

Oracle Banking Trade Finance Gateway Services User Guide



Release 14.8.2.0.0

G51004-01

April 2026

ORACLE®

Copyright © 2007, 2026, Oracle and/or its affiliates.

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

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

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

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

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

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

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Contents

1 Preface

1.1	Purpose	1
1.2	Audience	1
1.3	Documentation Accessibility	2
1.4	Critical Patches	2
1.5	Diversity and Inclusion	2
1.6	Organization	2
1.7	Conventions	2
1.8	Related Documents	3
1.9	Screenshot Disclaimer	3
1.10	Acronyms and Abbreviations	3
1.11	Symbols and Icons	3

2 Gateway Functions - An Overview

2.1	Introduction	1
2.2	Inbound Application Integration	2
2.2.1	EJB Based Synchronous Deployment Pattern	2
2.2.2	Web Services Based Synchronous Deployment Pattern	3
2.2.3	HTTP Servlet Based Synchronous Deployment Pattern	3
2.2.4	MDB Based Asynchronous Deployment Pattern	4
2.3	Outbound Application Integration	4
2.4	Responsibilities of Integration Gateway	4
2.5	Deployment of Oracle Banking Trade Finance Integration Gate- way	4
2.6	Deployment Patterns for Application Integration	5

3 External System Maintenance

3.1	Introduction	1
3.2	External System	1
3.2.1	Define an External System	2
3.2.2	Specifying External System Details	2
3.2.3	Indicating Correlation Pattern	3
3.2.4	Specifying Message Exchange Pattern	3

3.2.5	Specifying Queue Details	3
3.2.6	Specifying External System Queue Details	4
3.2.7	View External System Details	4
3.2.8	Accessing Services and Operations	5
3.3	Access Rights to an External System	6
3.3.1	Define Access Rights to an External System	6
3.3.2	View External System Function Details	7
3.3.3	Defining Access Rights to a User ID	8
3.3.4	Maintain Upload Source Details	8
3.3.5	Specify Upload Source Preferences	9
3.3.6	Maintain Function ID Preferences	10
3.4	Gateway Password Generation Logic For External System Authentication	11
3.4.1	Password and Message ID as input	12
3.5	Incoming Message Browser	12
3.5.1	View Incoming Message Details	14
3.6	Duplicate Message Browser	14
3.6.1	Query Duplicate Message Browser	16
3.7	Outgoing Message Browser	17
3.7.1	Query Outgoing Message Browser	19
3.8	Define Notification Messages	19
3.8.1	View Installed Notification Details	20
3.9	Define the Notifications Enroutes	21
3.9.1	View Summary	22
3.10	Amendment Maintenance	23
3.10.1	Maintain Gateway Amendment Details	23
3.11	Gateway Maintenance for Approot Deployment	25
3.11.1	Maintaining Gateway for Approot Deployment	26

4 Message Formats

4.1	Introduction	1
4.2	Oracle Banking Trade Finance Envelope	2
4.2.1	FCUBS_HEADER	3
4.2.2	FCUBS_BODY	5
4.2.2.1	FCUBS_ERROR_RESP	5
4.2.2.2	FCUBS_WARNING_RESP	6
4.3	Oracle Banking Trade Finance NOTIFICATION	6
4.3.1	OBTF NOTIFICATION HEADER	6

5 Annexure - List of Messages

5.1	Introduction	1
-----	--------------	---

5.2	Inbound Application Integration – Services	1
5.2.1	OBTFBCService	1
5.2.1.1	QueryContract	3
5.2.1.2	CreateContract	3
5.2.1.3	ModifyContract	3
5.2.1.4	LiquidateContract	4
5.2.1.5	AuthorizeContract	4
5.2.1.6	CloseContract	5
5.2.1.7	DeleteContract	5
5.2.1.8	ReverseContract	6
5.2.1.9	RolloverContract	6
5.2.1.10	CreateBCCContractSim	7
5.2.1.11	ModifyBCCContractSim	7
5.2.1.12	LiquidateBCCContractSim	8
5.2.1.13	QueryBCCContractSim	8
5.2.1.14	AuthorizeBCCContractSim	8
5.2.1.15	CloseBCCContractSim	9
5.2.1.16	DeleteBCCContractSim	9
5.2.1.17	ReverseBCCContractSim	10
5.2.1.18	RolloverBCCContractSim	10
5.2.2	OBTFCLService	11
5.2.2.1	QueryContract	13
5.2.2.2	CreateContract	13
5.2.2.3	ModifyContract	14
5.2.2.4	ReopenContract	14
5.2.2.5	AuthorizeContract	15
5.2.2.6	CloseContract	15
5.2.2.7	DeleteContract	16
5.2.2.8	ReverseContract	16
5.2.2.9	RolloverContract	17
5.2.2.10	CreateContractSim	17
5.2.2.11	ModifyContractSim	18
5.2.2.12	ReopenContractSim	18
5.2.2.13	QueryContractSim	19
5.2.2.14	AuthorizeContractSim	19
5.2.2.15	CloseContractSim	19
5.2.2.16	DeleteContractSim	20
5.2.2.17	ReverseContractSim	20
5.2.2.18	RolloverContractSim	21
5.2.2.19	AuthorizeAmendConfirm	21
5.2.2.20	CloseAmendConfirm	22
5.2.2.21	ConfirmAmendConfirm	22

5.2.2.22	DeleteAmendConfirm	23
5.2.2.23	QueryAmendConfirm	23
5.2.2.24	ModifyAmendConfirm	24
5.2.2.25	CreateAmendConfirm	24
5.2.2.26	QueryGuarantee	25
5.2.2.27	CreateGuarantee	25
5.2.2.28	ModifyGuarantee	25
5.2.2.29	ReopenGuarantee	26
5.2.2.30	AuthorizeGuarantee	26
5.2.2.31	CloseGuarantee	27
5.2.2.32	DeleteGuarantee	27
5.2.2.33	ReverseGuarantee	28

Glossary

1

Preface

- [Purpose](#)
This manual is designed to help you quickly get acquainted with the Integration Gateway of Oracle Banking Trade Finance.
- [Audience](#)
This manual is intended for the following User/User Roles:
- [Documentation Accessibility](#)
- [Critical Patches](#)
- [Diversity and Inclusion](#)
- [Organization](#)
The manual is organized in the following manner:
- [Conventions](#)
- [Related Documents](#)
- [Screenshot Disclaimer](#)
- [Acronyms and Abbreviations](#)
- [Symbols and Icons](#)

1.1 Purpose

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>.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

1.4 Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at [Critical Patches, Security Alerts and Bulletins](#). All critical patches should be applied in a timely manner to ensure effective security, as strongly recommended by [Oracle Software Security Assurance](#).

1.5 Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1.6 Organization

The manual is organized in the following manner:

Topics	Description
Preface	This topic gives information on the intended audience. It also lists the various chapters covered in this User Manual.
Gateway Functions - An Overview	This topic explains the features of this module.
External System Maintenance	This topic explains the maintenance of external systems and the various processes involved.
Message Formats	This topic explains the different message formats.
Annexure – List of Messages	This topic lists all the messages that you encounter while working with Oracle Banking Trade Finance Gateway services.
Function ID Glossary	This topic has alphabetical listing of Function/Screen ID's used in the module with page references for quick navigation.

1.7 Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1.8 Related Documents

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

1.9 Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

1.10 Acronyms and Abbreviations

Abbreviation	Description
OBTF	Oracle Banking Trade Finance
OBTFCS	Oracle Banking Trade Finance Cloud Service

1.11 Symbols and Icons

The list of symbols and icons available on the screens are as follows:

Table 1-1 Symbols and Icons - Common

Symbol/Icon	Function
	Open a list
	Date Range
	Add a new record
	Navigate to the first record

Table 1-1 (Cont.) Symbols and Icons - Common

Symbol/Icon	Function
	Navigate to the last record
	Navigate to the previous record
	Navigate to the next record
	Grid view
	List view
	Refresh
	Click this icon to add a new row.
	Click this icon to delete a row, which is already added.
	Calendar
	Alerts
	Unlock Option
	View Option
	Reopen Option

2

Gateway Functions - An Overview

- [Introduction](#)
Integration of multiple applications and solutions is a key aspect of modern systems. Specialized applications deployed across different platforms and infrastructures require seamless communication and integration with Oracle Banking Trade Finance for data exchange. The Oracle Banking Trade Finance Integration Gateway, referred to as the Gateway in this document, addresses these integration requirements.
- [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 are 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'.
- [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.
- [Responsibilities of Integration Gateway](#)
The primary responsibilities of Oracle Banking Trade Finance Integration Gateway include the following:
- [Deployment of Oracle Banking Trade Finance Integration Gate- way](#)
Message communication, incoming from or outgoing to an external system, occurs only through the Oracle Banking Trade Finance Integration Gateway. The Gateway serves as the first or last point of contact with the database in the message flow. The Oracle Banking Trade Finance Integration Gateway supports both distributed and single schema deployments of Oracle Banking Trade Finance.:
- [Deployment Patterns for Application Integration](#)

2.1 Introduction

Integration of multiple applications and solutions is a key aspect of modern systems. Specialized applications deployed across different platforms and infrastructures require seamless communication and integration with Oracle Banking Trade Finance for data exchange. The Oracle Banking Trade Finance Integration Gateway, referred to as the Gateway in this document, addresses these integration requirements.

From the Gateway perspective, supported integration needs are broadly categorized as follows:

- Inbound application integration – Inbound application integration is used when an external system adds, modifies, or queries information within Oracle Banking Trade Finance.
- Outbound application integration – Outbound application integration is used when an external system receives notifications of 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 are 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
- [EJB Based Synchronous Deployment Pattern](#)
The Enterprise Java Beans (EJB) deployment pattern is 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 uses the RMI/IIOP protocol to communicate with the Oracle Banking Trade Finance EJB.
- [Web Services Based Synchronous Deployment Pattern](#)
The web services deployment pattern is used in integration scenarios where an external system connects to Oracle Banking Trade Finance using standards based, interoperable web services.
- [HTTP Servlet Based Synchronous Deployment Pattern](#)
The HTTP servlet deployment pattern is used in integration scenarios where an external system connects to Oracle Banking Trade Finance using simple HTTP messages.
- [MDB Based Asynchronous Deployment Pattern](#)
The MDB deployment pattern is used in integration scenarios where an external system connects to Oracle Banking Trade Finance using JMS queues.

2.2.1 EJB Based Synchronous Deployment Pattern

The Enterprise Java Beans (EJB) deployment pattern is 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 uses the RMI/IIOP protocol to communicate with the Oracle Banking Trade Finance EJB.

In this deployment pattern, the EJB exposed by Oracle Banking Trade Finance is a stateless session bean. Requests are received as XML messages. After processing within Oracle Banking Trade Finance, the system returns the response to the external system as an XML message. Transaction control for processing remains with the Oracle Banking Trade Finance EJB.

2.2.2 Web Services Based Synchronous Deployment Pattern

The web services deployment pattern is used in integration scenarios where an external system connects to Oracle Banking Trade Finance using standards based, interoperable web services.

This deployment pattern is especially applicable to systems which meet the following 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 uses 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 are in the form of an XML message, but the request is 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 that is a 'payload' within the response SOAP message.

The transaction control for the processing stays with the Oracle Banking Trade Finance.

2.2.3 HTTP Servlet Based Synchronous Deployment Pattern

The HTTP servlet deployment pattern is used in integration scenarios where an external system connects 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 makes a HTTP request to the Oracle Banking Trade Finance servlet.

For this deployment pattern, Oracle Banking Trade Finance displays a single servlet. The actual request is 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 stays with the Oracle Banking Trade Finance.

2.2.4 MDB Based Asynchronous Deployment Pattern

The MDB deployment pattern is used in integration scenarios where an external system connects 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

The 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 are 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 from or outgoing to an external system, occurs only through the Oracle Banking Trade Finance Integration Gateway. The Gateway serves as the first or last point of contact with the database in the message flow. The Oracle Banking Trade Finance Integration Gateway supports both 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 selects to communicate only through Web Services.
	Asynchronous	FLEXCUBE OBTF MDB	This can be used if the external system selects 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 selects to communicate only through Web Services.
	Asynchronous	FLEXCUBE OBTF MDB	This can be used if the external system selects to communicate only through JMS queues
Handoffs from Oracle Banking Trade Finance	Asynchronous	FLEXCUBE OBTF Notify	Recommended

3

External System Maintenance

- [Introduction](#)
- [External System](#)
- [Access Rights to an External System](#)
- [Gateway Password Generation Logic For External System Authentication](#)
This section describes the Encryption information for gateway authentication.
- [Incoming Message Browser](#)
This topic provides the systematic instructions to incoming message browser.
- [Duplicate Message Browser](#)
This topic provides the systematic instructions to duplicate message browser.
- [Outgoing Message Browser](#)
This topic provides the systematic instructions to outgoing message browser.
- [Define Notification Messages](#)
This topic provides the systematic instructions to define notification message.
- [Define the Notifications Enroutes](#)
This topic provides the systematic instructions to define the notifications enroutes.
- [Amendment Maintenance](#)
- [Gateway Maintenance for Approot Deployment](#)

3.1 Introduction

This topic contains the following sub-topics:

3.2 External System

This topic contains the following sub-topics:

- [Define an External System](#)
This topic provides the systematic instructions to define an external system.
- [Specifying External System Details](#)
- [Indicating Correlation Pattern](#)
- [Specifying Message Exchange Pattern](#)
- [Specifying Queue Details](#)
- [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.
- [View External System Details](#)
This topic provides the systematic instructions to view external system details.

- [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.

3.2.1 Define an External System

This topic provides the systematic instructions to define an external system.

Define an external system that communicates with the Oracle Banking Trade Finance Integration Gateway, using the **External System Maintenance** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWDETSYS** in the text box, and click the next arrow.

The **External System Maintenance** screen is displayed.

2. On **External System Maintenance** screen, specify the required fields.

For information on fields, refer to [Specifying External System Details](#):

3.2.2 Specifying External System Details

Table 3-1 External System - Field Description

Field	Description
External System	Specify a name for the External System . Ensure this is same as the Source in an incoming message.
Description	Specify a brief Description for the External System.

3.2.3 Indicating Correlation Pattern

Table 3-2 Correlation Pattern - Field Description

Field	Description
Request	Define a method for correlating Request messages with response messages. The external system can use the Message ID of the request message as the Correlation ID in the response message. Alternatively, the Correlation ID of the request message can be retained as the Correlation ID of the corresponding response message.

3.2.4 Specifying Message Exchange Pattern

Table 3-3 Message Exchange Pattern - Field Description

Field	Description
Request Message	The Request Message can be selected as Full Screen or Input Only. When Full Screen is selected as the request message, the response message is also displayed as Full Screen.
Response Message	Select the Response Message to be Full Screen or Record Identification Msg. Note If Input Only is selected in the request message, select 'Record Identification Msg' as the response message.

3.2.5 Specifying Queue Details

Table 3-4 Queue Details - Field Description

Field	Description
Default Response Queue	User can define a response queue for each of the In Queue's through which the External System communicates 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 are redirected to a queue with the name of the request queue appended with '_E'.

Table 3-4 (Cont.) Queue Details - Field Description

Field	Description
XSD Validation Required	Check this box to indicate if the request message is validated against its corresponding XSD.
Register Response Queue Message ID	Check this box to indicate if the message ID provided by the Response Queue are 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.

Table 3-5 External System Queue Details - Field Description

Field	Description
In Queue	Specify the name of the queue from which the messages were received. The name of the queue helps to 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. Map multiple queues to a source. System allows a source to post messages to multiple queues.
Response Queue	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 View External System Details

This topic provides the systematic instructions to view external system details.

The details of previously defined external Systems can be viewed using the **External System Summary** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWSETSYS** in the text box, and click the next arrow.

The **External System Summary** screen is displayed.

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

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

The **Result** list displays the external systems which match the query.

2. On **External System Summary** screen, specify the fields.

For information on fields, refer to [Table 3-6](#):

Table 3-6 External System Summary - Field Description

Field	Description
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 the search criteria, click Query to view the list of results which match the 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.

User 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 to 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 topic contains the following sub-topics:

- [Define Access Rights to an External System](#)
This topic provides the systematic instructions to define access rights to an external system.
- [View External System Function Details](#)
This topic provides the systematic instructions to view external system function details.
- [Defining Access Rights to a User ID](#)
- [Maintain Upload Source Details](#)
This topic provides the systematic instructions to maintain upload source.
- [Specify Upload Source Preferences](#)
This topic provides the systematic instructions to specify upload source preferences.
- [Maintain Function ID Preferences](#)
This topic provides the systematic instructions to maintain function ID preferences.

3.3.1 Define Access Rights to an External System

This topic provides the systematic instructions to define access rights to an external system.

Define the access rights to an external system using the **External System Functions** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWDETFUN** in the text box, and click the next arrow.
The **External System Functions** screen is displayed.

The screenshot shows the 'External System Functions' screen. It features a title bar with a close button. Below the title bar are 'New' and 'Enter Query' buttons. The main content area is split into two panels. The left panel contains five input fields: 'External System *', 'Function *', 'Action *', 'Service Name', and 'Operation Code'. The first three fields have search icons. The right panel contains a 'Description' input field and a 'Bulk SMS Check' toggle switch. At the bottom, there are 'Fields', 'Audit', and 'Exit' buttons.

2. On **External System Functions** screen, specify the fields.
For information on fields, refer to [Table 3-7](#):

Table 3-7 External System Functions - Field Description

Field	Description
External System	Select an External System for access rights from the list of values. The adjoining option list displays all external systems maintained in the External System Functions screen.
Function ID	Select a Function ID from the list of values. The function ids are loaded from Gateway Functions.
Action	Specify the Action for the external system.
Service Name	This displays the Service Name based on the Function ID selected.
Operation Code	This displays the Operation Code based on the Function ID selected.

3.3.2 View External System Function Details

This topic provides the systematic instructions to view external system function details.

User can view the access rights details that are already defined using the **External System Functions Summary** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWSETFUN** in the text box, and click the next arrow.

The **External System Functions Summary** screen is displayed.

Search for external system functions that matches any of the following criteria:

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

The **Result** list displays the external system functions that matches the query.

2. On **External System Functions Summary** screen, specify the fields.

For information on fields, refer to [Table 3-8](#):

Table 3-8 External System Functions Summary - Field Description

Field	Description
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, to begin a new search.
Query	After specifying the search criteria, click Query to view the list of results that matches the 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, map each Function ID to a User ID.

Note

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

3.3.4 Maintain Upload Source Details

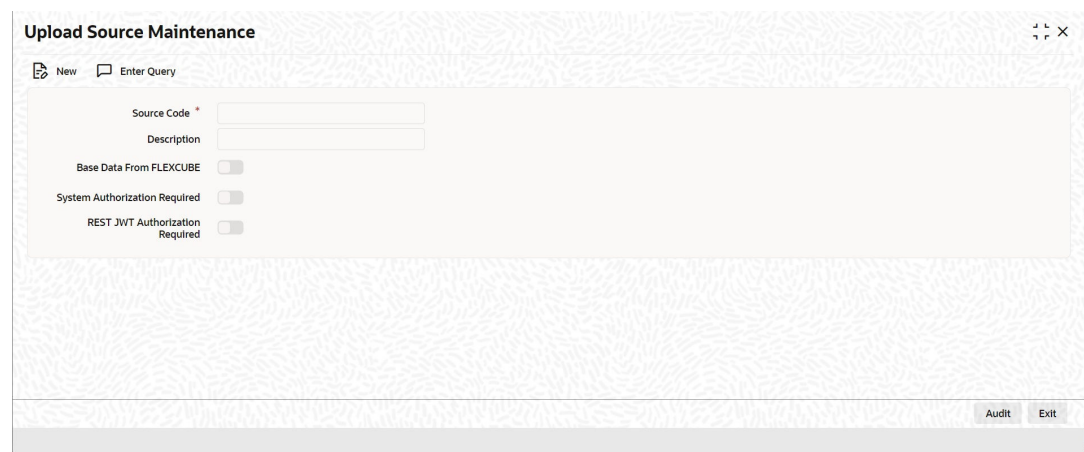
This topic provides the systematic instructions to maintain upload source.

Oracle Banking Trade Finance supports data upload from an external source. Source details for data upload are maintained in Oracle Banking Trade Finance using the **Upload Source Maintenance** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **CODSORCE** in the text box, and click the next arrow.

The **Upload Source Maintenance** screen is displayed.



2. On **Upload Source Maintenance** screen, specify as required fields.

For information on fields, refer to [Table 3-9](#):

Table 3-9 Upload Source Maintenance - Field Description

Field	Description
Source Code	Specify a Source Code for data upload to Oracle Banking Trade Finance.
Description	Specify a small Description for the specified source code.
Base Data From FLEXCUBE	Check this box to indicate if base data has to be uploaded from Oracle Banking Trade Finance.

3.3.5 Specify Upload Source Preferences

This topic provides the systematic instructions to specify upload source preferences.

User can set preferences for upload of data from an external source in the **Upload Source Preferences Maintenance** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **CODUPLDM** in the text box, and click the next arrow.

The **Upload Source Preferences Maintenance** screen is displayed.

2. On **Upload Source Preferences Maintenance** screen, specify the fields.

For information on fields, refer to [Table 3-10](#):

Table 3-10 Upload Source Preferences Maintenance - Field Description

Field	Description
Source Code	Select the Source Code from the option list. Based on the selected Source Code , the system uploads data from that source into Oracle Banking Trade Finance.
Module Code	Select to upload data from a source directly onto a module in Oracle Banking Trade Finance. This field indicates the module for data upload from the specified source.

Table 3-10 (Cont.) Upload Source Preferences Maintenance - Field Description

Field	Description
On Override	Oracle Banking Trade Finance generates override messages in case, it encounters any discrepancies during data upload. Select 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. Select to put the record with the error on hold. In thses cases, select Move To Queue from the list of options available. To reject the record, select Reject .
Post Upload Status	To automatically authorize the data that is uploaded into Oracle Banking Trade Finance, select the Authorize option. To put on hold the record , select this option in this field. To unauthorize the record, select the Unauthorized option in this field. The record is not authorized automatically on upload. 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 Maintain Function ID Preferences

This topic provides the systematic instructions to maintain function ID preferences.

Specify the **User ID** and **Password**, and login to Homepage.

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

The **Function ID Preferences** screen is displayed.

The screenshot displays the 'Function ID Preferences' window. It features a table with five columns: 'Function', 'Status', 'On Queue Exception', 'On Override', and 'Proceed with EOD'. The table is currently empty, with a message 'No data to display.' below it. A pagination bar at the bottom of the table indicates 'Page 1 (0 of 0 items)' and includes navigation arrows. At the bottom right of the window, there are 'Exit' and 'Save' buttons.

2. On **Function ID Preferences** screen, specify the details as required, click **OK** to save the details, else click **Exit** or **Cancel**.

For information on fields, refer to [Table 3-11](#):

Table 3-11 Upload Source Preferences Maintenance - Field Description

Field	Description
Function Id	Specify the Function Id . The adjoining option list displays all the function Ids maintained in the system.
Status	Select the Status of the record from the following drop-down list: • Authorized • Unauthorized • Put On hold
On Exception	Select the record status in case of any exception while saving from the following drop-down list: • Put On Hold • Reject
On Override	Select the record status in case of any overriding a saved record from the following drop-down list: • Put On Hold • Reject • Ignore
Proceed With EOD	Check this box, 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:

- [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.

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

This topic provides the systematic instructions to incoming message browser.

The messages received from the external system are displayed in the Incoming Message Browser screen.

Specify the **User ID** and **Password**, and login to Homepage.

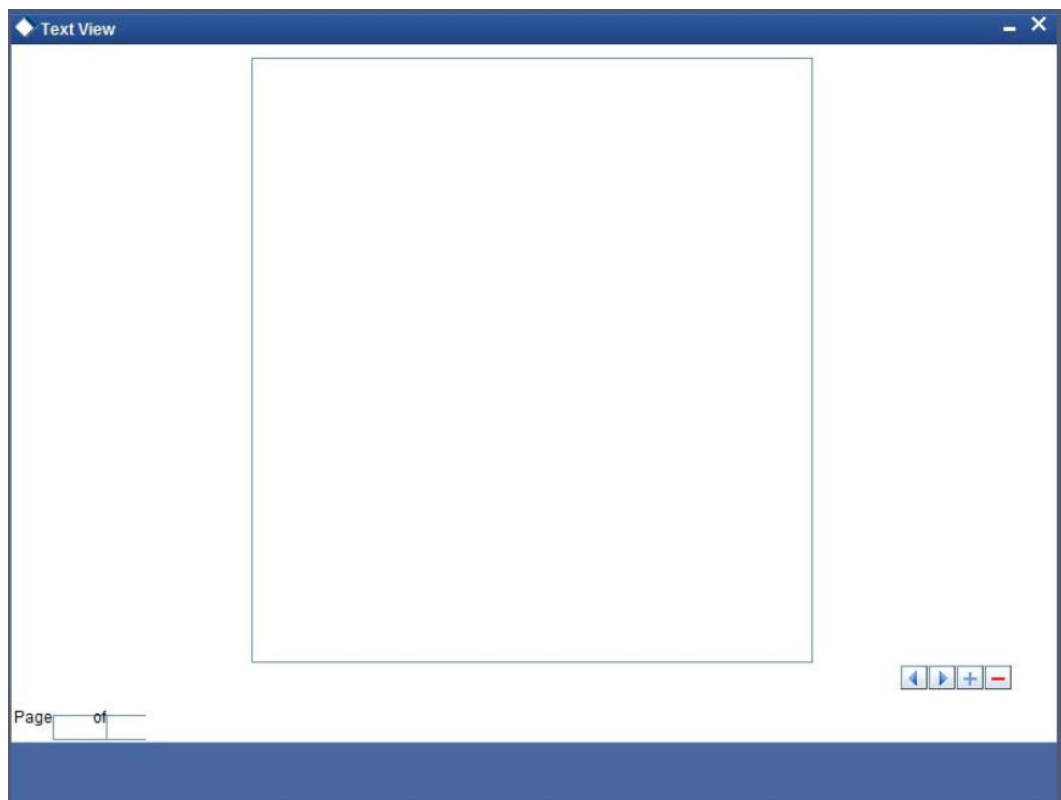
1. On the Homepage, type **GWDINBRW** in the text box, and click the next arrow.

The Incoming Message Browser - Detail

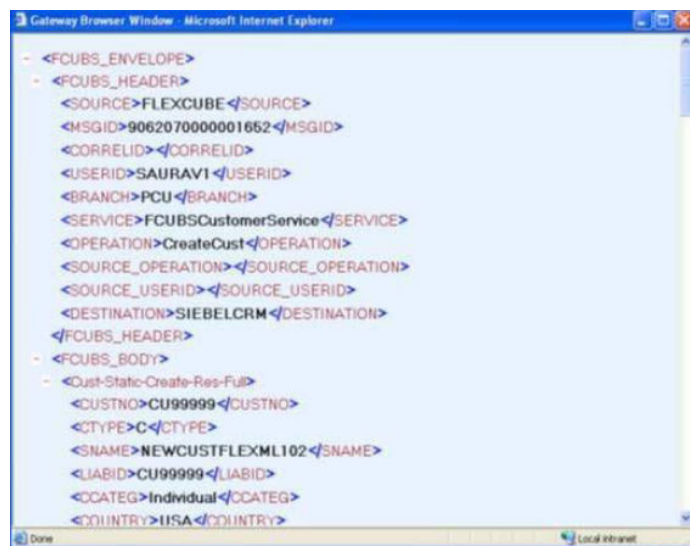
In the **Incoming Message Browser** screen, user can view the details of the messages received from the external systems. User can also view the messages in the XML format or the Text format.

2. Click the **Text View** button to view the incoming messages in text format.

The **Text View** screen is displayed.



3. Click the **XML View** button to view the **Gateway Browser Window** screen that displays the messages in XML format.



- [View Incoming Message Details](#)
This topic provides the systematic instructions to view incoming message details.

3.5.1 View Incoming Message Details

This topic provides the systematic instructions to view incoming message details.

The summary of all messages received from the external system can be viewed using the **Incoming Message Browser - Summary** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWSINBRW** in the text box, and click the next arrow.

Specify the criteria (Message Reference Number, External System, Service Name, etc) for an incoming messages. The **Search Result** list displays the messages that matches the query.

2. On **Incoming Message Browser** screen, specify the fields.

For information on fields, refer to [Table 3-12](#):

Table 3-12 Incoming Message Browser - Field Description

Field	Description
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, to begin a new search.
Search	Specify the Search criteria to view the list of results that matches the search criteria.

3.6 Duplicate Message Browser

This topic provides the systematic instructions to duplicate message browser.

If Oracle Banking Trade Finance Integration Gateway identifies a duplicate message it will be sent to the Duplicate Messages Browser.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWSPBROW** in the text box, and click the next arrow.

The **Duplicate Message Browser - Summary** screen is displayed.

Duplicate Message Browser Summary

Search Advanced Search Reset

Case Sensitive

Message Reference External System Correlation Id Service Name User Id Server Date Stamp

Branch Message Id Request Queue Message Id Operation Code Their User Id 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

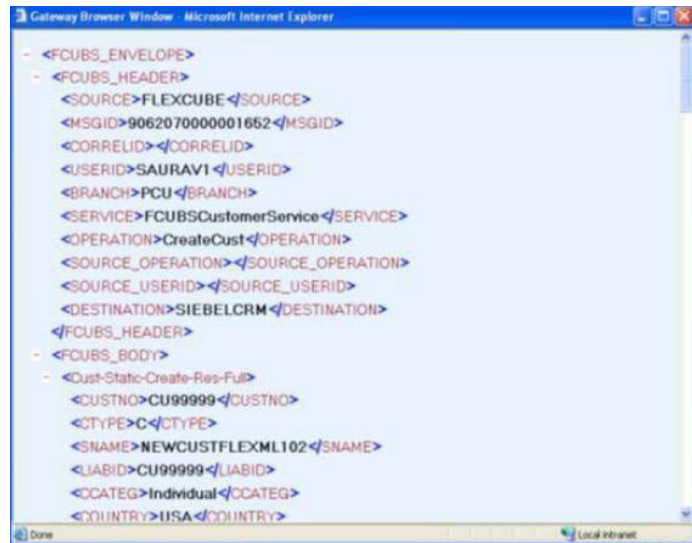
2. Click 'Text View' button in the 'Duplicate Message Browser - Summary' screen to view the duplicate messages in the text format.

Duplicate Message Browser - Text View

Page of

Navigation buttons: left arrow, right arrow, plus, minus

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



- [Query Duplicate Message Browser](#)
This topic provides the systematic instructions to query duplicate message browser.

3.6.1 Query Duplicate Message Browser

This topic provides the systematic instructions to query duplicate message browser.

User can query the list of duplicate messages using the **Duplicate Message Browser Summary** screen with functions for search.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWSPBROW** in the text box, and click the next arrow.
The Summary screen is displayed.

Summary

Advanced Search Reset

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

Message Reference	Branch	External System	Message Id	Correlation Id	Request Queue Message Id	Service Name	Operation Code	User Id

Exit

Specify the criteria (Message Reference Number, External System, Service Name etc) for duplicate messages. The **Search Result** list displays the messages that matches the query.

- On **Duplicate Message Browser Summary** screen is displayed, specify the fields.
For information on fields, refer to [Table 3-13](#):

Table 3-13 Duplicate Message Browser Summary - Field Description

Field	Description
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 fieldsto begin a new search.
Query	After specifying search criteria, click Query to view the list of results that matches the search criteria.
Refresh	Click Refresh to refresh the list of results.

3.7 Outgoing Message Browser

This topic provides the systematic instructions to outgoing message browser.

After processing incoming messages, the system sends a response message to the external systems with the processing status. Response messages are displayed in the **Outgoing Message Browser**.

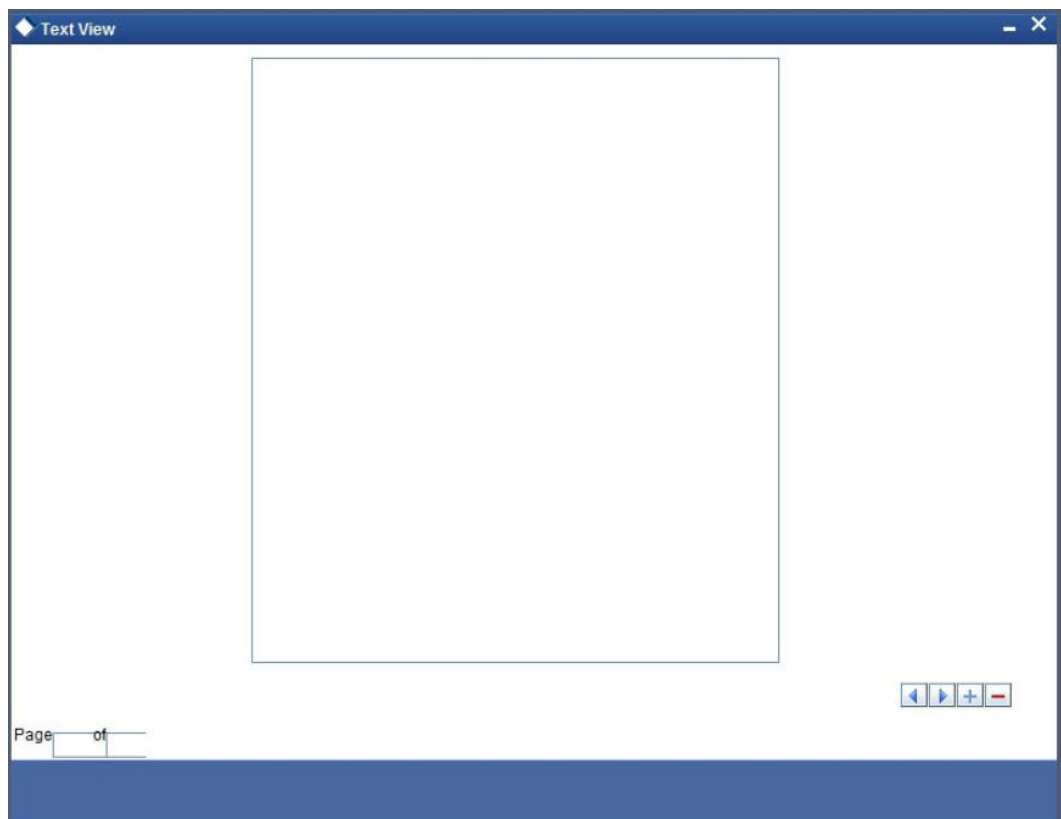
Specify the **User ID** and **Password**, and login to Homepage.

- On the Homepage, type **GWDOTBRW** in the text box, and click the next arrow.
The **Outgoing Message Browser** screen is displayed.

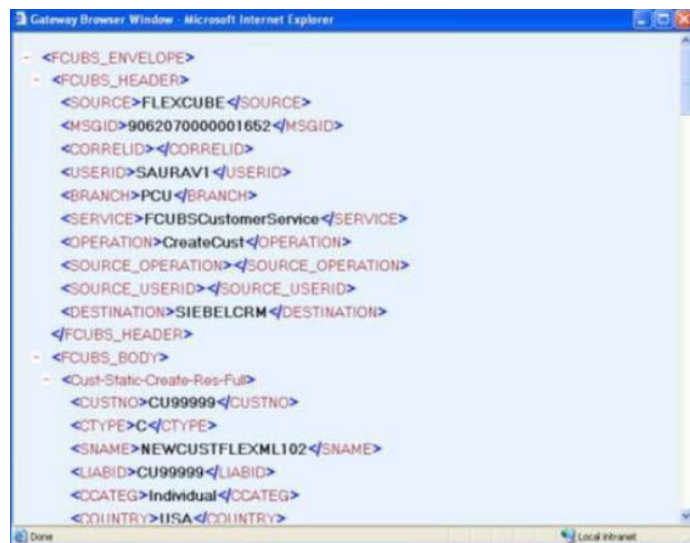
The screenshot shows the 'Outgoing Message Browser' web application. At the top, there is a title bar with the text 'Outgoing Message Browser' and standard window controls. Below the title bar is a search bar with the placeholder text 'Enter Query'. The main content area is divided into two columns of input fields. The left column contains fields for 'Message Reference' (with sub-fields 'Message Id', 'Message Status', 'Operation Code', 'FLEXCUBE Reference', 'Their User Id', 'Response Queue Message Id', and 'Queue Name'), each with a small icon to its right. The right column contains fields for 'External System' (with sub-fields 'Correlation Id', 'Service Name', 'Branch', 'User Id', 'Server Date Stamp', 'Branch Date', and 'Related Message Reference'), each with a small icon to its right. Below these columns is a 'Repair Reason' text area. At the bottom of the form, there are two buttons: 'Text View' and 'XML View'. In the bottom right corner, there is an 'Exit' button.

In the **Outgoing Message Browser** screen, user can view the details of the messages sent to the external systems. User can also view the messages in the XML format or the Text format.

- Click the **Text View** button to view the response message in text format.



3. Click the **XML View** button to view the response messages in XML format.



- [Query Outgoing Message Browser](#)
This topic provides the systematic instructions to query outgoing message browser.

3.7.1 Query Outgoing Message Browser

This topic provides the systematic instructions to query outgoing message browser.

User can query the list of outgoing messages using the **Outgoing Message Browser** screen. Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWSOTBRW** in the text box, and click the next arrow.
The **Outgoing Message Browser** screen is displayed.

Specify the criteria (Message Reference Number, External System, Service Name etc) for outgoing messages. The **Search Result** list displays the messages that matches the query.

2. On **Outgoing Message Browser** screen, specify the fields.
For information on fields, refer to [Table 3-14](#):

Table 3-14 Outgoing Message Browser - Field Description

Field	Description
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 to begin a new search.
Search	Specify the Search criteria to view the list of results that matches the search criteria.

3.8 Define Notification Messages

This topic provides the systematic instructions to define notification message.

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, maintain notification codes for each branch using the **Notifications Installed** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWDNTFIN** in the text box, and click the next arrow.

The **Notifications Installed** screen is displayed.

2. On **Notifications Installed** screen, specify the fields.

For information on fields, refer to [Table 3-15](#):

Table 3-15 Notifications Installed - Field Description

Field	Description
Branch Code	Select a Branch Code to assign a notification code. Once you select the branch code, the corresponding description is displayed.
Branch Name	The system displays Branch Name when the Branch Code is selected.
Notification Code	Select a Notification Code to link with the branch from list of values. This field generates notification for the entire branch. In addition to maintaining a Notification Code for a given branch, user can also specify a topic or a queue to which the notification messages should be sent. User can define a topic/queue for a given branch and notification code in the Notifications Enroute screen.
Description	The system displays Description when the Notification Code is selected.

- [View Installed Notification Details](#)
This topic provides the systematic instructions to view installed notification.

3.8.1 View Installed Notification Details

This topic provides the systematic instructions to view installed notification.

User can view and query the list of notification messages using the **Notifications Installed Summary** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWSNTFIN** in the text box, and click the next arrow.
The **Notifications Installed Summary** screen is displayed.

Specify the criteria (Message Reference Number, External System, Service Name, etc) for notification messages. The **Search Result** list displays the messages that matches the query.

2. On **Notifications Installed - Summary** screen, specify the fields.

For information on fields, refer to [Table 3-16](#):

Table 3-16 Notifications Installed Summary- Field Description

Field	Description
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 to begin a new search.
Search	Specify the Search criteria to view the list of results that matches the search criteria.

3.9 Define the Notifications Enroutes

This topic provides the systematic instructions to define the notifications enroutes.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWDNTFEN** in the text box, and click the next arrow.
The **Trade Notification Enroute Maintenance** screen is displayed.

2. On **Trade Notification Enroute Maintenance** screen, specify the fields.

For information on fields, refer to [Table 3-17](#):

Table 3-17 Trade Notification Enroute Summary - Field Description

Field	Description
Branch Code	Specify a branch for topic definition. To use a single topic for all branches for a given notification code, specify the wildcard ***.
Description	The system displays Description when the Branch Code is selected.
Notification Code	Select a Notification Code from the list of values. The user can specify the wildcard *** to use a single topic for all notifications in a branch.
Description	The system displays Description when the Notification Code is selected.
Destination Name	Specify the Destination Name . After all the details are specified, save the maintenance.

- [View Summary](#)
This topic provides the systematic instructions to view summary.

3.9.1 View Summary

This topic provides the systematic instructions to view summary.

User can view and query the list of notifications enroute using the **Trade Notification Enroute Summary** screen.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWSNTFEN** in the text box, and click the next arrow.
The **Trade Notification Enroute Summary** screen is displayed.

Specify the criteria (Message Reference Number, External System, Service Name etc) for notifications enroute. The **Search Result** list displays the messages that matches the query.

2. On **Notifications Enroutes - Summary** screen, specify the fields.

For information on fields, refer to [Table 3-18](#):

Table 3-18 Notifications Enroutes Summary- Field Description

Field	Description
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 to begin a new search.
Search	After specifying search criteria, click Search to view the list of results that matches the search criteria.

3.10 Amendment Maintenance

This topic contains the following sub-topics:

- [Maintain Gateway Amendment Details](#)
This topic provides the systematic instructions to maintain gateway amendment details.

3.10.1 Maintain Gateway Amendment Details

This topic provides the systematic instructions to maintain gateway amendment details.

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

- **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, user can consider OBTF BC Service. 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, user can consider Query 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 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.

In the **Gateway Amendment Maintenance** screen, a set of amendable fields is maintained. These fields can be amended in Oracle Banking Trade Finance when a corresponding request is received from an external system. Oracle Banking Trade Finance processes amendment requests based on this maintenance.

Specify the **User ID** and **Password**, and login to Homepage.

1. On the Homepage, type **GWDAMDMT** in the text box, and click the next arrow.

The **Gateway Amendment Maintenance** screen is displayed.

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

2. On **Gateway Amendment Maintenance** screen, specify the fields.

For information on fields, refer to [Table 3-19](#):

Table 3-19 Gateway Amendment Maintenance- Field Description

Field	Description
External System	Select the External System from the list of values. Based on the maintenance, only the fields that are selected as amendable can be modified if a request comes from the selected external system. Note The maintenance pertaining to external systems is factory shipped for the bank.
Origin System	Specify the Origin System for which the amendment details are applicable.

Table 3-19 (Cont.) Gateway Amendment Maintenance- Field Description

Field	Description
Source Operation	The free format text (without spaces) that identifies the amendment. This is mandatory. Note Source Operation is defaulted as (FUNCTIONID)_MODIFY, if the Source Operation is not sent from an external system and the function ID is derived from Service and Operation combination.
Service Name	Specify the Service Name from the list of values. 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 the bank.
Operation Code	Specify the Operation Code from the list of values. This is the operation that the external system performs within the selected service. The operation names are published by Oracle Banking Trade Finance. As an example, user can take 'Modify Contract', 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 the bank.
<u>Amendable Nodes</u>	
Node Name	Specify the Node Name . Alternatively, select the node name from the list of values. 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 topic contains the following sub-topics:

- [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:

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:

Function ID	Summary Function ID	Description
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

- [Introduction](#)
Data exchanged between Oracle Banking Trade Finance and external systems is in XML message format. These XML messages are defined in Oracle Banking Trade Finance as XML Schema Documents XSD and are referred to as OBTF formats.
- [Oracle Banking Trade Finance Envelope](#)
- [Oracle Banking Trade Finance NOTIFICATION](#)
The notification messages are generated in a standard format. The notification messages consist of two main components:

4.1 Introduction

Data exchanged between Oracle Banking Trade Finance and external systems is in XML message format. These XML messages are defined in Oracle Banking Trade Finance as XML Schema Documents XSD and are referred to as OBTF formats.

Each XML Schema is uniquely identified by its namespace and root element. The root element for gateway message XSDs is generated using 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 contains all the elements on the corresponding Oracle Banking Trade Finance screen.
 - IO (Input Only) - This indicates that the payload contains the only input elements on the corresponding Oracle Banking Trade Finance screen.
 - PK (Primary Key): This indicates that the payload contains only the Primary Key elements of the corresponding Oracle Banking Trade Finance screen.

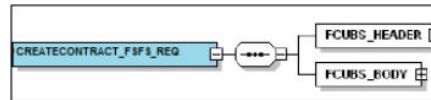
<MESSAGE_EXCHANGE_PATTERN> is a combination of message patterns. The <MESSAGE_EXCHANGE_PATTERN> available in OBTF is shown below:

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

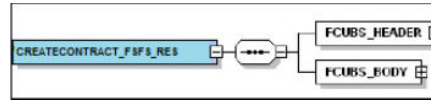
Non-Query Formats

For example, for the **CREATECONTRACT** operation, the root element of the associated messages is as follows:

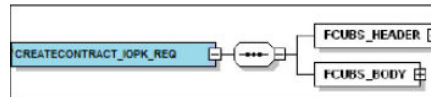
- CREATECONTRACT_FSFS_REQ



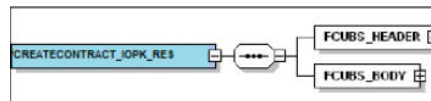
- `CREATECONTRACT_FSFS_RES`



- `CREATECONTRACT_IOPK_REQ`



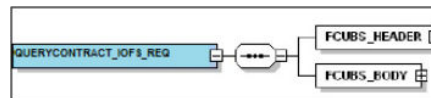
- `CREATECONTRACT_IOPK_RES`



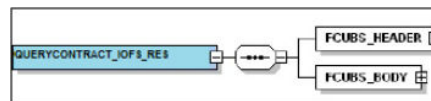
Query Formats

For example, for the **QUERYCONTRACT** operation, the root element of the associated messages is as follows:

- `QUERYCONTRACT_IOFS_REQ`



- `QUERYCONTRACT_IOFS_RES`



This chapter contains the following sections:

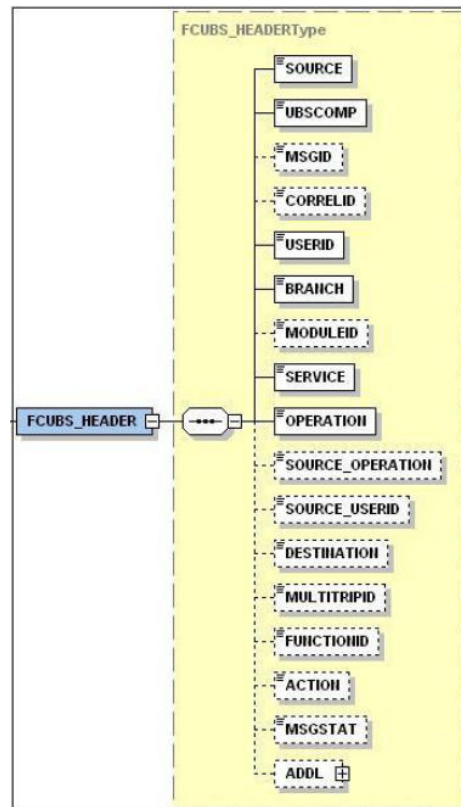
4.2 Oracle Banking Trade Finance Envelope

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

- **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 vary for each message.

This topic contains the following sub-topics:



- [FCUBS_HEADER](#)

- [FCUBS_BODY](#)

The FCUBS_BODY contains the actual payload to perform the respective transaction. The contents of the payload vary for each operation.

4.2.1 FCUBS_HEADER

The tags under FCUBS_HEADER are 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 incoming or outgoing message in Oracle Banking Trade Finance. Each message has a distinct Message ID.

CORRELID

This ID enables an external system that sends a request to Oracle Banking Trade Finance to correlate the response. In **External System Maintenance**, the correlation pattern is configured for each external system.

The configuration specifies whether the **MSGID** or the **CORRELID** from the request message is returned as the **CORRELID** in the response message. Based on this configuration, Oracle Banking Trade Finance sets the appropriate value in the response message.

USERID

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

For response messages, the value is null.

BRANCH

This indicates the Oracle Banking Trade Finance branch code where the request message is processed. If **BRANCH** is not specified in the header, the system processes the request message in the 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 used to load the request message from the source.

DESTINATION

For incoming messages, the destination is Oracle Banking Trade Finance. For response messages, system populates the **SOURCE** of the request message as **DESTINATION**.

MULTITRIPID

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**.

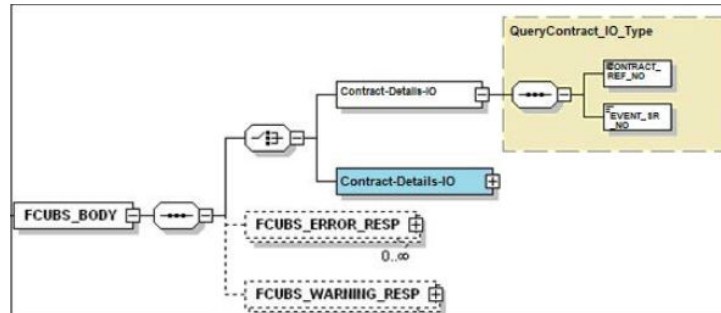
ADDL

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

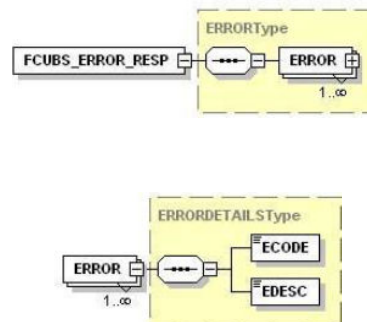
4.2.2 FCUBS_BODY

The FCUBS_BODY contains the actual payload to perform the respective transaction. The contents of the payload vary for each operation.

The following snapshot displays a sample FCUBS_BODY of QUERYCONTRACT operation.



FCUBS_BODY contains additional nodes for error response and warning response. A diagrammatic representation of the Error response is as follows:



- [FCUBS_ERROR_RESP](#)
Oracle Banking Trade Finance sends an error response message when errors occur during a transaction. The error response includes an **ERROR** tag.
- [FCUBS_WARNING_RESP](#)
A warning response message is sent when overrides are raised during a transaction. The warning response includes a **WARNING** tag.

4.2.2.1 FCUBS_ERROR_RESP

Oracle Banking Trade Finance sends an error response message when errors occur during a transaction. The error response includes an **ERROR** tag.

ERROR

The **ERROR** node contains tags for the error code and error description. Oracle Banking Trade Finance generates an **ERROR** node for each error raised.

4.2.2.2 FCUBS_WARNING_RESP

A warning response message is sent when overrides are raised during a transaction. The warning response includes a **WARNING** tag.

WARNING

This node contains tags for the warning code and warning description. Oracle Banking Trade Finance generates a **WARNING** node for each override raised.

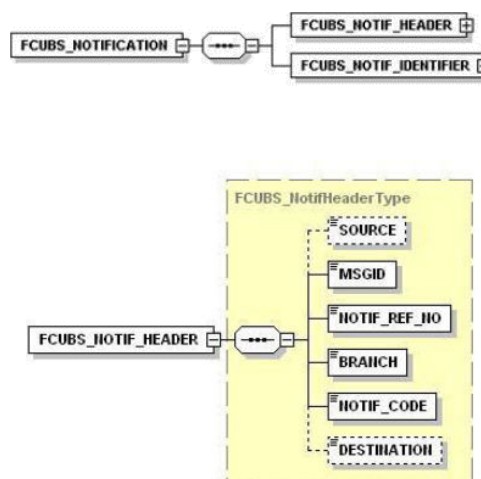
4.3 Oracle Banking Trade Finance NOTIFICATION

The notification messages are generated in a standard format. The notification messages 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 identifies maintenance records based on the information provided under this node. The node contents vary for each notification.

A diagrammatic representation of OBTF NOTIFICATION is as follows:



- [OBTF NOTIFICATION HEADER](#)

The tags under OBTF NOTIFICATION HEADER are described as follows:

4.3.1 OBTF NOTIFICATION HEADER

The tags under OBTF NOTIFICATION HEADER are described as follows:

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 is triggered.

DESTINATION

For incoming messages, DESTINATION is Oracle Banking Trade Finance. For response messages, the system populates DESTINATION with the SOURCE value from the request message.

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

5

Annexure - List of Messages

- [Introduction](#)
This chapter lists the messages under the scope of the following Gateway services:
- [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 loaded by any external system to perform the relevant operation in Oracle Banking Trade Finance.

5.1 Introduction

This chapter lists the messages under the scope of the following Gateway services:

- Inbound application integration is used when an external system adds, modifies, or queries information within Oracle Banking Trade Finance.
- Outbound application integration is used when any external system notifies 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 loaded by any external system to perform the relevant operation in Oracle Banking Trade Finance.

This topic contains the following sub-topics:

- [OBTFBCService](#)
All Bills and Collections operations are logically grouped under a service called 'OBTFBCService'.
- [OBTFCLService](#)
All Letter of Credit operations are logically grouped under a service called 'OBTFCLService'.

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](#)
This is the operation code for querying the BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [CreateContract](#)
This is the operation code for creating new BC contract in Oracle Banking Trade Finance. It involves the following messages:

- [ModifyContract](#)
This is the operation code for modifying the BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [LiquidateContract](#)
This is the operation code for liquidating the BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [AuthorizeContract](#)
This is the operation code for authorizing the BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [CloseContract](#)
This is the operation code for closing the BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [DeleteContract](#)
This is the operation code for deleting the BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [ReverseContract](#)
This is the operation code for reversing the BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [RolloverContract](#)
This is the operation code for rollover the BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [CreateBCContractSim](#)
This is the operation code for simulating BC contract in Oracle Banking Trade Finance. It involves the following messages:
- [ModifyBCContractSim](#)
This is the operation code for simulating the BC contract modification in Oracle Banking Trade Finance. It involves the following messages:
- [LiquidateBCContractSim](#)
This is the operation code for simulating the BC contract liquidation in Oracle Banking Trade Finance. It involves the following messages:
- [QueryBCContractSim](#)
This is the operation code for querying the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [AuthorizeBCContractSim](#)
This is the operation code for authorizing the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [CloseBCContractSim](#)
This is the operation code for closing the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [DeleteBCContractSim](#)
This is the operation code for deleting the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [ReverseBCContractSim](#)
This is the operation code for reversing the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [RolloverBCContractSim](#)
This is the operation code for rollover the BC contract simulation in Oracle Banking Trade Finance. It involves the following messages:

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 LiquidateBCCContractSim

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

- **LiquidateBCCContractSim-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.
- **LiquidateBCCContractSim-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.
- **LiquidateBCCContractSim-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.
- **LiquidateBCCContractSim-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 QueryBCCContractSim

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

- **QueryBCCContractSim-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.
- **QueryBCCContractSim-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 AuthorizeBCCContractSim

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

- **AuthorizeBCCContractSim-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](#)
This is the operation code for querying the LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [CreateContract](#)
This is the operation code for creating new LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [ModifyContract](#)
This is the operation code for modifying the LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [ReopenContract](#)
This is the operation code for reopening the LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [AuthorizeContract](#)
This is the operation code for authorizing the LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [CloseContract](#)
This is the operation code for closing the LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [DeleteContract](#)
This is the operation code for deleting the LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [ReverseContract](#)
This is the operation code for reversing the LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [RolloverContract](#)
This is the operation code for rollover the LC contract in Oracle Banking Trade Finance. It involves the following messages:

- [CreateContractSim](#)
This is the operation code for simulating LC contract in Oracle Banking Trade Finance. It involves the following messages:
- [ModifyContractSim](#)
This is the operation code for simulating the LC contract modification in Oracle Banking Trade Finance. It involves the following messages:
- [ReopenContractSim](#)
This is the operation code for simulating the LC contract reopen in Oracle Banking Trade Finance. It involves the following messages:
- [QueryContractSim](#)
This is the operation code for querying the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [AuthorizeContractSim](#)
This is the operation code for authorizing the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [CloseContractSim](#)
This is the operation code for closing the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [DeleteContractSim](#)
This is the operation code for deleting the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [ReverseContractSim](#)
This is the operation code for reversing the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [RolloverContractSim](#)
This is the operation code for rollover the LC contract simulation in Oracle Banking Trade Finance. It involves the following messages:
- [AuthorizeAmendConfirm](#)
This is the operation code for authorizing the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:
- [CloseAmendConfirm](#)
This is the operation code for closing the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:
- [ConfirmAmendConfirm](#)
This is the operation code for confirming the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:
- [DeleteAmendConfirm](#)
This is the operation code for deleting the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:
- [QueryAmendConfirm](#)
This is the operation code for querying the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:
- [ModifyAmendConfirm](#)
This is the operation code for modifying the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:
- [CreateAmendConfirm](#)
This is the operation code for creating the LC contract amendment in Oracle Banking Trade Finance. It involves the following messages:

- [QueryGuarantee](#)
This is the operation code for querying the guarantee in Oracle Banking Trade Finance. It involves the following messages:
- [CreateGuarantee](#)
This is the operation code for creating new guarantee in Oracle Banking Trade Finance. It involves the following messages:
- [ModifyGuarantee](#)
This is the operation code for modifying the guarantee in Oracle Banking Trade Finance. It involves the following messages:
- [ReopenGuarantee](#)
This is the operation code for reopening the guarantee in Oracle Banking Trade Finance. It involves the following messages:
- [AuthorizeGuarantee](#)
This is the operation code for authorizing the guarantee in Oracle Banking Trade Finance. It involves the following messages:
- [CloseGuarantee](#)
This is the operation code for closing the guarantee in Oracle Banking Trade Finance. It involves the following messages:
- [DeleteGuarantee](#)
This is the operation code for deleting the guarantee in Oracle Banking Trade Finance. It involves the following messages:
- [ReverseGuarantee](#)
This is the operation code for reversing the guarantee in Oracle Banking Trade Finance. It involves the following messages:

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.

Glossary

CODSORCE

Upload Source Maintenance - [Maintain Upload Source Details](#)

CODUPLDM

Upload Source Preferences Maintenance - [Specify Upload Source Preferences](#)

GWDAMDMT

Gateway Amendment Maintenance - [Maintain Gateway Amendment Details](#)

GWDINBRW

Incoming Message Browser - Detail - [Incoming Message Browser](#)

GWDNTFEN

Notification Enroutes Maintenance - [Define the Notifications Enroutes](#)

GWDNTFIN

Notifications Installed - [Define Notification Messages](#)

GWDOTBRW

Outgoing Message Browser - [Outgoing Message Browser](#)

GWSINBRW

Incoming Message Browser - [View Incoming Message Details](#)

GWSNTFEN

Notifications Enroutes Summary - [View Summary](#)

GWSNTFIN

Notifications Installed Summary - [View Installed Notification Details](#)

GWSOTBRW

Outgoing Message Browser - [Query Outgoing Message Browser](#)

GWSPBROW

Duplicate Message Browser Summary - [Duplicate Message Browser](#)