Oracle® FLEXCUBE Investor Servicing

Gateway Services User Guide

Release 14.7.6.0.0
G30525-01
April 2025

ORACLE

Oracle FLEXCUBE Investor Servicing Gateway Services User Guide, Release 14.7.6.0.0

G30525-01

Contents

Preface

Oracle FLEXCUBE Investor Servicing is a comprehensive mutual funds automation software from Oracle® Financial Servicing Software Ltd.©.

You can use the system to achieve optimum automation of all your mutual fund investor servicing processes, as it provides guidelines for specific tasks, descriptions of various features and processes, and general information.

This topic contains the following sub-topics:

Purpose

You are intended to become familiar with the **Oracle Flexcube Investor Servicing** application through this guide. This guide offers responses to particular features and procedures that are necessary for the module to operate effectively.

Audience

This user guide is intended for the Fund Administrator users and System operators in the AMC.

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 customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

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.

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 or the glossary.
italic	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.

Acronyms and Abbreviations

The list of the acronyms and abbreviations used are as follows:

Table Acronyms and Abbreviations

Abbreviation	Description
CIF	Customer Information File
EOD	End of Day
EPU	Earnings per unit
FCIS	Oracle FLEXCUBE Investor Servicing
FMG	The Fund Manager component of the system
FPADMIN	Oracle FLEXCUBE Administrator
GTA	Global Transfer Agency
ID	Identification
IHPP	Inflation Hedged Pension Plan
IPO	Initial Public Offering
LEP	Life and Endowment Products
LOI	Letter of Intent
NAV	Net Asset Value
REG	The Registrar component of the system
ROA	Rights of Accumulation
ROI	Return on Investment
SI	Standing Instructions
SMS	Security Management System
URL	Uniform Resource Locator
VAT	Value Added Tax
WAUC	Weighted Average Unit Cost

1

Gateway Functions - An Overview

This topic gives an overview on Gateway functions.

An overview on Gateway Functions
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 Investor Servicing** in order to exchange data.

The **Oracle Flexcube Investor Servicing 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 Investor Servicing**.
- Outbound application integration – used when any external system needs to be notified of the various events that occur within **Oracle Flexcube Investor Servicing**.

This topic contains the following sub-topics:

1.1 Inbound Application Integration

This topic describes the inbound application integration.

Inbound Application Integration

Oracle Flexcube Investor Servicing 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 Investor Servicing** and the external systems will be in the form of XML messages. These XML messages are defined in **FCIS** in the form of **XML Schema Documents (XSD)** and are referred to as **FCIS formats**.

FCIS 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 Investor Servicing** EJB Based Synchronous Inbound Application Integration Deployment Pattern
- **Oracle Flexcube Investor Servicing** Web Services Based Synchronous Inbound Application Integration Deployment Pattern
- **Oracle Flexcube Investor Servicing** HTTP Servlet Based Synchronous Inbound Application Integration Deployment Pattern

The **Asynchronous Deployment Pattern** is:

- **Oracle Flexcube Investor Servicing** MDB Based Asynchronous Inbound Application Integration Deployment Patten

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 Investor Servicing** is **EJB literate**, i.e., the external system is capable of interacting with **Oracle Flexcube Investor Servicing** based upon the EJB interface. In this deployment pattern, the external system will use the RMI/IIOP protocol to communicate with the **Oracle Flexcube Investor Servicing** EJB.

In this deployment pattern the EJB displayed by **Oracle Flexcube Investor Servicing** 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 Investor Servicing** 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 Investor Servicing** EJB.

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 Investor Servicing** wants to connect using standards-based, interoperable 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 Investor Servicing** 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 Investor Servicing** web services.

The services displayed by **Oracle Flexcube Investor Servicing** 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 Investor Servicing** 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 Investor Servicing**.

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 Investor Servicing** wants to connect to **Oracle Flexcube Investor Servicing** 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 Investor Servicing** 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 Investor Servicing servlet.

For this deployment pattern, **Oracle Flexcube Investor Servicing** 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 Investor Servicing** servlet. After the necessary processing is done in **Oracle Flexcube Investor Servicing** 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 Investor Servicing**.

MDB Based Asynchronous Deployment Pattern

The MDB deployment pattern is used in integration scenarios where the external system connecting to **Oracle Flexcube Investor Servicing** wants to connect to **Oracle Flexcube Investor Servicing** 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 Investor Servicing**, based on the request, the response is sent to the response queue as an XML message.

1.2 Outbound Application Integration

This topic describes the outbound application integration.

Outbound Application Integration

The Outbound Application Integration is also called the **Oracle Flexcube Investor Servicing** Notify Application Integration layer. This application layer sends out notification messages to the external system whenever events occur in **Oracle Flexcube Investor Servicing**.

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

1.3 Responsibilities of Integration Gateway

This topic describes the responsibilities of Integration Gateway.

The primary responsibilities of **Oracle Flexcube Investor Servicing** Integration Gateway include the following:

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

1.4 Deployment of Oracle Flexcube Investor Servicing Integration Gateway

This topic provides the systematic instructions to deploy **Oracle Flexcube Investor Servicing** Integration Gateway.

Message communication - incoming or outgoing from/to an external system in **Oracle Flexcube Investor Servicing** will happen only through an **Oracle Flexcube Investor Servicing** Integration Gateway. Hence, it becomes the first point of contact or last point of contact with the database in message flow

The **Oracle Flexcube Investor Servicing** Integration Gateway can be deployed to support both the distributed and single schema deployments of **Oracle Flexcube Investor Servicing**:

- **Distributed deployment of FCIS** – In this situation the database components of the Gateway are deployed as two or more schemas. They are:
  - 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 FCIS** – In this situation the database components of the Gateway (messaging and business) are both deployed as part of the single **Oracle Flexcube Investor Servicing** schema.

1.5 Deployment Patterns for Application Integration

This topic provides the systematic instructions to deploy patterns for Application Integration.

Table 1-1 Deployment Patterns for Application Integration

Business Integration Needs	Nature of Integration	Oracle Flexcube Investor Servicing Deployment Pattern	Remarks
Inbound Transactions into Oracle Flexcube Investor Servicing	Synchronous	Oracle Flexcube Investor Servicing EJB	Recommended
		Oracle Flexcube Investor Servicing HTTP Servlet	This can be used if the external system cannot communicate to Oracle Flexcube Investor Servicing using EJB.
		Oracle Flexcube Investor Servicing Web Services	This can be used if the external system chooses to communicate only through Web Services.

Table 1-1 (Cont.) Deployment Patterns for Application Integration

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

2

Message Formats

This topic explains the different message formats of the data exchange between **Oracle Flexcube Investor Servicing** and the external systems.

Introduction

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

- <OPERATION>_<MESSAGE EXCHANGE PATTERN>_REQ
- <OPERATION>_<MESSAGE EXCHANGE PATTERN>_RES, Where <OPERATION> refers to an **Oracle Flexcube Investor Servicing** operation like **CREATEUNITHOLDER**.
- <MESSAGE_EXCHANGE_PATTERN> refers to the request and response message patterns exchanged. The Message Pattern is of the following types:
  - Full Screen (FS) -This indicates that the payload will contain all the elements on the corresponding **Oracle Flexcube Investor Servicing** screen.
  - Input Only (IO) - This indicates that the payload will contain the only input elements on the corresponding**Oracle Flexcube Investor Servicing** screen.
  - Primary Key (PK): This indicates that the payload will contain only the Primary Key elements of the corresponding **Oracle Flexcube Investor Servicing** screen.

<MESSAGE_EXCHNAGE_PATTERN> is a combination of message patterns.

The <MESSAGE_EXCHNAGE_PATTERN> available in **Oracle Flexcube Investor Servicing** is shown below:

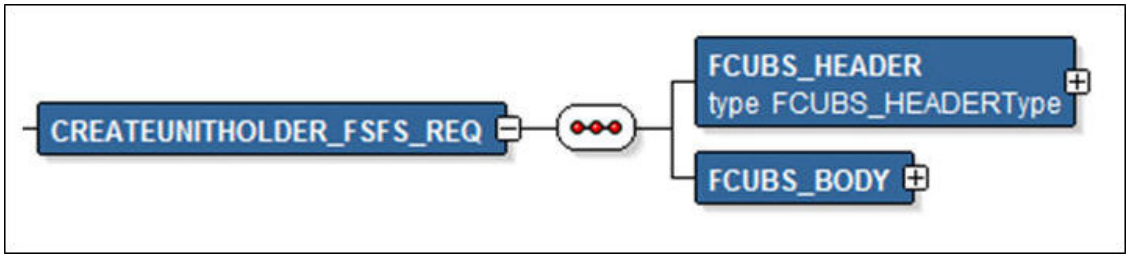
- **FSFS** - Full Screen Request and Full Screen Response
- **IOPK** - Input Only Request and Primary Key Response
- **IOFS** - Input Only Request and Full Screen Response

Non-Query Formats
For instance, consider the Operation **CREATEUNITHOLDER**.

The root node of messages under this operation will be as under:

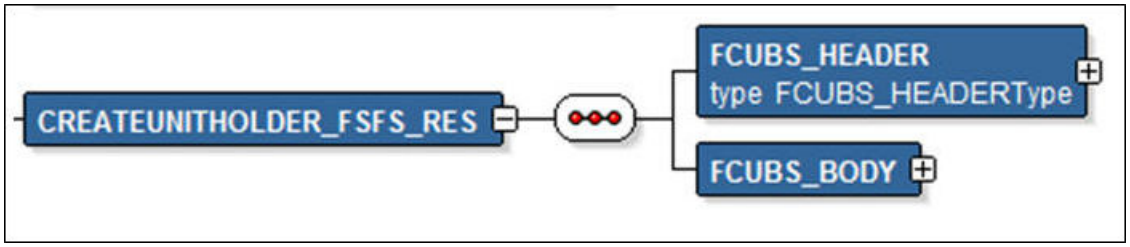
CREATEUNITHOLDER_FSFS_REQ

Figure 2-1 CREATEUNITHOLDER_FSFS_REQ



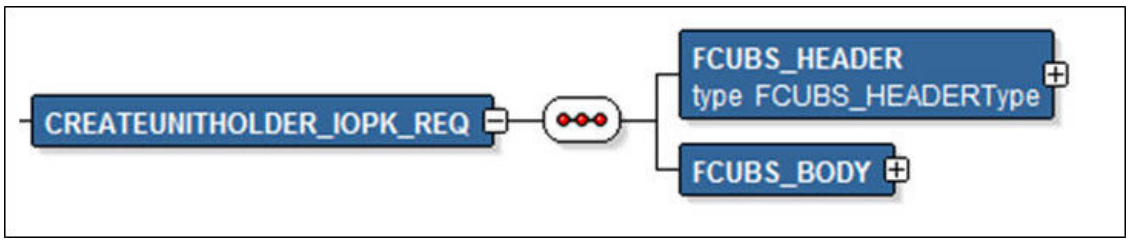
CREATEUNITHOLDER_FSFS_RES

Figure 2-2 CREATEUNITHOLDER_FSFS_RES



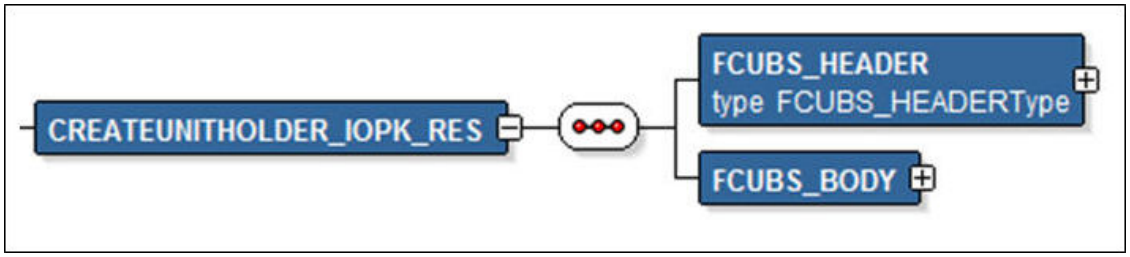
CREATEUNITHOLDER_IOPK_REQ

Figure 2-3 CREATEUNITHOLDER_IOPK_REQ



CREATEUNITHOLDER_IOPK_RES

Figure 2-4 CREATEUNITHOLDER_IOPK_RES

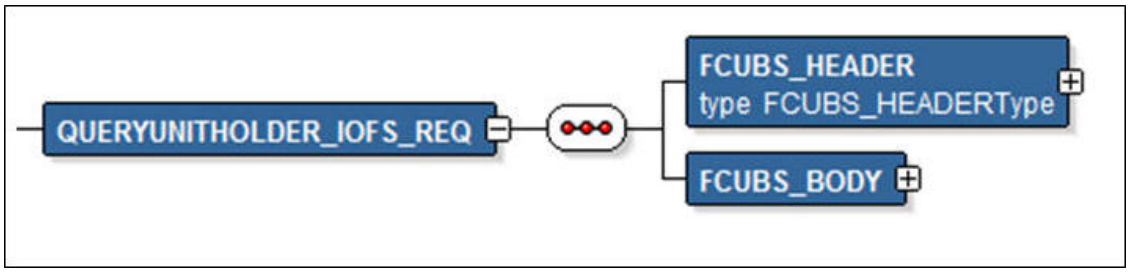


Query Formats
For instance, consider the Operation **QUERYUNITHOLDER**.

The root node of messages under this operation will be as under:

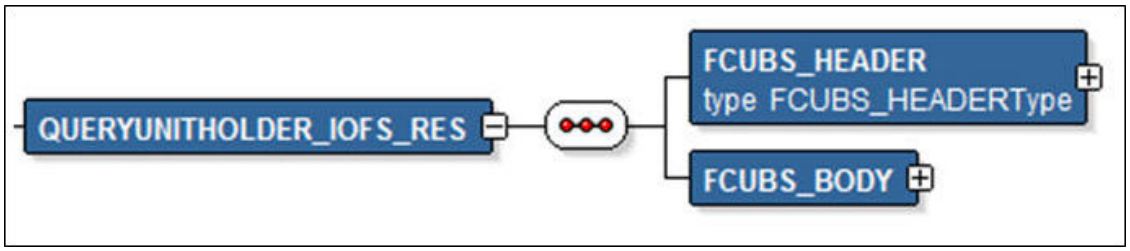
QUERYCUNITHOLDER_IOFS_REQ

Figure 2-5 QUERYCUNITHOLDER_IOFS_REQ



QUERYCUNITHOLDER_IOFS_RES

Figure 2-6 QUERYCUNITHOLDER_IOFS_RES



Oracle Flexcube Investor Servicing Envelope

A standard gateway message in the **Oracle Flexcube Investor Servicing** 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 HEADER
The tags under FCUBS HEADER have been described in below table:

Table 2-1 FCUBS HEADER

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

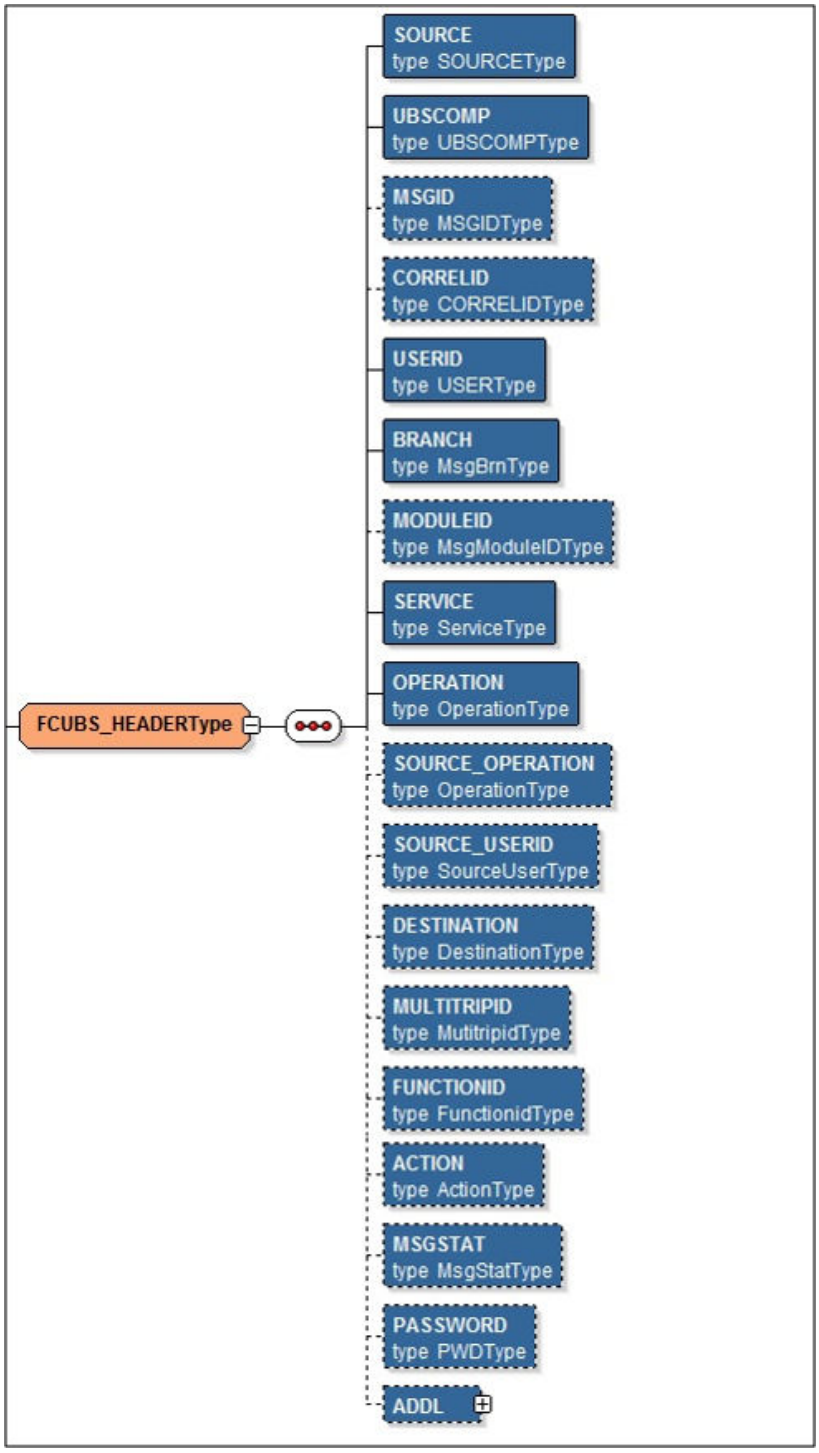
Table 2-1 (Cont.) FCUBS HEADER

Tags	Description
UBSCOMP	This indicates the **Oracle Flexcube Investor Servicing** component of the message - whether **FCIS** or **FCUBS**.
MSGID	This unique ID identifies each message – incoming or outgoing in **Oracle Flexcube Investor Servicing**. Every message will have a distinct message ID
CORRELID	This is the id using which any system which has sent a request to **FCUBS** 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 Investor Servicing** 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 Investor Servicing** will process this request using this id. For response messages, the value of this will be '**null**'.
BRANCH	This indicates the **Oracle Flexcube Investor Servicing** 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 Investor Servicing**. For every incoming message in **Oracle Flexcube Investor Servicing**, the service name is mandatory.
OPERATION	This indicates the functional operation.
SOURCE_OPERATION	This indicates the functional operation as registered in **Oracle Flexcube Investor Servicing**.
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 Investor Servicing**. For response messages, system will populate the **SOURCE** of the request message as **DESTINATION**.
MULTITRIPID	This is a unique id which indicates overrides.
FUNCTIONID	This indicates the **Oracle Flexcube Investor Servicing** Function ID
ACTION	This indicates the action type.
MSGSTAT	This indicates whether the transaction is a `SUCCESS` or `FAILURE`.
PASSWORD	This indicates the password.
ADDL	This is used to send additional parameters i.e. parameters not available in **Oracle Flexcube Investor Servicing**.

FCUBS HEADERType

The image represents the node of FCUBS_HEADERType is as shown below:

Figure 2-7 FCUBS HEADERType

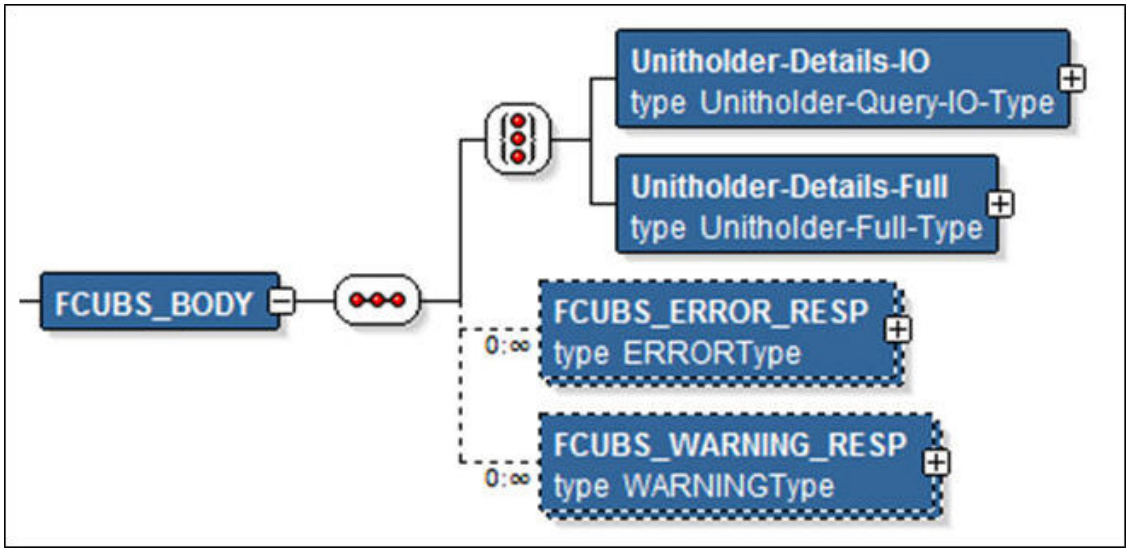


FCUBS BODY

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

The following snapshot shows a sample **FCUBS_BODY of QUERYUNITHOLDER** operation:

Figure 2-8 FCUBS_BODY of QUERYUNITHOLDER

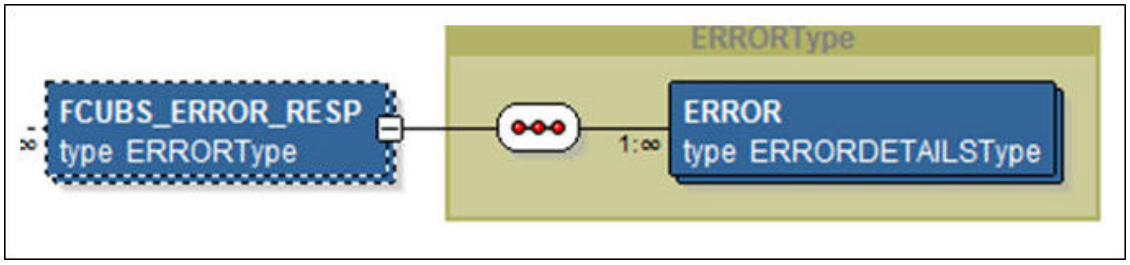


Nodes for Error and Warning response

The **FCUBS_BODY** will contain additional nodes for error and warning responses.

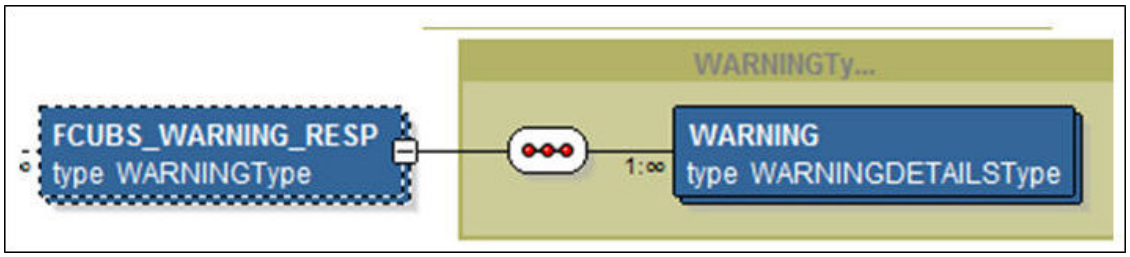
The image represents the node for error response is as shown below:

Figure 2-9 Nodes for Error Response



The image represents the node for warning response is as shown below:

Figure 2-10 Nodes for Warning Response



The nodes for error and warning responses have been described in the below table:

Table 2-2 Nodes for Error and Warning Response

Nodes for Error	Warning Response
FCUBS_ERROR_RESP	The error response message will be sent from **Oracle Flexcube Investor Servicing** 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 **FCIS**.
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 **FCIS**.

FCIS NOTIFICATION

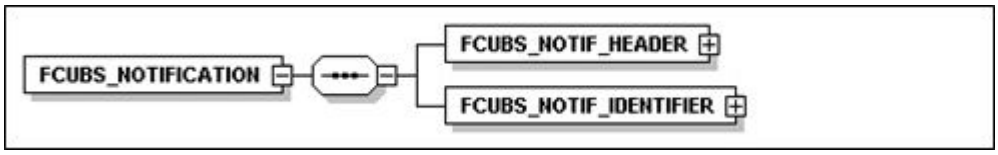
The notification messages are generated in a standard format.

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.

The image represents the node of FCUBS NOTIFICATION is as shown below:

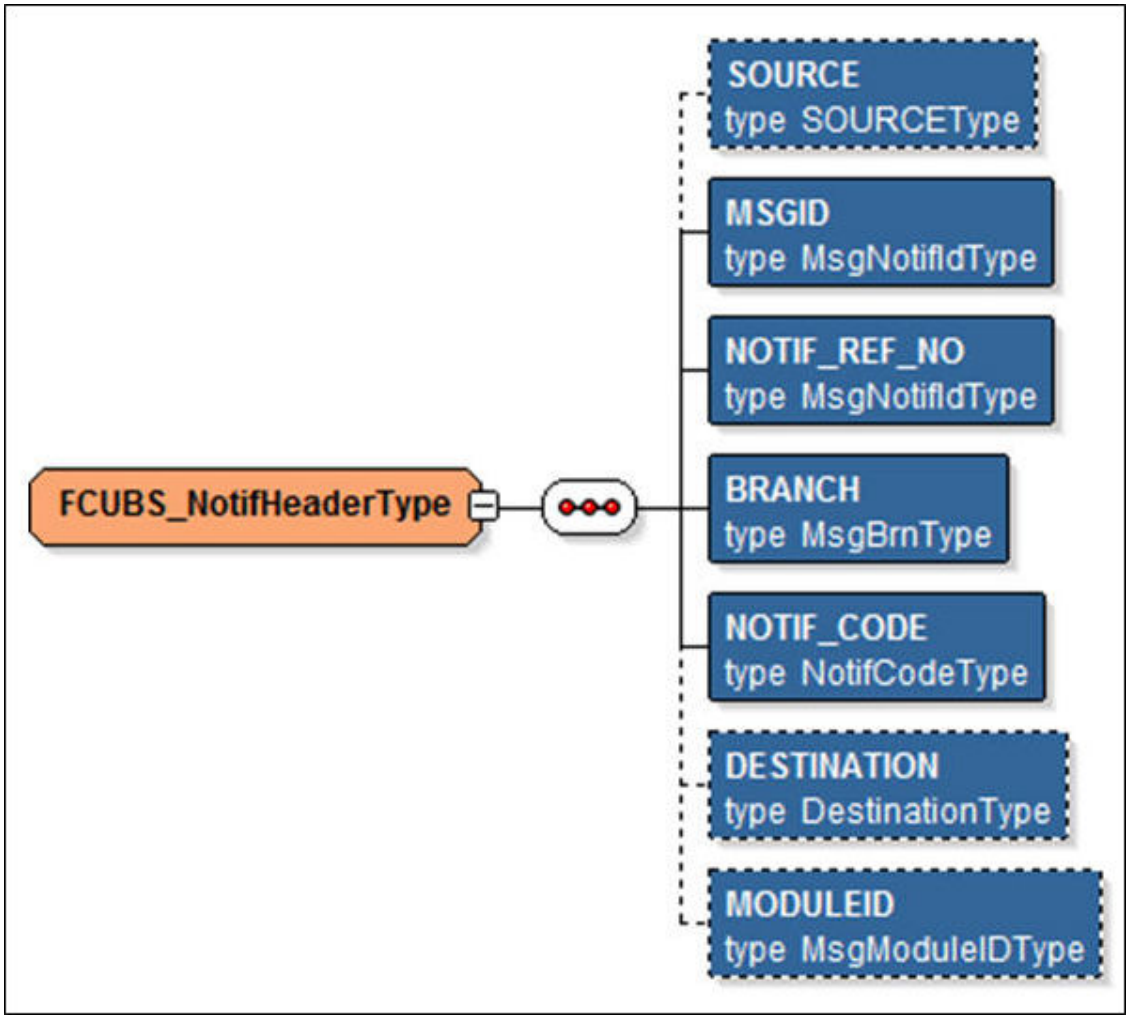
Figure 2-11 FCUBS NOTIFICATION



FCIS NOTIFICATION HEADER

The image represents the node of FCIS NOTIFICATION HEADER is as shown below:

Figure 2-12 FCIS NOTIFICATION HEADER



The tags under FCIS NOTIFICATION HEADER have been described in the below table:

Table 2-3 FCIS NOTIFICATION HEADER

Tags	Description
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 Investor Servicing**.
NOTIF_REF_NO.	This unique reference number identifies each notification message generated in Oracle Flexcube Investor Servicing.
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 Investor Servicing**. For response messages, system will populate the **SOURCE** of the request message as **DESTINATION**.
MODULEID	This indicates the module ID.

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