

Gateway Services
Oracle FLEXCUBE Universal Banking
Release 11.11.0.0.0
[May] [2022]
Part No. F55411-01



Table of Contents

1. ABOUT THIS MANUAL.....	1-1
1.1 INTRODUCTION.....	1-1
1.1.1 Audience	1-1
1.1.2 Acronyms and Abbreviations.....	1-1
1.1.3 Organization.....	1-1
1.1.4 Conventions Used in this Manual.....	1-1
1.2 GLOSSARY OF ICONS	1-2
2. GATEWAY FUNCTIONS - AN OVERVIEW	2-1
2.1 INTRODUCTION.....	2-1
2.2 INBOUND APPLICATION INTEGRATION.....	2-1
2.2.1 EJB Based Synchronous Deployment Pattern	2-1
2.2.2 Web Services Based Synchronous Deployment Pattern.....	2-2
2.2.3 HTTP Servlet Based Synchronous Deployment Pattern	2-2
2.2.4 MDB Based Asynchronous Deployment Pattern	2-3
2.3 OUTBOUND APPLICATION INTEGRATION.....	2-3
2.4 RESPONSIBILITIES OF INTEGRATION GATEWAY	2-3
2.5 DEPLOYMENT OF ORACLE FLEXCUBE INTEGRATION GATEWAY	2-3
2.6 DEPLOYMENT PATTERNS FOR APPLICATION INTEGRATION	2-4
2.7 FLEXML ADAPTER	2-5
3. EXTERNAL SYSTEM MAINTENANCE	3-1
3.1 DEFINING AN EXTERNAL SYSTEM.....	3-1
3.1.1 Viewing External System Details	3-3
3.1.2 Accessing Services and Operations	3-4
3.2 DEFINING ACCESS RIGHTS TO AN EXTERNAL SYSTEM.....	3-4
3.2.1 Viewing External System Function Details.....	3-5
3.2.2 Defining Access Rights to a User ID	3-7
3.2.3 Maintaining Upload Source Details	3-7
3.2.4 Specifying Upload Source Preferences.....	3-7
3.3 INCOMING MESSAGE BROWSER.....	3-9
3.3.1 Viewing Incoming Message Details.....	3-12
3.4 DUPLICATE MESSAGE BROWSER	3-13
3.4.1 Querying Duplicate Message Browser	3-15
3.5 OUTGOING MESSAGE BROWSER	3-16
3.5.1 Querying Outgoing Message Browser.....	3-18
3.6 DEFINING NOTIFICATION MESSAGES.....	3-19
3.6.1 Viewing Installed Notification Details.....	3-20
3.7 DEFINING THE NOTIFICATIONS ENROUTES	3-22
3.7.1 Viewing Summary screen.....	3-22
4. MESSAGE FORMATS.....	4-1
4.1 ORACLE FLEXCUBE ENVELOPE.....	4-1
4.1.1 FCUBS HEADER	4-2
4.2 ORACLE FLEXCUBE NOTIFICATION.....	4-4
4.2.1 FCUBS NOTIFICATION HEADER	4-4
5. FLEXML ADAPTER MESSAGE CONVERSION.....	5-1
5.1 INTRODUCTION.....	5-1
5.2 INBOUND FLEXML ADAPTER.....	5-1

5.2.1	<i>Processing Incoming FLEXML Request Message</i>	5-2
5.3	OUTBOUND FLEXML ADAPTER.....	5-2
5.3.1	<i>Processing Outbound Messages</i>	5-2
5.3.2	<i>Mapping FLEXML Format Messages to Gateway Format</i>	5-3

1. About this Manual

1.1 Introduction

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

It provides an overview to the module, and provides information on using the Integration Gateway module of Oracle FLEXCUBE.

Besides this User Manual, you can find answers to specific features and procedures in the Online Help, which can be invoked, by choosing 'Help Contents' from the Help Menu of the software. You can further obtain information specific to a particular field by placing the cursor on the relevant field and striking <F1> on the keyboard.

1.1.1 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

1.1.2 Acronyms and Abbreviations

Abbreviation	Description
FLEXCUBE UBS	FLEXCUBE Universal Banking Solution

1.1.3 Organization

The manual is organized in the following manner:

Chapter 1	<i>About this Manual</i> gives information on the intended audience. It also lists the various chapters covered in this User Manual.
Chapter 2	<i>Gateway Functions - An Overview</i> explains the features of this module
Chapter 3	<i>External System Maintenance</i> explains the maintenance of external systems and the various processes involved
Chapter 4	<i>Message Formats</i> explains the different message formats
Chapter 5	<i>FLEXML Adapter Message Conversion</i> explains details about the FLEXML Adapter.

1.1.4 Conventions Used in this Manual

Important information is preceded with the  symbol.

1.2 Glossary of Icons

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

Icons	Function
	Exit
	Add row
	Delete row
	Option List

2. Gateway Functions - An Overview

2.1 Introduction

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

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

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

2.2 Inbound Application Integration

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

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

FCUBS 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 FLEXCUBE EJB Based Synchronous Inbound Application Integration Deployment Pattern
- Oracle FLEXCUBE Web Services Based Synchronous Inbound Application Integration Deployment Pattern
- Oracle FLEXCUBE HTTP Servlet Based Synchronous Inbound Application Integration Deployment Pattern

Asynchronous Deployment Pattern is:

- Oracle FLEXCUBE MDB Based Asynchronous Inbound Application Integration Deployment Pattern

2.2.1 EJB Based Synchronous Deployment Pattern

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

In this deployment pattern the EJB displayed by Oracle FLEXCUBE will be a stateless session bean. The actual request will be in the form of an XML message. After the necessary processing is done in Oracle FLEXCUBE based on the request, the response is returned to the external system as an XML message. The transaction control for the processing will stay with the Oracle FLEXCUBE EJB.

2.2.2 Web Services Based Synchronous Deployment Pattern

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

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

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

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

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

2.2.3 HTTP Servlet Based Synchronous Deployment Pattern

The HTTP servlet deployment pattern will be used in integration scenarios where the external system connecting to Oracle FLEXCUBE wants to connect to Oracle FLEXCUBE 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 FLEXCUBE based upon the EJB interface; and/or
- Systems that prefer to use a simple http message based approach without wanting to use SOAP as the standard

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

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

2.2.4 **MDB Based Asynchronous Deployment Pattern**

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

This is especially applicable to systems such as the following:

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

Here external system sends messages in XML format to request queue on which an MDB is listening. When a message arrives on the queue, it is picked up for processing. After the necessary processing is done in Oracle FLEXCUBE, 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 FLEXCUBE Notify Application Integration layer. This application layer sends out notification messages to the external system whenever events occur in Oracle FLEXCUBE.

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

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

2.4 **Responsibilities of Integration Gateway**

The primary responsibilities of Oracle FLEXCUBE Integration Gateway include the following:

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

2.5 **Deployment of Oracle FLEXCUBE Integration Gateway**

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

- Distributed deployment of FCUBS – 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 FCUBS – In this situation the database components of the Gateway (messaging and business) are both deployed as part of the single Oracle FLEXCUBE schema.

2.6 Deployment Patterns for Application Integration

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

2.7 **FLEXML Adapter**

Oracle FLEXCUBE Gateway is the common channel for data exchange between Oracle FLEXCUBE and external systems. But, older versions of Oracle FLEXCUBE use FLEXML Gateway to interact with the external systems. The message or communication format of FLEXML is different from that of the gateway.

Hence, in order to facilitate the exchange of messages between such systems, Oracle FLEXCUBE uses a conversion mechanism called the FLEXML Adapter. FLEXML Adapter converts messages from FLEXML format to gateway format and vice versa.

For more details on the conversion mechanism of FLEXML Adapter, refer the chapter 'FLEXML Adapter Message Conversion'.

3. External System Maintenance

3.1 Defining an External System

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

You can define an external system using the 'External System – Detailed' screen.

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

The screenshot shows the 'External System - Detailed' configuration window. It includes the following elements:

- External System:** Fields for 'External System *' and 'Description'.
- Correlation Pattern:** A 'Request' dropdown menu currently set to 'Message ID'.
- Message Exchange Pattern:** 'Request Message' dropdown set to 'Input Only' and 'Response Message' dropdown set to 'Full Screen'. A checkbox for 'XSD Validation Required' is present.
- Queue:** Text boxes for 'Default Response Queue' and 'Dead Letter Queue'. A checkbox for 'Register Response Queue Message Id' is present.
- External System Queues:** A table with two columns: 'In Queue *' and 'Response Queue'. The table contains one row with empty input fields.
- Status Bar:** Displays 'Fields', 'FTP Parameter', 'Input By TEST12', 'Authorized By', 'Modification Number', 'Date Time', 'Date Time', 'Authorized' (unchecked), 'Open' (checked), and a 'Cancel' button.

The various details required by the 'External System - Detailed' screen are described below.

3.1.1.1 Specifying External System Details

External System

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

Description

Specify a brief description for the External System.

3.1.1.2 Indicating Correlation Pattern

Request

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

3.1.1.3 Specifying Message Exchange Pattern

Request Message

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

Response Message

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



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

3.1.1.4 Specifying Queue Details

Default Response Queue

You can define a response queue for each of the In Queue's through which the External System will communicate with Oracle FLEXCUBE. 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.



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

XSD Validation Required

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

Register Response Queue Message ID

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

3.1.1.5 Specifying External System Queue Details

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

In Queue

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

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

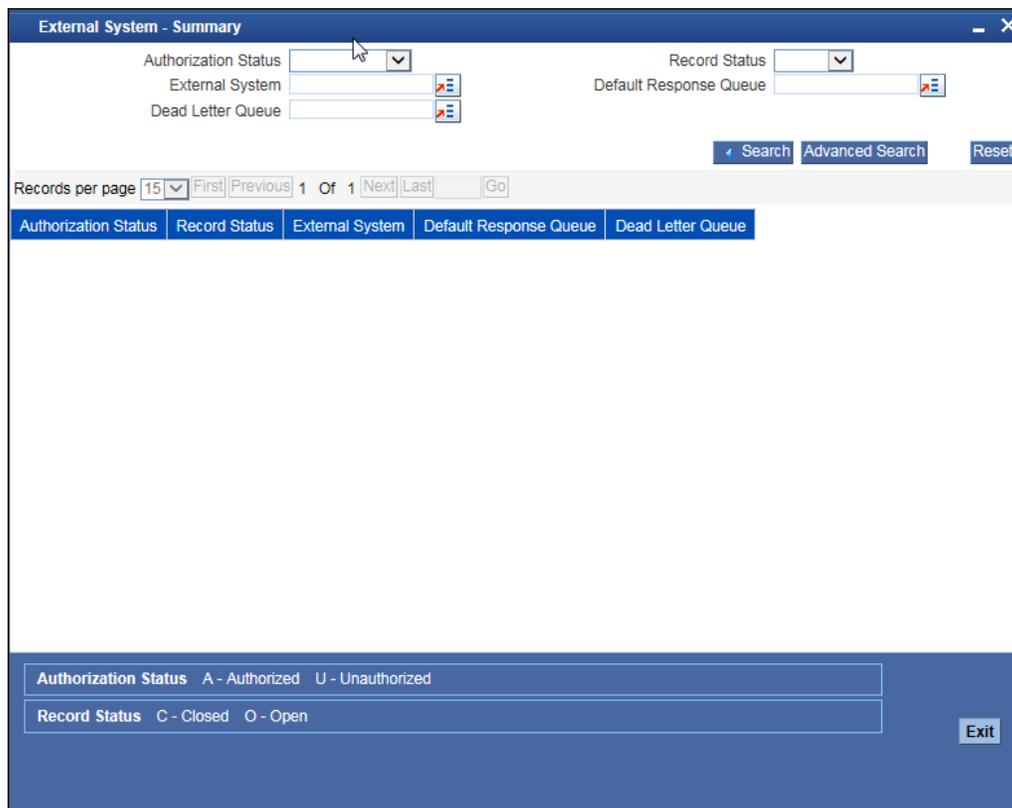
You can map multiple queues to a source. System will allow a source to post messages to multiple queues.

Response Queue

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

3.1.1 Viewing External System Details

The details of previously defined external Systems can be viewed using the 'External System - Summary' screen as shown below. You can invoke this screen by typing 'GWSEXSYS' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.



External System - Summary

Authorization Status Record Status

External System Default Response Queue

Dead Letter Queue

Search Advanced Search Reset

Records per page 15 First Previous 1 Of 1 Next Last Go

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

Authorization Status A - Authorized U - Unauthorized

Record Status C - Closed O - Open

Exit

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

Advanced

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

Reset

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

Query

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

Refresh

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

3.1.2 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 FLEXCUBE is linked with a function ID. This information is called Gateway Functions.

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

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

3.2 Defining Access Rights to an External System

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

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

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

External System

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

Function ID

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

Action

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

Service Name

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

Operation Code

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

3.2.1 Viewing External System Function Details

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

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

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

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

Advanced

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

Reset

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

Query

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

Refresh

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

3.2.2 **Defining Access Rights to a User ID**

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



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

3.2.3 **Maintaining Upload Source Details**

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

A screenshot of a web application window titled "Upload Source Maintenance -- Web Page Dialog". The window contains a form with the following fields: "Source Code *" (with a red asterisk), "Description", and a checkbox labeled "Base Data From Flexcube". Below the form is a "Fields" section with a table. The table has columns for "Input By", "Date Time", "Modification Number", "Authorized", and "Open". The "Authorized" and "Open" columns contain checkboxes. An "Exit" button is located in the bottom right corner of the window.

The following details need to be captured here:

Source Code

Specify a code for the for the source from which data has to be uploaded to Oracle FLEXCUBE.

Description

Give a small description for the source code specified.

Base Data From FLEXCUBE

Check this box to indicate if base data has to be uploaded from Oracle FLEXCUBE.

3.2.4 **Specifying Upload Source Preferences**

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

The following details are captured here:

Source Code

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

Module Code

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

On Override

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

- Ignore – Select this option to ignore such error messages and continue with the upload process
- Put on Hold – 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 FLEXCUBE generates an error message. You can choose to put the record with the error on hold. In such a case, choose 'Put on Hold' from the list of options available. If you would like to reject the record altogether, choose 'Reject'.

Post Upload Status

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

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

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

Purge Days (Calendar)

Specify the days maintained for purging of the data uploaded.

Allow Deferred Processing

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

Allow EOD with Deferred

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

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

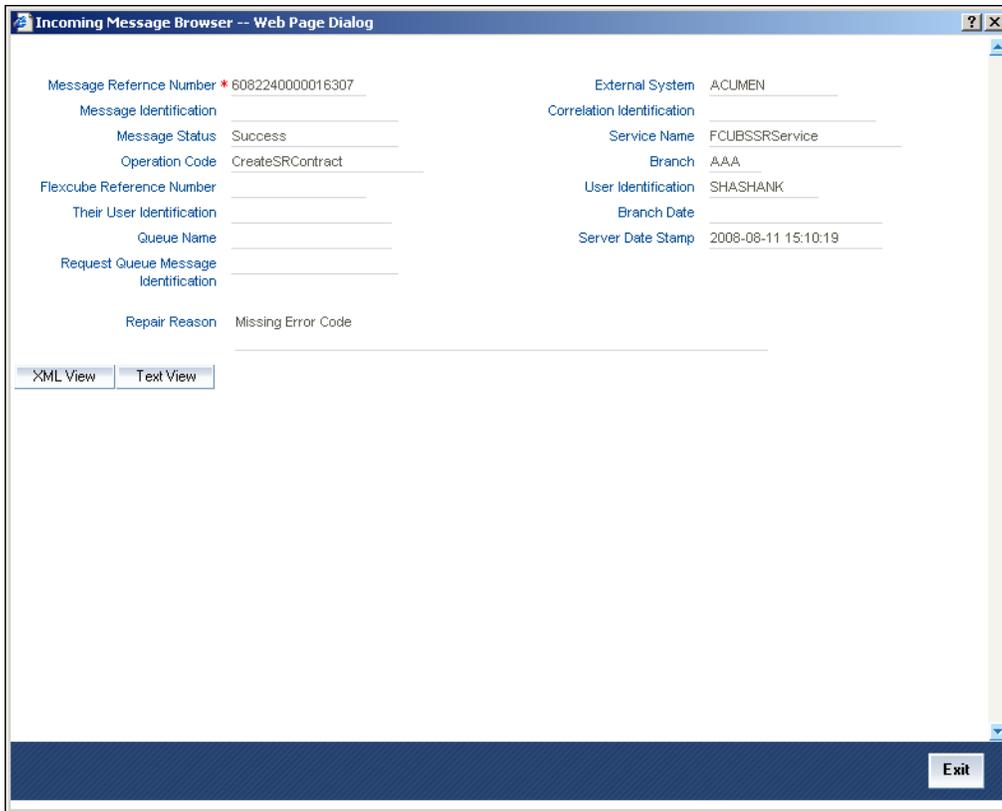
Allow Delete

Check this option to delete the process log.

3.3 Incoming Message Browser

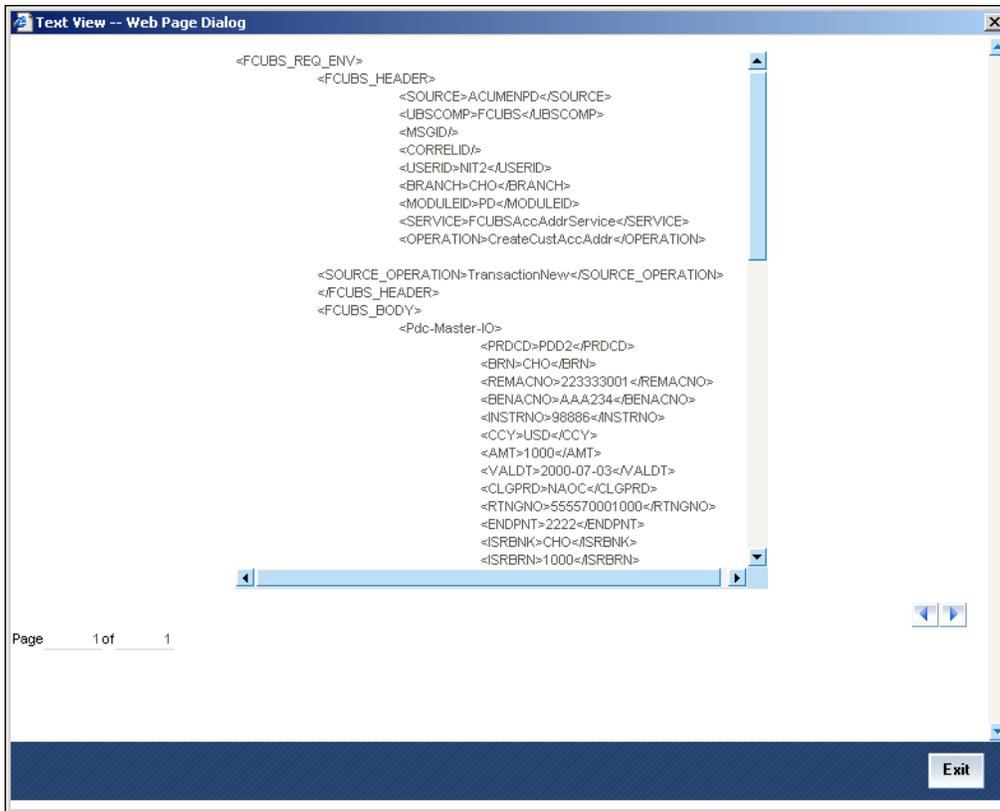
The messages received from the external system will be displayed in the Incoming Message Browser.

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

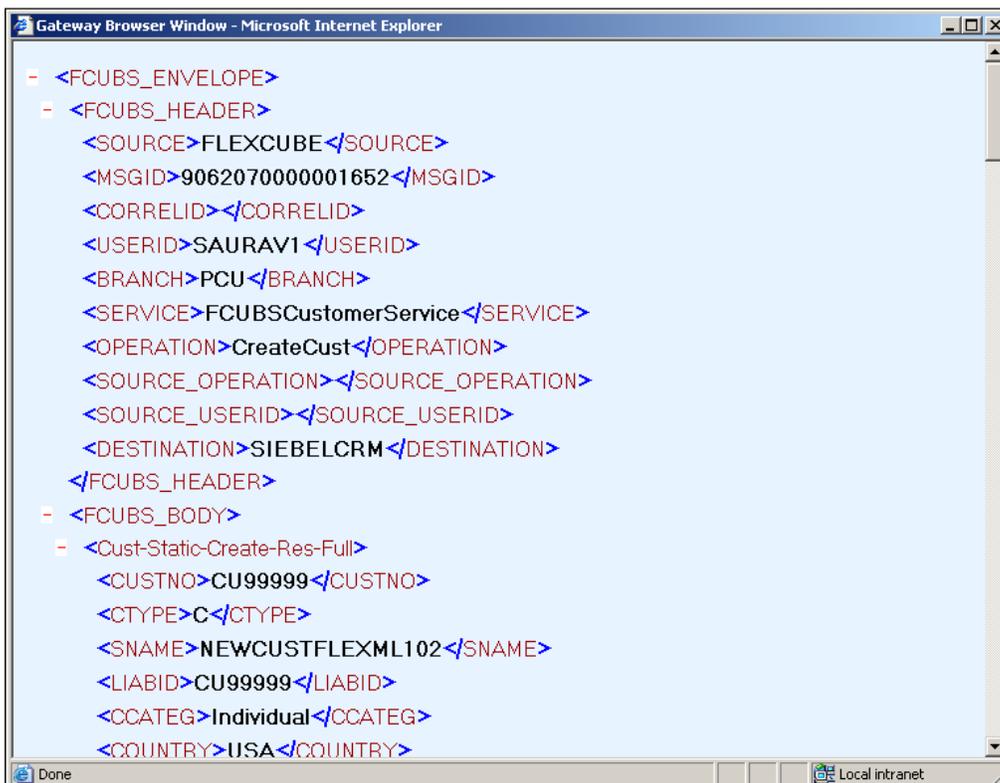


In the 'Incoming Message Browser' screen you can view the details of the messages received from the external systems. You can also view the messages in the XML format or the Text format.

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



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



3.3.1 Viewing Incoming Message Details

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

The screenshot displays the 'Incoming Message Browser - Summary' application window. The window title is 'Incoming Message Browser - Summary'. The interface includes a search form with the following fields: Message Reference Number, External System, Correlation Identification, Service Name, User Identification, Branch Date, Message Status, Branch, Message Identification, Request Queue Message Identification, Operation Code, Their User Identification, Server Date Stamp, and Flexcube Reference Number. Below the form are buttons for Search, Advanced Search, and Reset. A pagination bar shows 'Records per page 15', 'First', 'Previous', '1 Of 1', 'Next', 'Last', and 'Go'. Below the pagination bar is a table header with columns: Message Reference Number, Branch, External System, Message Identification, Correlation Identification, Request Queue Message Identifica. The table body is empty. At the bottom right of the window is an 'Exit' button.

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

Advanced

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

Reset

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

Query

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

Refresh

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

3.4 Duplicate Message Browser

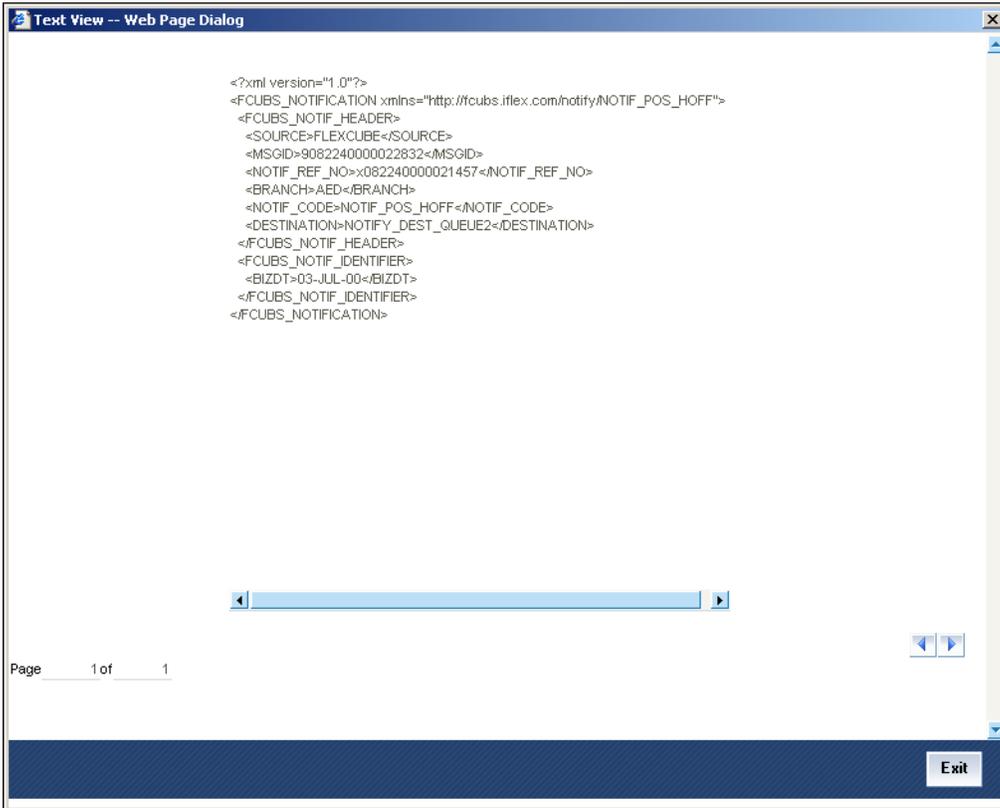
If Oracle FLEXCUBE Integration Gateway identifies a duplicate message it will be sent to the Duplicate Messages Browser. You can invoke the 'Duplicate Message Browser - Summary' screen by typing 'GWSPBROW' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

The screenshot shows a web application window titled "Duplicate Message Browser - Summary". The window contains a search form with the following fields:

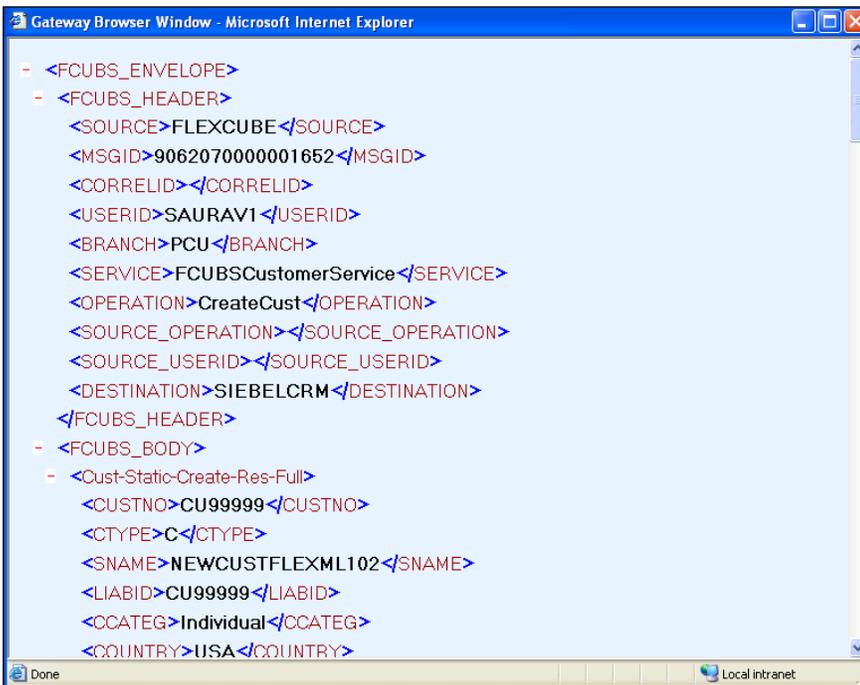
Message Reference Number	<input type="text"/>
External System	<input type="text"/>
Correlation Identification	<input type="text"/>
Service Name	<input type="text"/>
User Identification	<input type="text"/>
Server Date Stamp	<input type="text"/>
Branch	<input type="text"/>
Message Identification	<input type="text"/>
Request Queue Message Identification	<input type="text"/>
Operation Code	<input type="text"/>
Their User Identification	<input type="text"/>
Status	<input type="text"/>

Below the search form, there are buttons for "Search", "Advanced Search", and "Reset". A pagination bar shows "Records per page" set to 15, with "First", "Previous", "1 Of 1", "Next", "Last", and "Go" options. Below the pagination bar is a table header with columns: "Message Reference Number", "Branch", "External System", "Message Identification", "Correlation Identification", and "Request Queue Message Identification". The table body is currently empty. At the bottom right of the window is an "Exit" button.

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



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



3.4.1 Querying Duplicate Message Browser

You can query the list of duplicate messages using the 'Duplicate Message Browser - Summary' screen with functions for search. You can invoke this screen by typing 'GWSPBROW' in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.

The screenshot shows the 'Duplicate Message Browser - Summary' application window. It features a search interface with two columns of input fields for various criteria. The left column includes fields for Message Reference Number, External System, Correlation Identification, Service Name, User Identification, and Server Date Stamp. The right column includes fields for Branch, Message Identification, Request Queue Message Identification, Operation Code, Their User Identification, and Status. Below the search fields are buttons for 'Search', 'Advanced Search', and 'Reset'. A pagination bar indicates 'Records per page 15', 'First', 'Previous', '1 Of 1', 'Next', 'Last', and 'Go'. Below the pagination bar is a table header with columns: 'Message Reference Number', 'Branch', 'External System', 'Message Identification', 'Correlation Identification', and 'Request Queue Message Identifica'. The table body is empty. At the bottom right of the window is an 'Exit' button.

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

Advanced

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

Reset

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

Query

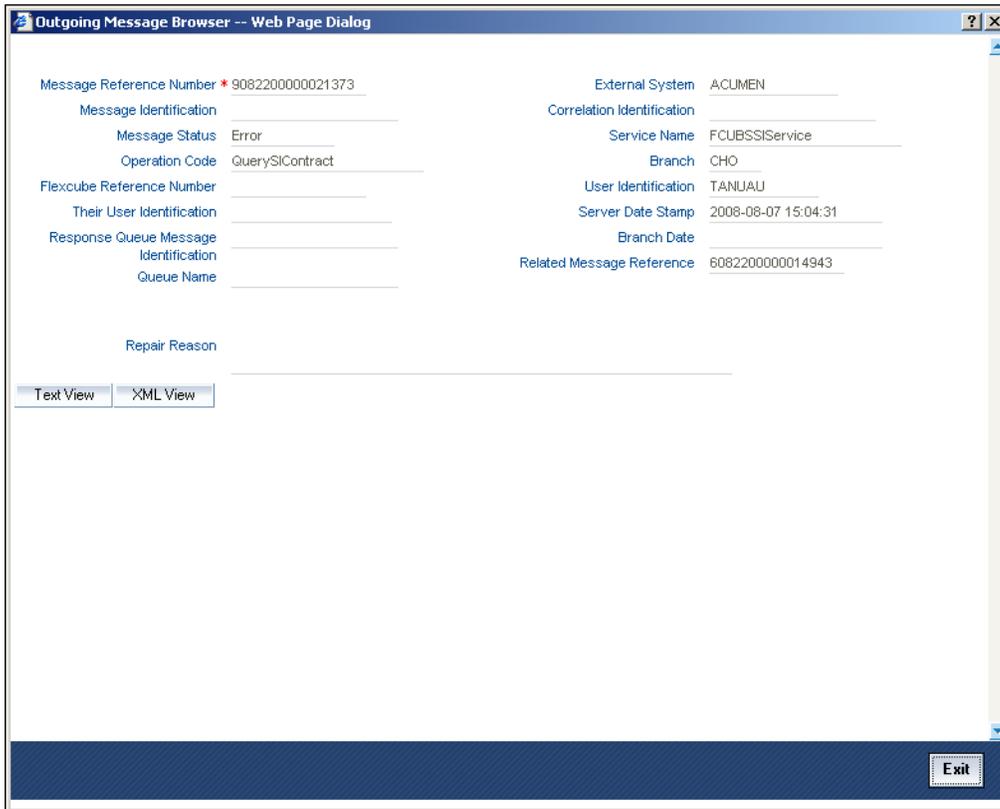
After specifying your search criteria click **Query** to view the list of results which match your search criteria.

Refresh

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

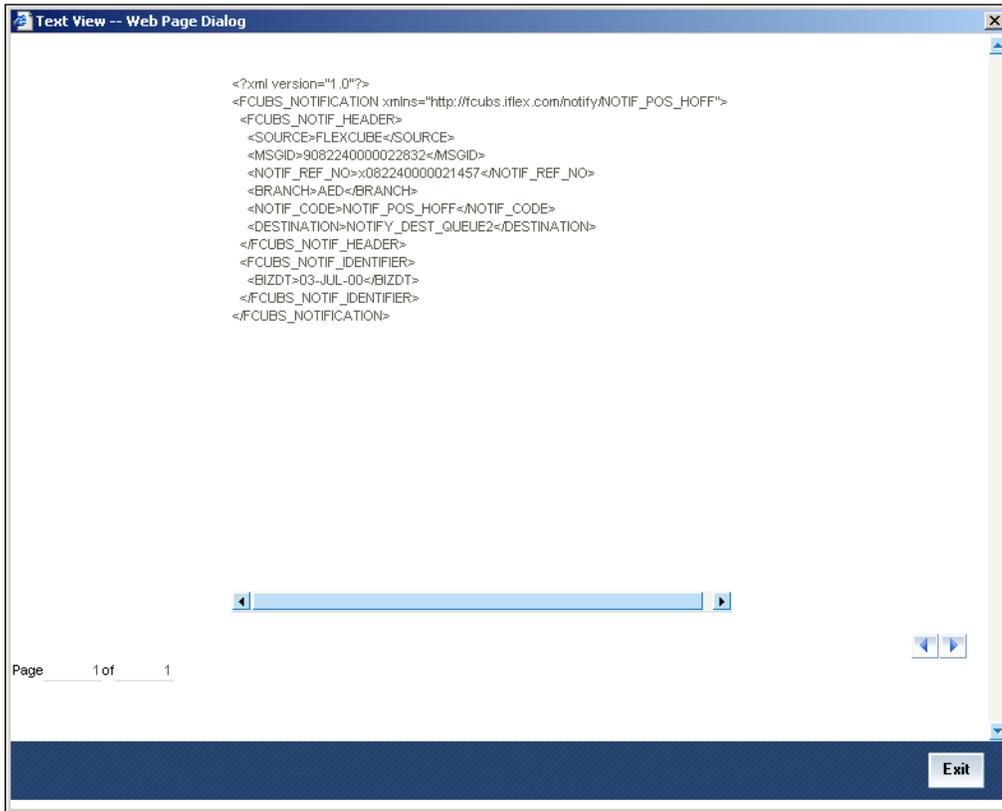
3.5 Outgoing Message Browser

Once the incoming messages have been processed, a response message will be sent to the external systems along with the status of the processed messages. The response messages will be displayed in the 'Outgoing Message Browser'. You can invoke the 'Outgoing Message Browser' screen by typing 'GWDOTBRW' in the field at the top right corner of the Application tool bar and clicking on the adjoining arrow button.

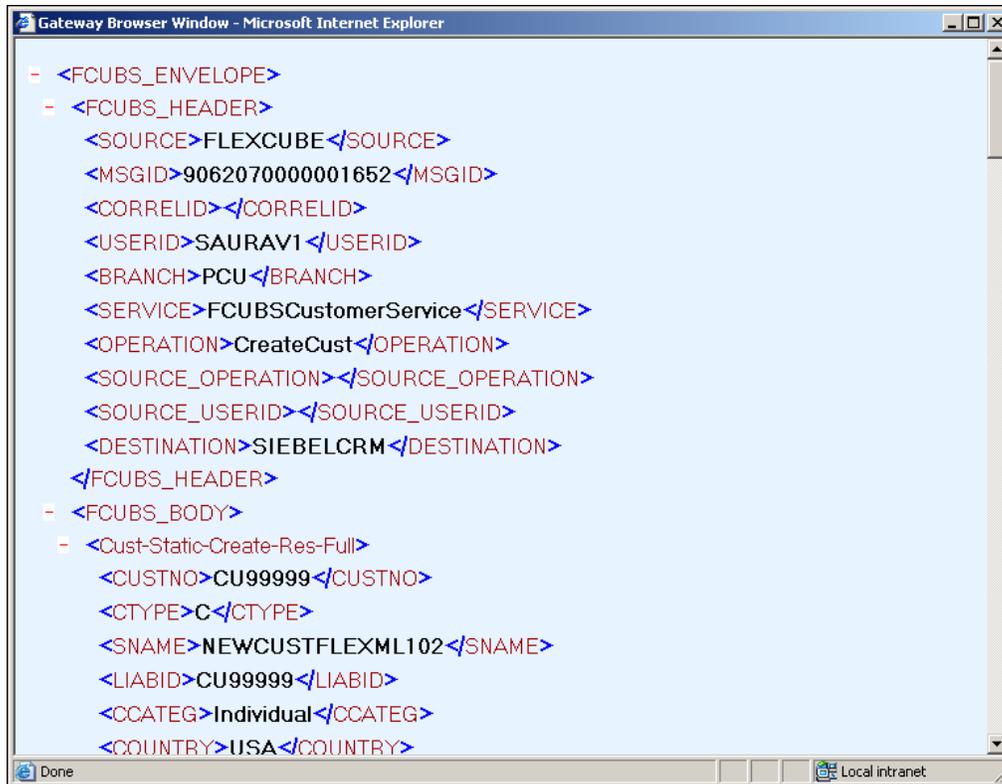


In the 'Outgoing Message Browser' screen you can view the details of the messages sent to the external systems. You can also view the messages in the XML format or the Text format.

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



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



3.5.1 **Querying Outgoing Message Browser**

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

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

Advanced

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

Reset

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

Query

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

Refresh

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

3.6 Defining Notification Messages

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

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

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

The screenshot shows a software window titled "Notifications Installed - Detailed". At the top left is a "Save" button. The main area contains four input fields: "Branch Code *" (a dropdown menu), "Notification Code *" (a dropdown menu), "Branch Name" (a text field), and "Description" (a text field). At the bottom, there is a "Fields" tab and a status bar with the following information: "Input By: TEST12", "Date Time", "Authorized By", "Date Time", "Modification Number", "Authorized" (checkbox), "Open" (checkbox), and a "Cancel" button.

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

Branch Code

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

Notification Code

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

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

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

3.6.1 Viewing Installed Notification Details

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

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

Advanced

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

Reset

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

Query

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

Refresh

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

3.7 Defining the Notifications Enroutes

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

The screenshot shows a software window titled "Notifications Enroute - Detailed". At the top left, there is a "Save" button. The main area is divided into two columns. The left column contains three input fields: "Branch Code *", "Description", and "Destination Name *". The right column contains two input fields: "Notification Code *" and "Description". Below these fields is a "Fields" section with a grid of fields: "Input By TEST12", "Date Time", "Authorized By", "Date Time", "Modification Number", "Authorized" (checkbox), and "Open" (checkbox). A "Cancel" button is located at the bottom right.

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

Branch

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

Notification Code

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

Topic/Queue Name

Define a topic or a queue here. If you define a topic, you can assign multiple notification codes for a branch. In case of a queue, you can define only one notification code for each branch.

3.7.1 Viewing Summary screen

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

The screenshot displays the 'Notifications Enroute - Summary' window. At the top, there are search filters: 'Authorization Status' (dropdown), 'Branch Code' (text input with a search icon), 'Destination Name' (text input with a search icon), 'Record Status' (dropdown), and 'Notification Code' (text input with a search icon). Below these are 'Search', 'Advanced Search', and 'Reset' buttons. A pagination bar shows 'Records per page 15', 'First', 'Previous', '1 Of 1', 'Next', 'Last', and 'Go' buttons. A table header lists the search criteria: 'Authorization Status', 'Record Status', 'Branch Code', 'Notification Code', and 'Destination Name'. At the bottom, there is a legend for 'Authorization Status' (A - Authorized, U - Unauthorized) and 'Record Status' (C - Closed, O - Open). An 'Exit' button is located in the bottom right corner.

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

Advanced

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

Reset

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

Query

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

Refresh

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

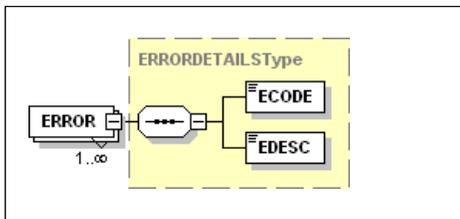
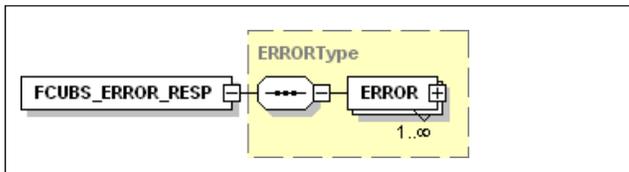
4. Message Formats

4.1 Oracle FLEXCUBE Envelope

A standard message in the Oracle FLEXCUBE Envelope contains two main components namely:

- FCUBS_HEADER – This is a node that forms the header of the FCUBS Envelope. The tags under the header portion of a message will identify a partner system, a service, an operation, user, branch, etc. These tags are constant across all messages
- FCUBS_BODY – This node contains the actual payload of a message. The contents of this node will vary for each message

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



4.1.1.1 FCUBS ERROR RESP

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

ERROR

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

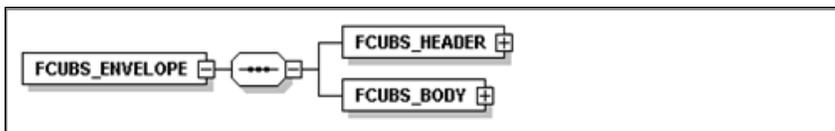
4.1.1.2 FCUBS WARNING RESP

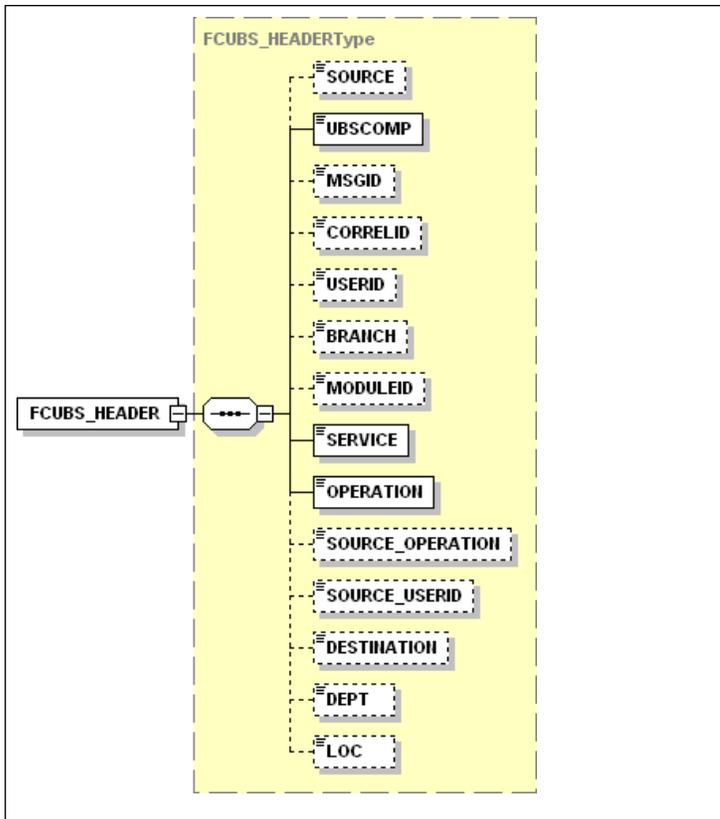
The warning response message will be sent when overrides are raised in a transaction. The Warning response will have another tag 'WARNING' within it.

WARNING

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

A diagrammatic representation of the Oracle FLEXCUBE Envelope is as shown below:





4.1.1 **FCUBS HEADER**

The tags under FCUBS HEADER have been described below:

SOURCE

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

UBSCOMP

This indicates the Oracle FLEXCUBE component of the message - whether FCIS or FCUBS

MSGID

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

CORRELID

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

USERID

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

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

BRANCH

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

MODULEID

This indicates the module ID.

SERVICE

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

OPERATION

This indicates the functional operation.

SOURCE_OPERATION

This indicates the functional operation as registered in Oracle FLEXCUBE.

SOURCE_USERID

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

DESTINATION

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

DEPT

This message indicates the department.

LOC

This indicates the location.

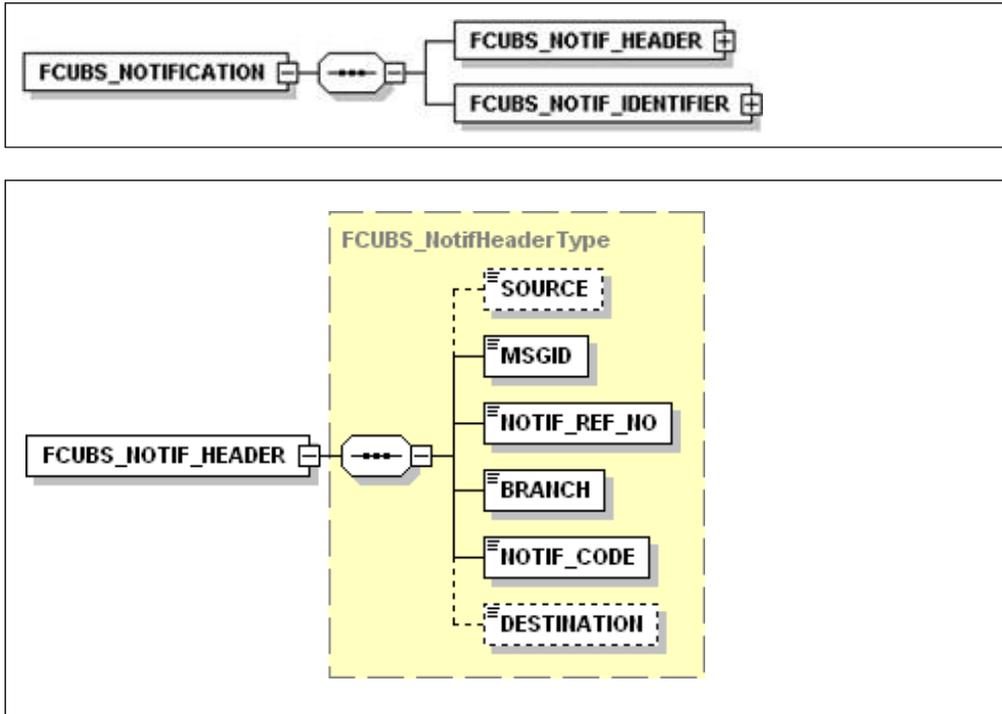
4.2 Oracle FLEXCUBE NOTIFICATION

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

FCUBS_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.

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

A diagrammatic representation of FCUBS NOTIFICATION is as shown below:



4.2.1 FCUBS NOTIFICATION HEADER

The tags under FCUBS NOTIFICATION HEADER have been described below:

SOURCE

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

MSGID

This is the unique reference number generated by Oracle FLEXCUBE.

NOTIF_REF_NO.

This unique reference number identifies each notification message generated in Oracle FLEXCUBE.

BRANCH

This indicates the branch in which notification has been triggered.

NOTIF_CODE

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

DESTINATION

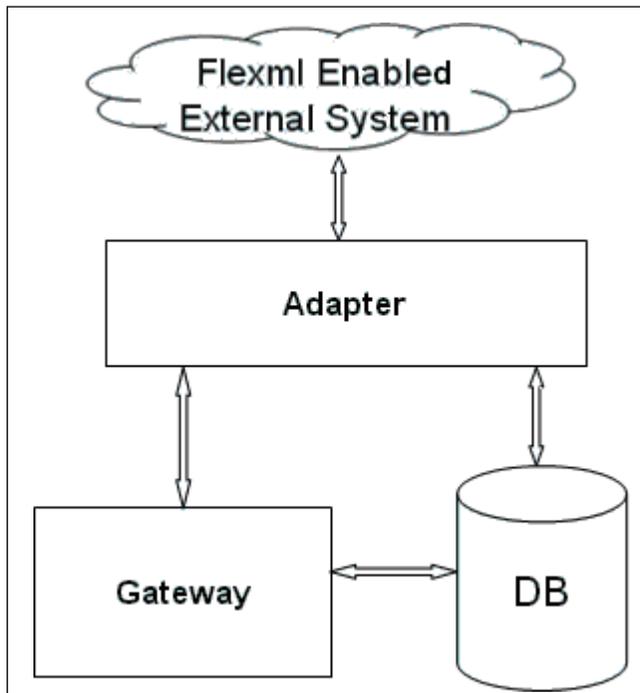
For incoming messages, the DESTINATION should be Oracle FLEXCUBE. For response messages, system will populate the SOURCE of the request message as DESTINATION.

5. FLEXML Adapter Message Conversion

5.1 Introduction

FLEXML Adapter is a component which facilitates the communication between existing FLEXML enabled systems and newer version of Oracle FLEXCUBE based on Gateways. The messages are converted from FLEXML format to Gateway format and vice versa.

The diagram given below indicates that the adapter acts as a middle layer facilitating the exchange of messages between FLEXML based systems and the Oracle FLEXCUBE database.



The different channels used by the adapter to communicate with the gateway can be categorized as follows:

- Inbound FLEXML Adapter
- Outbound FLEXML Adapter

5.2 Inbound FLEXML Adapter

Inbound FLEXML Adapter provides an XML based interface for the FLEXML enabled external system, to communicate with Oracle FLEXCUBE database. It converts the messages from FLEXML format to gateway format and invokes GW EJB, which handles further processing of the message.

The following communication mechanisms are available to process the incoming FLEXML messages:

- EJB Inbound Adapter – This is used if the external system is compatible with EJB interface. The external system invokes the adapter using EJB interface API method
- MDB Inbound Adapter - This is used if the external system is JMS compatible. The external system invokes the adapter by sending message to JMS Queue.

5.2.1 Processing Incoming FLEXML Request Message

The inbound adapter processes an incoming request message in FLEXML format as follows:

1. Adapter receives the request message in FLEXML format
2. The request message is logged in table, if the value of the property file parameter 'FXG_MSG_LOG_FLG' is 'Y'
3. XML message is validated using FLEXML DTD
4. If the message is invalid, the adapter generates parser error message in FLEXML format and sends it back to external system. If the message is valid, the source name, service name, and operation type is retrieved from the message
5. Based on source name, service name and operation type, request and reply XSLT file name is retrieved from database
6. XSLT is applied to convert the message from FLEXML format to GW format
7. GW EJB component is invoked, which processes the request and replies with a message in gateway format
8. XSLT is applied to convert GW reply message to FLEXML format.
9. The reply message is logged in table, if the value of the property file parameter 'FXG_MSG_LOG_FLG' is 'Y'
10. Reply message in FLEXML format is send back to the external system

5.3 Outbound FLEXML Adapter

An outbound adapter notifies any internal events within Oracle FLEXCUBE to the external system. For eg, internal events associated with static maintenances or transactions that happen in Oracle FLEXCUBE need to be communicated to the external system.

FLEXML uses the handoff mechanism to transfer data to the external system, whereas Gateway uses notification process for the same. Handoff messages contain complete transaction information, but notification message contains only the primary key of the transaction.

5.3.1 Processing Outbound Messages

The outbound messages to the external system are processed as follows:

1. Outbound adapter receives request notification XML message from the notification queue
2. Notification code is retrieved from the notification message
3. Using XSLT maintenance table and notification code, query XSLT, handoff XSLT and all the externals system names related to the notification code are retrieved by the adapter
4. Query XSLT is applied to convert GW notification XML message to GW query message and handoff XSLT is applied to convert GW reply to FLEXML format handoff message
5. The handoff queue name is retrieved using external system maintenance
6. The handoff message is logged in table, if the value of the property file parameter 'FXG_MSG_LOG_FLG' is 'Y'

7. Handoff XML message is send to each external system

5.3.2 Mapping FLEXML Format Messages to Gateway Format

FLEXML uses DTDs to define the structure of an XML document, whereas Gateways use XSDs for the same. Extensible Style Sheet Language (XSL), an XML based transformation language, is used to convert XML documents from one format to another.

The mapping can be done using any of the following methods:

- Optional Attribute Map Approach – X-Path language is used to map between the two XML formats
- Mandatory Attribute Map Approach – the mandatory attributes in the target XML document are mapped in this case



Gateway Services
[May] [2022]
Version 11.11.0.0.0

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:
Phone: +91 22 6718 3000
Fax: +91 22 6718 3001
www.oracle.com/financialservices/

Copyright © [2007], [2022], Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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