- **LiquidateContract-Req-Full-** This is the request message to liquidate a BC contract in Oracle Banking Trade Finance. An external system which intends to liquidate a BC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **LiquidateContract-Req-IO-** This is the request message to liquidate BC contract in Oracle Banking Trade Finance. An external system which intends to liquidate BC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for liquidating BC contract.
- **LiquidateContract-Res-Full-** This is the response message which is sent back to the external system when a BC contract is successfully liquidated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **LiquidateContract-Res-PK-** This is the response message which is sent back to the external system when a BC contract is successfully liquidated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the BC contract liquidation in the Oracle Banking Trade Finance data store.

#### **5.2.1.5 AuthorizeContract**

This is the operation code for authorizing the BC contract in Oracle Banking Trade Finance. It involves the following messages:

- **AuthorizeContract-Req-Full-** This is the request message to authorize a BC contract in Oracle Banking Trade Finance. An external system which intends to authorize a BC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **AuthorizeContract-Req-IO-** This is the request message to authorize BC contract in Oracle Banking Trade Finance. An external system which intends to authorize BC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for authorizing BC contract.
- **AuthorizeContract-Res-Full-** This is the response message which is sent back to the external system when a BC contract is successfully authorized in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **AuthorizeContract-Res-PK-** This is the response message which is sent back to the external system when a BC contract is successfully authorized in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to authorize BC contract in the Oracle Banking Trade Finance data store.

#### **5.2.1.6 CloseContract**



This is the operation code for closing the BC contract in Oracle Banking Trade Finance. It

involves the following messages:

- **CloseContract-Req-Full-** This is the request message to close BC contract in Oracle Banking Trade Finance. An external system which intends to close BC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **CloseContract-Req-IO-** This is the request message to close BC contract in Oracle Banking Trade Finance. An external system which intends to close BC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for closing BC contract.
- **CloseContract-Res-Full-** This is the response message which is sent back to the external system when a BC contract is successfully closed in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **CloseContract-Res-PK-** This is the response message which is sent back to the external system when a BC contract is successfully closed in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to close BC contract in the Oracle Banking Trade Finance data store.

#### **5.2.1.7 DeleteContract**

This is the operation code for deleting the BC contract in Oracle Banking Trade Finance. It involves the following messages:

- **DeleteContract-Req-Full-** This is the request message to delete BC contract in Oracle Banking Trade Finance. An external system which intends to delete BC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **DeleteContract-Req-IO-** This is the request message to delete BC contract in Oracle Banking Trade Finance. An external system which intends to delete BC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for deleting BC contract.
- **DeleteContract-Res-Full-** This is the response message which is sent back to the external system when a BC contract is successfully deleted in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **DeleteContract-Res-PK-** This is the response message which is sent back to the external system when a BC contract is successfully deleted in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to delete BC contract in the Oracle Banking Trade Finance data store.

#### **5.2.1.8 ReverseContract**

This is the operation code for reversing the BC contract in Oracle Banking Trade Finance. It involves the following messages:

- **ReverseContract-Req-Full-** This is the request message to reverse BC contract in Oracle Banking Trade Finance. An external system which intends to reverse BC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **ReverseContract-Req-IO-** This is the request message to reverse BC contract in Oracle Banking Trade Finance. An external system which intends to reverse BC

contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for reversing BC contract.

- ReverseContract-Res-Full- This is the response message which is sent back to the external system when a BC contract is successfully reversed in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- ReverseContract-Res-PK- This is the response message which is sent back to the external system when a BC contract is successfully reversed in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to reverse BC contract in the Oracle Banking Trade Finance data store.

#### **5.2.1.9 RolloverContract**

This is the operation code for rollover the BC contract in Oracle Banking Trade Finance. It involves the following messages:

- RolloverContract-Req-Full- This is the request message to rollover BC contract in Oracle Banking Trade Finance. An external system which intends to rollover BC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- RolloverContract-Req-IO- This is the request message to rollover BC contract in Oracle Banking Trade Finance. An external system which intends to rollover BC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for rolling over BC contract.
- RolloverContract-Res-Full- This is the response message which is sent back to the external system when a BC contract is successfully rolled over in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- RolloverContract-Res-PK- This is the response message which is sent back to the external system when a BC contract is successfully rolled over in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to rollover BC contract in the Oracle Banking Trade Finance data store.

#### **5.2.1.10 CreateBCContractSim**

This is the operation code for simulating BC contract in Oracle Banking Trade Finance. It involves the following messages:

- CreateBCContractSim-Req-Full - This is the request message to simulate a BC contract in Oracle Banking Trade Finance. An external system which intends to simulate BC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- CreateBCContractSim-Req-IO - This is the request message to simulate a BC contract in Oracle Banking Trade Finance. An external system which intends to simulate BC contract has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for simulating the BC contract.
- CreateBCContractSim-Res-Full - This is the response message which is sent back to the external system when a BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

- CreateBCContractSim-Res-PK - This is the response message which is sent back to the external system when a BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which

includes only the primary key fields of the BC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.1.11 ModifyBCContractSim**

This is the operation code for simulating the BC contract modification in Oracle Banking Trade Finance. It involves the following messages:

- **ModifyBCContractSim-Req-Full** - This is the request message to simulate the BC contract modification in Oracle Banking Trade Finance. An external system which intends to simulate the BC contract modification has to send the request in the format specified. The format of the message is called 'Full- screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **ModifyBCContractSim-Req-IO** - This is the request message to simulate the BC contract modification in Oracle Banking Trade Finance. An external system which intends to simulate the BC contract modification has to send the request in the specified format. The format of the message is called 'Input- only', which has only those fields that are actually required for simulating the BC contract modification.
- **ModifyBCContractSim-Res-Full** - This is the response message which is sent back to the external system when modification of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **ModifyBCContractSim-Res-PK** - This is the response message which is sent back to the external system when modification of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of simulating the BC contract modification in the Oracle Banking Trade Finance data store.

#### **5.2.1.12 LiquidateBCContractSim**

This is the operation code for simulating the BC contract liquidation in Oracle Banking Trade Finance. It involves the following messages:

- **LiquidateBCContractSim-Req-Full** - This is the request message to simulate the BC contract liquidation in Oracle Banking Trade Finance. An external system which intends to simulate the BC contract liquidation has to send the request in the format specified. The format of the message is called 'Full- screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **LiquidateBCContractSim-Req-IO** - This is the request message to simulate the BC contract liquidation in Oracle Banking Trade Finance. An external system which intends to simulate the BC contract liquidation has to send the request in the specified format. The format of the message is called 'Input- only', which has only those fields that are actually required for simulating the BC contract liquidation.
- **LiquidateBCContractSim-Res-Full** - This is the response message which is sent back to the external system when liquidation of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **LiquidateBCContractSim-Res-PK** - This is the response message which is sent back to the external system when liquidation of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of simulating the BC contract liquidation in the Oracle Banking Trade Finance data store.

#### **5.2.1.13 QueryBCContractSim**

This is the operation code for querying the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- QueryBCContractSim-Req-IO - This is the request message to query BC contract simulation in Oracle Banking Trade Finance. An external system which intends to query BC contract has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for querying the BC contract.
- QueryBCContractSim-Res-Full - This is the response message which is sent back to the external system when a BC contract simulation is successfully queried in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

#### **5.2.1.14 AuthorizeBCContractSim**

This is the operation code for authorizing the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- AuthorizeBCContractSim-Req-Full- This is the request message to authorize a BC contract simulation in Oracle Banking Trade Finance. An external system which intends to authorize a BC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- AuthorizeBCContractSim-Req-IO- This is the request message to authorize BC contract simulation in Oracle Banking Trade Finance. An external system which intends to authorize BC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for authorizing BC contract simulation.
- AuthorizeBCContractSim-Res-Full- This is the response message which is sent back to the external system when authorization of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- AuthorizeBCContractSim-Res-PK- This is the response message which is sent back to the external system when authorization of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to authorize BC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.1.15 CloseBCContractSim**

This is the operation code for closing the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- CloseBCContractSim-Req-Full- This is the request message to close BC contract simulation in Oracle Banking Trade Finance. An external system which intends to close BC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- CloseBCContractSim-Req-IO- This is the request message to close BC contract simulation in Oracle Banking Trade Finance. An external system which intends to close BC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for closing BC contract simulation.
- CloseBCContractSim-Res-Full- This is the response message which is sent back to the external system when closing of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.



- CloseBCContractSim-Res-PK- This is the response message which is sent back to the external system when closing of BC contract is successfully simulated in Oracle

Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to close BC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.1.16 DeleteBCContractSim**

This is the operation code for deleting the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- **DeleteBCContractSim-Req-Full-** This is the request message to delete BC contract simulation in Oracle Banking Trade Finance. An external system which intends to delete BC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **DeleteBCContractSim-Req-IO-** This is the request message to delete BC contract simulation in Oracle Banking Trade Finance. An external system which intends to delete BC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for deleting BC contract simulation.
- **DeleteBCContractSim-Res-Full-** This is the response message which is sent back to the external system when a deletion of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **DeleteBCContractSim-Res-PK-** This is the response message which is sent back to the external system when deletion of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to delete BC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.1.17 ReverseBCContractSim**

This is the operation code for reversing the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- **ReverseBCContractSim-Req-Full-** This is the request message to reverse BC contract simulation in Oracle Banking Trade Finance. An external system which intends to reverse BC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **ReverseBCContractSim-Req-IO-** This is the request message to reverse BC contract simulation in Oracle Banking Trade Finance. An external system which intends to reverse BC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for reversing BC contract simulation.
- **ReverseBCContractSim-Res-Full-** This is the response message which is sent back to the external system when reversing of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **ReverseBCContractSim-Res-PK-** This is the response message which is sent back to the external system when reversing of BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to reverse BC contract simulation in the Oracle Banking Trade Finance data store.

#### 5.2.1.18 RolloverBCContractSim

This is the operation code for rollover the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- RolloverBCContractSim-Req-Full- This is the request message to rollover BC contract simulation in Oracle Banking Trade Finance. An external system which intends to rollover BC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- RolloverBCContractSim-Req-IO- This is the request message to rollover BC contract simulation in Oracle Banking Trade Finance. An external system which intends to rollover BC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for rolling over BC contract simulation.
- RolloverBCContractSim-Res-Full- This is the response message which is sent back to the external system when rollover of a BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- RolloverBCContractSim-Res-PK- This is the response message which is sent back to the external system when rollover of a BC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to rollover BC contract simulation in the Oracle Banking Trade Finance data store.

## 5.2.2 **OBTFCLService**

All Letter of Credit operations are logically grouped under a service called 'OBTFCLService'.

Oracle Banking Trade Finance supports the following operations under this service:

- QueryContract
- CreateContract
- ModifyContract
- ReopenContract
- AuthorizeContract
- CloseContract
- DeleteContract
- ReverseContract
- RolloverContract
- CreateContractSim
- ModifyContractSim
- ReopenContractSim
- QueryContractSim
- AuthorizeContractSim
- CloseContractSim
- DeleteContractSim
- ReverseContractSim
- RolloverContractSim
- AuthorizeAmendConfirm
- CloseAmendConfirm
- ConfirmAmendConfirm

- DeleteAmendConfirm
- QueryAmendConfirm

- ModifyAmendConfirm
- CreateAmendConfirm
- QueryGuarantee
- CreateGuarantee
- ModifyGuarantee
- ReopenGuarantee
- AuthorizeGuarantee
- CloseGuarantee
- DeleteGuarantee
- ReverseGuarantee

### 5.2.2.1 **QueryContract**

This is the operation code for querying the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- QueryContract-Req-IO - This is the request message to query LC contract in Oracle Banking Trade Finance. An external system which intends to query LC contract has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for querying the LC contract.
- QueryContract-Res-Full - This is the response message which is sent back to the external system when a LC contract is successfully queried in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

### 5.2.2.2 **CreateContract**

This is the operation code for creating new LC contract in Oracle Banking Trade Finance. It involves the following messages:

- CreateContract-Req-Full - This is the request message to create a new LC contract in Oracle Banking Trade Finance. An external system which intends to create new LC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- CreateContract-Req-IO - This is the request message to create a new LC contract in Oracle Banking Trade Finance. An external system which intends to create new LC contract has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for creating the LC contract.
- CreateContract-Res-Full - This is the response message which is sent back to the external system when a LC contract is successfully created in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- CreateContract-Res-PK - This is the response message which is sent back to the external system when a LC contract is successfully created in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the LC contract creation in the Oracle Banking Trade Finance data store.

### 5.2.2.3 **ModifyContract**

This is the operation code for modifying the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- **ModifyContract-Req-Full** - This is the request message to modifying the LC contract in Oracle Banking Trade Finance. An external system which intends modify the LC contract has to send the request in the format specified. The format of the message is called 'Full- screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **ModifyContract-Req-IO** - This is the request message to modifying the LC contract in Oracle Banking Trade Finance. An external system which intends modify the LC contract has to send the request in the specified format. The format of the message is called 'Input- only', which has only those fields that are actually required for modifying the contract.
- **ModifyContract-Res-Full** - This is the response message which is sent back to the external system when a LC contract is successfully modified in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **ModifyContract-Res-PK**- This is the response message which is sent back to the external system when a LC contract is successfully modified in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the LC contract modification in the Oracle Banking Trade Finance data store

#### **5.2.2.4 ReopenContract**

This is the operation code for reopening the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- **ReopenContract-Req-Full**- This is the request message to reopen a LC contract in Oracle Banking Trade Finance. An external system which intends to reopen a LC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **ReopenContract-Req-IO**- This is the request message to reopen LC contract in Oracle Banking Trade Finance. An external system which intends to reopen LC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for reopening LC contract.
- **ReopenContract-Res-Full**- This is the response message which is sent back to the external system when a LC contract is successfully reopened in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **ReopenContract-Res-PK**- This is the response message which is sent back to the external system when a LC contract is successfully reopened in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the LC contract reopen in the Oracle Banking Trade Finance data store.

#### **5.2.2.5 AuthorizeContract**

This is the operation code for authorizing the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- **AuthorizeContract-Req-Full**- This is the request message to authorize a LC contract in Oracle Banking Trade Finance. An external system which intends to authorize a LC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **AuthorizeContract-Req-IO**- This is the request message to authorize LC contract in



Oracle Banking Trade Finance. An external system which intends to authorize LC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for

authorizing LC contract.

- AuthorizeContract-Res-Full- This is the response message which is sent back to the external system when a LC contract is successfully authorized in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- AuthorizeContract-Res-PK- This is the response message which is sent back to the external system when a LC contract is successfully authorized in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to authorize LC contract in the Oracle Banking Trade Finance data store.

#### **5.2.2.6 CloseContract**

This is the operation code for closing the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- CloseContract-Req-Full- This is the request message to close LC contract in Oracle Banking Trade Finance. An external system which intends to close LC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- CloseContract-Req-IO- This is the request message to close LC contract in Oracle Banking Trade Finance. An external system which intends to close LC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for closing LC contract.
- CloseContract-Res-Full- This is the response message which is sent back to the external system when a LC contract is successfully closed in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- CloseContract-Res-PK- This is the response message which is sent back to the external system when a LC contract is successfully closed in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to close LC contract in the Oracle Banking Trade Finance data store.

#### **5.2.2.7 DeleteContract**

This is the operation code for deleting the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- DeleteContract-Req-Full- This is the request message to delete LC contract in Oracle Banking Trade Finance. An external system which intends to delete LC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- DeleteContract-Req-IO- This is the request message to delete LC contract in Oracle Banking Trade Finance. An external system which intends to delete LC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for deleting LC contract.
- DeleteContract-Res-Full- This is the response message which is sent back to the external system when a LC contract is successfully deleted in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- DeleteContract-Res-PK- This is the response message which is sent back to the external system when a LC contract is successfully deleted in Oracle Banking Trade

Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to delete LC contract in the Oracle Banking Trade Finance data store.

#### 5.2.2.8 **ReverseContract**

This is the operation code for reversing the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- ReverseContract-Req-Full- This is the request message to reverse LC contract in Oracle Banking Trade Finance. An external system which intends to reverse LC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- ReverseContract-Req-IO- This is the request message to reverse LC contract in Oracle Banking Trade Finance. An external system which intends to reverse LC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for reversing LC contract.
- ReverseContract-Res-Full- This is the response message which is sent back to the external system when a LC contract is successfully reversed in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- ReverseContract-Res-PK- This is the response message which is sent back to the external system when a LC contract is successfully reversed in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to reverse LC contract in the Oracle Banking Trade Finance data store.

#### 5.2.2.9 **RolloverContract**

This is the operation code for rollover the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- RolloverContract-Req-Full- This is the request message to rollover LC contract in Oracle Banking Trade Finance. An external system which intends to rollover LC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- RolloverContract-Req-IO- This is the request message to rollover LC contract in Oracle Banking Trade Finance. An external system which intends to rollover LC contract has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for rolling over LC contract.
- RolloverContract-Res-Full- This is the response message which is sent back to the external system when a LC contract is successfully rolled over in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- RolloverContract-Res-PK- This is the response message which is sent back to the external system when a LC contract is successfully rolled over in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to rollover LC contract in the Oracle Banking Trade Finance data store.

#### 5.2.2.10 **CreateContractSim**

This is the operation code for simulating LC contract in Oracle Banking Trade Finance. It involves the following messages:

- CreateContractSim-Req-Full - This is the request message to simulate a LC contract

in Oracle Banking Trade Finance. An external system which intends to simulate LC contract has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields)

as in the corresponding Oracle Banking Trade Finance front-end screen.

- **CreateContractSim-Req-IO** - This is the request message to simulate a LC contract in Oracle Banking Trade Finance. An external system which intends to simulate LC contract has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for simulating the LC contract.
- **CreateContractSim-Res-Full** - This is the response message which is sent back to the external system when a LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **CreateContractSim-Res-PK** - This is the response message which is sent back to the external system when a LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the LC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.2.11 ModifyContractSim**

This is the operation code for simulating the LC contract modification in Oracle Banking Trade Finance. It involves the following messages:

- **ModifyContractSim-Req-Full** - This is the request message to simulate the LC contract modification in Oracle Banking Trade Finance. An external system which intends to simulate the LC contract modification has to send the request in the format specified. The format of the message is called 'Full- screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **ModifyContractSim-Req-IO** - This is the request message to simulate the LC contract modification in Oracle Banking Trade Finance. An external system which intends to simulate the LC contract modification has to send the request in the specified format. The format of the message is called 'Input- only', which has only those fields that are actually required for simulating the LC contract modification.
- **ModifyContractSim-Res-Full** - This is the response message which is sent back to the external system when modification of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **ModifyContractSim-Res-PK** - This is the response message which is sent back to the external system when modification of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of simulating the LC contract modification in the Oracle Banking Trade Finance data store.

#### **5.2.2.12 ReopenContractSim**

This is the operation code for simulating the LC contract reopen in Oracle Banking Trade Finance. It involves the following messages:

- **ReopenContractSim-Req-Full** - This is the request message to simulate the LC contract reopen in Oracle Banking Trade Finance. An external system which intends to simulate the LC contract reopen has to send the request in the format specified. The format of the message is called 'Full- screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **ReopenContractSim-Req-IO** - This is the request message to simulate the LC contract reopen in Oracle Banking Trade Finance. An external system which intends to simulate the LC contract reopen has to send the request in the specified format. The format of the message is called 'Input- only', which has only those fields that are actually

required for simulating the LC contract reopen.

- ReopenContractSim-Res-Full- This is the response message which is sent back to the external system when reopening of LC contract is successfully simulated in

Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

- ReopenContractSim-Res-PK- This is the response message which is sent back to the external system when reopening of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of simulating the LC contract reopen in the Oracle Banking Trade Finance data store.

#### **5.2.2.13 QueryContractSim**

This is the operation code for querying the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- QueryContractSim-Req-IO - This is the request message to query LC contract simulation in Oracle Banking Trade Finance. An external system which intends to query LC contract has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for querying the LC contract.
- QueryContractSim-Res-Full - This is the response message which is sent back to the external system when a LC contract simulation is successfully queried in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

#### **5.2.2.14 AuthorizeContractSim**

This is the operation code for authorizing the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- AuthorizeContractSim-Req-Full- This is the request message to authorize a LC contract simulation in Oracle Banking Trade Finance. An external system which intends to authorize a LC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- AuthorizeContractSim-Req-IO- This is the request message to authorize LC contract simulation in Oracle Banking Trade Finance. An external system which intends to authorize LC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for authorizing LC contract simulation.
- AuthorizeContractSim-Res-Full- This is the response message which is sent back to the external system when authorization of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- AuthorizeContractSim-Res-PK- This is the response message which is sent back to the external system when authorization of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to authorize LC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.2.15 CloseContractSim**



This is the operation code for closing the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- **CloseContractSim-Req-Full-** This is the request message to close LC contract simulation in Oracle Banking Trade Finance. An external system which intends to close LC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **CloseContractSim-Req-IO-** This is the request message to close LC contract simulation in Oracle Banking Trade Finance. An external system which intends to close LC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for closing LC contract simulation.
- **CloseContractSim-Res-Full-** This is the response message which is sent back to the external system when closing of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **CloseContractSim-Res-PK-** This is the response message which is sent back to the external system when closing of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to close LC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.2.16 DeleteContractSim**

This is the operation code for deleting the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- **DeleteContractSim-Req-Full-** This is the request message to delete LC contract simulation in Oracle Banking Trade Finance. An external system which intends to delete LC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **DeleteContractSim-Req-IO-** This is the request message to delete LC contract simulation in Oracle Banking Trade Finance. An external system which intends to delete LC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for deleting LC contract simulation.
- **DeleteContractSim-Res-Full-** This is the response message which is sent back to the external system when a deletion of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **DeleteContractSim-Res-PK-** This is the response message which is sent back to the external system when deletion of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to delete LC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.2.17 ReverseContractSim**

This is the operation code for reversing the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- **ReverseContractSim-Req-Full-** This is the request message to reverse LC contract simulation in Oracle Banking Trade Finance. An external system which intends to reverse LC contract simulation has to send the request in the format specified. The

format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.

- ReverseContractSim-Req-IO- This is the request message to reverse LC contract simulation in Oracle Banking Trade Finance. An external system which intends to reverse LC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for reversing LC contract simulation.
- ReverseContractSim-Res-Full- This is the response message which is sent back to the external system when reversing of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- ReverseContractSim-Res-PK- This is the response message which is sent back to the external system when reversing of LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to reverse LC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.2.18 RolloverContractSim**

This is the operation code for rollover the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

- RolloverContractSim-Req-Full- This is the request message to rollover LC contract simulation in Oracle Banking Trade Finance. An external system which intends to rollover LC contract simulation has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- RolloverContractSim-Req-IO- This is the request message to rollover LC contract simulation in Oracle Banking Trade Finance. An external system which intends to rollover LC contract simulation has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for rolling over LC contract simulation.
- RolloverContractSim-Res-Full- This is the response message which is sent back to the external system when rollover of a LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- RolloverContractSim-Res-PK- This is the response message which is sent back to the external system when rollover of a LC contract is successfully simulated in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to rollover LC contract simulation in the Oracle Banking Trade Finance data store.

#### **5.2.2.19 AuthorizeAmendConfirm**

This is the operation code for authorizing the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:

- AuthorizeAmendConfirm-Req-Full- This is the request message to authorize a LC contract amendment in Oracle Banking Trade Finance. An external system which intends to authorize a LC contract amendment has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- AuthorizeAmendConfirm-Req-IO- This is the request message to authorize LC contract amendment in Oracle Banking Trade Finance. An external system which intends to authorize LC contract amendment has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields

that are actually required for authorizing LC contract amendment.

- AuthorizeAmendConfirm-Res-Full- This is the response message which is sent back

to the external system when amendment of LC contract is successfully authorized in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

- AuthorizeAmendConfirm-Res-PK- This is the response message which is sent back to the external system when amendment of LC contract is successfully authorized in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to authorize LC contract amendment in the Oracle Banking Trade Finance data store.

#### **5.2.2.20 CloseAmendConfirm**

This is the operation code for closing the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:

- CloseAmendConfirm-Req-Full- This is the request message to close LC contract amendment in Oracle Banking Trade Finance. An external system which intends to close LC contract amendment has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- CloseAmendConfirm-Req-IO- This is the request message to close LC contract amendment in Oracle Banking Trade Finance. An external system which intends to close LC contract amendment has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for closing LC contract amendment.
- CloseAmendConfirm-Res-Full- This is the response message which is sent back to the external system when amendment of LC contract is successfully closed in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- CloseAmendConfirm-Res-PK- This is the response message which is sent back to the external system when amendment of LC contract is successfully closed in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to close LC contract amendment in the Oracle Banking Trade Finance data store.

#### **5.2.2.21 ConfirmAmendConfirm**

This is the operation code for confirming the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:

- ConfirmAmendConfirm-Req-Full- This is the request message to confirm LC contract amendment in Oracle Banking Trade Finance. An external system which intends to confirm LC contract amendment has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- ConfirmAmendConfirm-Req-IO- This is the request message to confirm LC contract amendment in Oracle Banking Trade Finance. An external system which intends to confirm LC contract amendment has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for confirming LC contract amendment.
- ConfirmAmendConfirm-Res-Full- This is the response message which is sent back to the external system when amendment of LC contract is successfully confirmed in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

- ConfirmAmendConfirm-Res-PK- This is the response message which is sent back to the external system when amendment of LC contract is successfully confirmed in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response',

which includes only the primary key fields to confirm LC contract amendment in the Oracle Banking Trade Finance data store.

#### **5.2.2.22 DeleteAmendConfirm**

This is the operation code for deleting the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:

- DeleteAmendConfirm-Req-Full- This is the request message to delete LC contract amendment in Oracle Banking Trade Finance. An external system which intends to delete LC contract amendment has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- DeleteAmendConfirm-Req-IO- This is the request message to delete LC contract amendment in Oracle Banking Trade Finance. An external system which intends to delete LC contract amendment has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for deleting LC contract amendment.
- DeleteAmendConfirm-Res-Full- This is the response message which is sent back to the external system when amendment of LC contract is successfully deleted in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- DeleteAmendConfirm-Res-PK- This is the response message which is sent back to the external system when amendment of LC contract is successfully deleted in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to delete LC contract amendment in the Oracle Banking Trade Finance data store.

#### **5.2.2.23 QueryAmendConfirm**

This is the operation code for querying the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:

- QueryAmendConfirm-Req-IO- This is the request message to query LC contract amendment in Oracle Banking Trade Finance. An external system which intends to query LC contract amendment has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for querying LC contract amendment.
- QueryAmendConfirm-Res-Full- This is the response message which is sent back to the external system when amendment of LC contract is successfully queried in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

#### **5.2.2.24 ModifyAmendConfirm**

This is the operation code for modifying the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:

- ModifyAmendConfirm-Req-Full- This is the request message to modify LC contract amendment in Oracle Banking Trade Finance. An external system which intends to modify LC contract amendment has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- ModifyAmendConfirm-Req-IO- This is the request message to modify LC contract amendment in Oracle Banking Trade Finance. An external system which intends to modify LC contract amendment has to send the request in the specified format. The



format of the message is called 'Input-only', which has only those fields that are actually required for modifying LC contract amendment.

- **ModifyAmendConfirm-Res-Full-** This is the response message which is sent back to the external system when amendment of LC contract is successfully modified in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **ModifyAmendConfirm-Res-PK-** This is the response message which is sent back to the external system when amendment of LC contract is successfully modified in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to modify LC contract amendment in the Oracle Banking Trade Finance data store.

#### **5.2.2.25 CreateAmendConfirm**

This is the operation code for creating the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:

- **CreateAmendConfirm-Req-Full-** This is the request message to create LC contract amendment in Oracle Banking Trade Finance. An external system which intends to create LC contract amendment has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **CreateAmendConfirm-Req-IO-** This is the request message to create LC contract amendment in Oracle Banking Trade Finance. An external system which intends to create LC contract amendment has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for creating LC contract amendment.
- **CreateAmendConfirm-Res-Full-** This is the response message which is sent back to the external system when amendment of LC contract is successfully created in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **CreateAmendConfirm-Res-PK-** This is the response message which is sent back to the external system when amendment of LC contract is successfully created in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to create LC contract amendment in the Oracle Banking Trade Finance data store.

#### **5.2.2.26 QueryGuarantee**

This is the operation code for querying the guarantee in Oracle Banking Trade Finance. It involves the following messages:

- **QueryGuarantee-Req-IO -** This is the request message to query guarantee in Oracle Banking Trade Finance. An external system which intends to query guarantee has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for querying the guarantee.
- **QueryGuarantee-Res-Full -** This is the response message which is sent back to the external system when a guarantee is successfully queried in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

#### **5.2.2.27 CreateGuarantee**

This is the operation code for creating new guarantee in Oracle Banking Trade Finance. It involves the following messages:

- **CreateGuarantee-Req-Full** - This is the request message to create a new guarantee in Oracle Banking Trade Finance. An external system which intends to create new guarantee has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **CreateGuarantee-Req-IO** - This is the request message to create a new guarantee in Oracle Banking Trade Finance. An external system which intends to create new guarantee has to send the request in the format specified. The format of the message is called 'Input-only', which has only those fields that are actually required for creating the guarantee.
- **CreateGuarantee-Res-Full** - This is the response message which is sent back to the external system when a guarantee is successfully created in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **CreateGuarantee-Res-PK** - This is the response message which is sent back to the external system when a guarantee is successfully created in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of guarantee creation in the Oracle Banking Trade Finance data store.

#### **5.2.2.28 ModifyGuarantee**

This is the operation code for modifying the guarantee in Oracle Banking Trade Finance. It involves the following messages:

- **ModifyGuarantee-Req-Full** - This is the request message to modifying the guarantee in Oracle Banking Trade Finance. An external system which intends modify the guarantee has to send the request in the format specified. The format of the message is called 'Full- screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- **ModifyGuarantee-Req-IO** - This is the request message to modifying the guarantee in Oracle Banking Trade Finance. An external system which intends modify the guarantee has to send the request in the specified format. The format of the message is called 'Input- only', which has only those fields that are actually required for modifying the guarantee.
- **ModifyGuarantee-Res-Full** - This is the response message which is sent back to the external system when a guarantee is successfully modified in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- **ModifyGuarantee-Res-PK**- This is the response message which is sent back to the external system when a guarantee is successfully modified in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the guarantee modification in the Oracle Banking Trade Finance data store.

#### **5.2.2.29 ReopenGuarantee**

This is the operation code for reopening the guarantee in Oracle Banking Trade Finance. It involves the following messages:

- **ReopenGuarantee-Req-Full**- This is the request message to reopen a guarantee in Oracle Banking Trade Finance. An external system which intends to reopen a guarantee has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.

- ReopenGuarantee-Req-IO- This is the request message to reopen guarantee in Oracle Banking Trade Finance. An external system which intends to reopen guarantee has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for reopening guarantee.

ReopenGuarantee-Res-Full- This is the response message which is sent back to the external system when a guarantee is successfully reopened in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

- ReopenGuarantee-Res-PK- This is the response message which is sent back to the external system when a guarantee is successfully reopened in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields of the guarantee reopen in the Oracle Banking Trade Finance data store.

### 5.2.2.30 **AuthorizeGuarantee**

This is the operation code for authorizing the guarantee in Oracle Banking Trade Finance. It involves the following messages:

- AuthorizeGuarantee-Req-Full- This is the request message to authorize a guarantee in Oracle Banking Trade Finance. An external system which intends to authorize a guarantee has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- AuthorizeGuarantee-Req-IO- This is the request message to authorize guarantee in Oracle Banking Trade Finance. An external system which intends to authorize guarantee has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for authorizing guarantee.
- AuthorizeGuarantee-Res-Full- This is the response message which is sent back to the external system when a guarantee is successfully authorized in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- AuthorizeGuarantee-Res-PK- This is the response message which is sent back to the external system when a guarantee is successfully authorized in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to authorize guarantee in the Oracle Banking Trade Finance data store.

### 5.2.2.31 **CloseGuarantee**

This is the operation code for closing the guarantee in Oracle Banking Trade Finance. It involves the following messages:

- CloseGuarantee-Req-Full- This is the request message to close guarantee in Oracle Banking Trade Finance. An external system which intends to close guarantee has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- CloseGuarantee-Req-IO- This is the request message to close guarantee in Oracle Banking Trade Finance. An external system which intends to close guarantee to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for closing guarantee.
- CloseGuarantee-Res-Full- This is the response message which is sent back to the

external system when guarantee is successfully closed in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.

- CloseGuarantee-Res-PK- This is the response message which is sent back to the external system when a guarantee is successfully closed in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to close guarantee in the Oracle Banking Trade Finance data store.

#### **5.2.2.32 DeleteGuarantee**

This is the operation code for deleting the guarantee in Oracle Banking Trade Finance. It involves the following messages:

- DeleteGuarantee-Req-Full- This is the request message to delete guarantee in Oracle Banking Trade Finance. An external system which intends to delete guarantee has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- DeleteGuarantee-Req-IO- This is the request message to delete guarantee in Oracle Banking Trade Finance. An external system which intends to delete guarantee has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for deleting guarantee.
- DeleteGuarantee-Res-Full- This is the response message which is sent back to the external system when a guarantee is successfully deleted in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- DeleteGuarantee-Res-PK- This is the response message which is sent back to the external system when a guarantee is successfully deleted in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes only the primary key fields to delete guarantee in the Oracle Banking Trade Finance data store.

#### **5.2.2.33 ReverseGuarantee**

This is the operation code for reversing the guarantee in Oracle Banking Trade Finance. It involves the following messages:

- ReverseGuarantee-Req-Full- This is the request message to reverse guarantee in Oracle Banking Trade Finance. An external system which intends to reverse guarantee has to send the request in the format specified. The format of the message is called 'Full-screen', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance front-end screen.
- ReverseGuarantee-Req-IO- This is the request message to reverse guarantee in Oracle Banking Trade Finance. An external system which intends to reverse guarantee has to send the request in the specified format. The format of the message is called 'Input-only', which has only those fields that are actually required for reversing guarantee.
- ReverseGuarantee-Res-Full- This is the response message which is sent back to the external system when a guarantee is successfully reversed in Oracle Banking Trade Finance. The format of the message is called 'Full-screen response', which includes all the fields (input fields and the display fields) as in the corresponding Oracle Banking Trade Finance screen.
- ReverseGuarantee-Res-PK- This is the response message which is sent back to the external system when a guarantee is successfully reversed in Oracle Banking Trade Finance. The format of the message is called 'Primary-Key response', which includes

only the primary key fields to reverse guarantee in the Oracle Banking Trade Finance data store.

---

## 6. Function ID Glossary

### C

CODSORCE .....3-7  
CODUPLDM .....3-8

### G

GWDAMDMT ..... 3-25  
GWDEXFUN .....3-5  
GWDINBRW ..... 3-11  
GWDNTFEN ..... 3-22  
GWDNTFIN ..... 3-20

GWDOTBRW ..... 3-17  
GWSEXFUN ..... 3-6  
GWSEXSYS ..... 3-3  
GWSINBRW ..... 3-13  
GWSNTFEN ..... 3-23  
GWSNTFIN ..... 3-21  
GWSOTBRW ..... 3-19  
GWSPBROW ..... 3-14, 3-16

### S

STDGWI

