

SWIFTNet Services Integrator Messaging Hub Interface  
User Guide

## **Oracle FLEXCUBE Universal Banking**

Release 12.1.0.0.0

**Part No. E64763-01**

September 2015

SWIFTNet Services Integrator Messaging Hub Interface User Guide  
September 2015  
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/](http://www.oracle.com/financialservices/)

Copyright © 2007, 2015, Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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

---

# Contents

|   |            |
|---|------------|
| <b>1. Preface .....</b>                               | <b>1-1</b> |
| 1.1 Introduction.....                                 | 1-1        |
| 1.2 Audience.....                                     | 1-1        |
| 1.3 Documentation Accessibility.....                  | 1-1        |
| 1.4 Organization .....                                | 1-1        |
| 1.5 Abbreviations.....                                | 1-1        |
| 1.6 Related Documents .....                           | 1-1        |
| 1.7 Glossary of Icons.....                            | 1-2        |
| <b>2. Oracle FLEXCUBE – FC SSI MH Interface .....</b> | <b>2-1</b> |
| 2.1 Inbound and Outbound Services .....               | 2-1        |
| 2.1.1 <i>Handling Inbound Services</i> .....          | 2-1        |
| 2.1.2 <i>Handling Outbound Services</i> .....         | 2-3        |
| 2.1.3 <i>Processing the ACK/Error File</i> .....      | 2-4        |
| 2.1.4 <i>Viewing Transfer File Status</i> .....       | 2-5        |
| 2.1.5 <i>Message Formats</i> .....                    | 2-6        |

---

# 1.Preface

## 1.1 Introduction

This manual talks about the interface between Oracle FLEXCUBE and Oracle FLEXCUBE SWIFTNet Services Integrator Messaging Hub (FC SSI MH) for SWIFTNet connectivity and SWIFTNet Services.

## 1.2 Audience

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

| Role                          | Function   |
|-------------------------------|--|
| Back office data entry Clerks | Input functions for maintenance related to the interface |
| Back office Managers/Officers | Authorization functions                                  |

## 1.3 Documentation Accessibility

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

## 1.4 Organization

This manual is organized as follows:

| Chapter   | Description   |
|-----------|---|
| Chapter 1 | <i>About this Manual</i> gives information on the intended audience. It also lists the various chapters covered in this User Manual.                        |
| Chapter 2 | <i>Oracle FLEXCUBE – FC SSI MH Interface</i> explains on how Oracle FLEXCUBE communicates with the SWIFT Correspondent via FC SSI MH for transfer of files. |

## 1.5 Abbreviations

The table lists the abbreviations used in this User Manual:

| Abbreviation | Description  |
|--------------|--|
| FC SSI MH    | Oracle FLEXCUBE SWIFTNet Services Integrator Messaging Hub |
| SC           | SWIFT Correspondent  |

## 1.6 Related Documents

- The Procedure User manual

# 1.7 Glossary of Icons

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

| Icons   | Function    |
|---|-------------|
|  | Exit        |
|  | Add row     |
|  | Delete row  |
|  | Option List |

---

## 2. Oracle FLEXCUBE – FC SSI MH Interface

Oracle FLEXCUBE communicates with the SWIFT Correspondent via FC SSI MH for transfer of files. It uses the SWIFT Net protocol 'Fileact' for this dispatch. The mode of communication is folder based with XML Data Layer. The Oracle FLEXCUBE MH adapter will handle the transfer of files from Oracle FLEXCUBE to FC SSI MH and vice versa.

To facilitate the dispatch of files, you have to maintain the following:

- Details of the file to be dispatched: You can maintain this in the 'Dispatch File Parameters' screen.
- Dispatch file generation details: You can maintain this in the 'Dispatch File Generation' screen.

*For more details on dispatch file parameters, refer the section 'Maintaining Dispatch File Parameters' in the 'Maintaining Information specific to the Payments and Collections Module' chapter of the PC User Manual.*

*For more details on dispatch file generation, refer the section 'Generating Dispatch File' in the 'Processing a Payment or Collection Transaction' chapter of the PC User Manual.*

This interface is capacitated to handle to both inbound and outbound services. Each of these services and the corresponding message formats are discussed in the following sections of this User Manual.

This chapter contains the following section:

- [Section 2.1, "Inbound and Outbound Services"](#)

### 2.1 Inbound and Outbound Services

This section contains the following topics:

- [Section 2.1.1, "Handling Inbound Services"](#)
- [Section 2.1.2, "Handling Outbound Services"](#)
- [Section 2.1.3, "Processing the ACK/Error File"](#)
- [Section 2.1.4, "Viewing Transfer File Status"](#)
- [Section 2.1.5, "Message Formats"](#)

#### 2.1.1 Handling Inbound Services

Oracle FLEXCUBE will receive the files from the SWIFT Correspondent (SC) through the FC SSIMH using the folder based communication mode of the Messaging Hub.

The sequence of the message exchanges between Oracle FLEXCUBE SSI MH for file transfers that are initiated from SC to Oracle FLEXCUBE is provided in the table below:

| SWIFTNet Request | Origin | Request/Response Message | Destination | Remarks |
|------------------|--------|--------------------------|-------------|---------|
|------------------|--------|--------------------------|-------------|---------|

|   |                  |                            |                 |  |
|---|------------------|----------------------------|-----------------|--|
| FileAct PUT from SC to Oracle FLEX-CUBE | FC SSI           | AcceptExchangeFileRequest  | Oracle FLEXCUBE | This request will not be processed in Oracle FLEXCUBE and will be configured for Auto Accept in SSI MH |
|   | Oracle FLEX-CUBE | AcceptExchangeFileResponse | FC SSI          | This response will not be Generated in Oracle FLEXCUBE and will be configured as Auto Accept in SSI    |
|   | FC SSI           | IsExchangeFileRequest      | Oracle FLEXCUBE | This message will be processed and the corresponding payload will be processed.                        |
|   | FC SSI           | ErrorInfo                  | Oracle FLEXCUBE | This message will not be processed.  |

You have to maintain the following folder structures in FC SSI MH for Oracle FLEXCUBE as a Business Application for file transfers initiated from SC:

| Folder Name                              | Explanation  |
|--|--|
| <FCC_LINE-ID>/CLIENT/REQ                 | SSI MH Adapter will receive the following in this folder:<br>IsExchangeFileRequest<br>ACKFile<br>ErrorInfo<br>The first two will contain FileAct envelope. |
| <FCC_LINE_ID>/CLIENT/FILEACT/PAYLOAD/PUT | SSI MH Adapter will receive payload for FileAct PUT request initiated by SC in this folder.  |

The sequence of events is listed below:

1. The Oracle FLEXCUBE MH adapter will poll on the envelope messages 'IsExchangeFileRequest' in the 'Client/Request' folder of SSI MH.
2. On receiving the envelope 'IsExchangeFileRequest', the corresponding payload file in the 'Payload' folder is transferred in to Oracle FLEXCUBE Application Server. The system picks up the appropriate payload file based on the logical file name in the envelope message.
3. The payload file from the Oracle FLEXCUBE Application Server will then be moved to the Oracle FLEXCUBE Database Server.
4. The adapter will make a request message 'SEPA-Exchange-File-Req-MSG' with service name 'SEPAFileServices' and operation as 'FileUpload' and place the same on the Oracle FLEXCUBE Gateway MDB queue.
5. The request message 'SEPA-Exchange-File-Req-MSG' will now have the path of the payload file moved to the Database Server.

---

**Note**

The inbound services of the SSI MH Adapter will also process the error file messages and ACK file for the File transfer request initiated by Oracle FLEXCUBE.

---

*For details, refer the section titled 'Handling Outbound Services' later in this chapter.*

## 2.1.2 Handling Outbound Services

The outbound services of adapter will transfer files from the Oracle FLEXCUBE Database Server to FC SSI MH. The file to be transmitted will be created in the Oracle FLEXCUBE Database Server and the process will generate a notification alert to indicate the creation of the file.

The process will then follow the following sequence:

1. On receiving the notification alert from Oracle FLEXCUBE, the File handoff process in SSI MH Adaptor layer will start. The notification alert will have the reference to the file in the database server which needs to be moved to the FC SSI folder.
2. The Handoff file will move from the Database Server to the Application Server from where it will be transferred to the 'Payload' folder in FC SSI MH.
3. After completing the file transfer, the SSI MH adapter will put the envelope XML 'BaExchangeFileRequest message' in the FC SSI MH envelope folder.

The sequence of the message exchanges between the Oracle FLEXCUBE SSI MH for outbound services is given in the table below:

| SWIFTNet Request                               | Origin               | Request/Response Message | Destination | Remarks   |
|--|----------------------|--------------------------|-------------|---|
| FileAct PUT from Oracle Oracle FLEX-CUBE to SC | Business Application | BaExchangeFileRequest    | FC SSI      | This message would be generated by the MH adapter |



| SWIFTNet Request | Origin | Request/Response Message | Destination      | Remarks                                       |
|------------------|--------|--------------------------|------------------|---|
|                  | FC SSI | ErrorInfo*               | Oracle FLEX-CUBE | This message would be processed by MH adapter |
|                  | FC SSI | AckFile                  | Oracle FLEX-CUBE | This message would be processed by MH adapter |

You have to maintain the following folder structures in FC SSI MH for Oracle FLEXCUBE as a Business Application for file transfers initiated from Oracle FLEXCUBE:

| Folder Name                              | Explanation   |
|--|---|
| <FCC_LINE_ID>/SERVER/REQ                 | SSI MH Adapter will put the request file containing 'BaExchangeFileRequest' which will contain FileAct envelope |
| <FCC_LINE_ID>/SERVER/FILEACT/PAYLOAD/GET | SSI MH Adapter will put payload for FileAct PUT request initiated by Oracle FLEXCUBE in this folder.            |
| <FCC_LINE_ID>/SERVER/RESP                | SSI MH Adapter will receive response file containing following in this folder:<br><br>AckFile<br><br>ErrorInfo  |

### 2.1.3 Processing the ACK/Error File

On receiving the ACK/Error file, the adapter will make 'SEPA-ACK-File-Req-MSG' with service name as 'SEPAFILESERVICE'S and operation as 'FILESTATUS UPDATE' and place the same in the MDB queue of Oracle FLEXCUBE gateway.

This request will be further processed as follows:

- If the file transfer to the SC is successful, the file transfer status will be updated to 'Complete' and file will be moved to the Archive folder.
- If the file transfer to SC fails, the file transfer status will be updated to 'Failed'. The reason for the failure will also be mentioned.

## 2.1.4 Viewing Transfer File Status

You can view the status of the file that will be transmitted in the 'Payment Gateway Browser' screen.

The screenshot displays the 'Payment Gateway - SEPA Outgoing File Details' interface. It includes search filters for 'File Reference', 'File Path', 'Failure Reason', 'File Name', and 'File Transfer Status'. Below the filters are 'Search', 'Advanced Search', and 'Reset' buttons. A pagination bar shows 'Records per page: 15', '1 Of 1', and 'Log' button. A table header includes 'File Reference', 'File Name', 'File Path', 'File Transfer Status', and 'Failure Reason'. At the bottom, there are 'Resend' and 'View' buttons, a legend for 'File Transfer Status' (U - Pending, C - Complete, F - Failure, R - Resend), and an 'Exit' button.

The following details are available in this screen:

- Reference number of the file transmitted
- Name of the file
- Path/location of the file
- File Transfer Status: The files will be in any one of the following status at any given point of time:
  - Pending (P)
  - Complete (C)
  - Failure (F)
- Reason for failure

You have the option to resend the files with transfer status as 'Failure'. Click 'Resend' button to initiate the transfer.

## 2.1.5 Message Formats

The message formats are given below:

| <b>BaExchangeFileRequest Field Tag</b>                 | <b>Field Description</b>   | <b>Optional / Mandatory</b> | <b>Restrictions</b>  |
|--|--|-----------------------------|--|
| BaExchangeFileRequest:: Envelope :: TransactionRef     | Transaction reference for a given transaction. Same reference is sent back in response | Mandatory                   |  |
| BaExchangeFileRequest:: Envelope:: LogicalName         | Logical file name  | Mandatory                   | Maximum length = 254   |
| BaExchangeFileRequest:: Envelope:: Requestor           | Application entity. DN of the Requestor  | Optional                    | Maximum length = 100   |
| BaExchangeFileRequest:: Envelope:: Responder           | Responder  | Optional                    | Maximum length = 100   |
| BaExchangeFileRequest:: Envelope:: Service             | SWIFT service name   | Optional                    | Maximum length = 30  |
| BaExchangeFileRequest:: Envelope:: OpType              | FileAct operation type   | Optional                    | Possible values=[PUT, GET]                                     |
| BaExchangeFileRequest:: Envelope:: TransferDescription | Free Text about file transfer  | Optional                    |  |
| BaExchangeFileRequest:: Envelope:: TransferInfo        | Structured data that can be analyzed by the server                                     | Optional                    |  |
| BaExchangeFileRequest:: Envelope:: MsgId               | E2E application identifier of the message.   | Optional                    | Maximum length = 40  |
| BaExchangeFileRequest:: Envelope:: CreationTime        | Request creation time  | Optional                    | Date Format:<br>YYYY-MM-DDTHH:MM:SSZ<br>or YYYY-MM-DDTHH:MM:SS |

| <b>BaExchangeFileRequest Field Tag</b>                  | <b>Field Description</b>  | <b>Optional / Mandatory</b>                | <b>Restrictions</b>  |
|---|---|--|--|
| BaExchangeFileRequest::<br>Envelope::<br>AckIndicator   | The client application indicates to FC SSI to send an acknowledgement of message sent to SWIFTnet.  | Optional<br>The field will be configurable | Possible values=[TRUE, FALSE]  |
| BaExchangeFileRequest::<br>Envelope::<br>RequestCrypto  | SWIFTNet Link will operate signature and encryption automatically on request if this is set to TRUE   | Optional                                   | Possible values=[TRUE, FALSE]<br>Note: In case value of ReqNrIndicator is TRUE, value of RequestCrypto must be set to TRUE |
| BaExchangeFileRequest::<br>Envelope::<br>ReqNrIndicator | TRUE Indicates that non-repudiation is requested for the Request.   | Optional                                   | Possible values=[TRUE, FALSE]  |
| BaExchangeFileRequest::<br>Envelope::<br>DeliveryMode   | Indicates whether store-and-forward (SnF) is used.  | Optional                                   | Possible values=[SnF, REALTIME]  |
| BaExchangeFileRequest::<br>Envelope::<br>DeliveryNotif  | Indicates whether a delivery notification is required   | Optional                                   | Possible values=[TRUE, FALSE]<br>Note: In case value of ReqNrIndicator is TRUE, value of DeliveryNotif must be set to TRUE |
| BaExchangeFileRequest::<br>Envelope::<br>AckResponder   | DN of the Responder that must be used in the header of the acknowledgement of reception of a file to address the server in charge of handling the acknowledgement | Optional                                   | Maximum length = 100   |

| <b>BaExchangeFileRequest<br/>Field Tag</b>               | <b>Field Description</b>                             | <b>Optional /<br/>Mandatory</b> | <b>Restrictions</b>                      |
|--|--|---------------------------------|--|
| BaExchangeFileRequest::<br>Envelope::<br>RequestType     | Request type   | Optional                        | Maximum length<br>= 30                   |
| BaExchangeFileRequest::<br>Envelope::<br>Priority        | The Priority of<br>delivery                          | Optional                        | Possible val-<br>ues=[Urgent,<br>Normal] |
| BaExchangeFileRequest::<br>Envelope::<br>Size            | Size of file if<br>OpType is PUT                     | Optional                        |  |
| BaExchangeFileRequest::<br>Envelope::<br>FileInfo        | User information<br>about the file.                  | Optional                        |  |
| BaExchangeFileRequest::<br>Envelope::<br>MaxSize         | Maximum accept-<br>able Size if the<br>OpType is GET | Optional                        |  |
| BaExchangeFileRequest::<br>Envelope::<br>RequestRef      | Request reference                                    | Optional                        | Maximum length<br>= 30                   |
| BaExchangeFileRequest::<br>Envelope::<br>OrigTransferRef | Origin transfer ref-<br>erence                       | Optional                        |  |
| BaExchangeFileRequest::<br>Envelope::<br>FileDescription | Description of file                                  | Optional                        |  |
| BaExchangeFileRequest::<br>Envelope::<br>PdIndication    | Indicator of possi-<br>ble duplicate of<br>emission  | Optional                        | Possible val-<br>ues=[TRUE,<br>FALSE]    |
| BaExchangeFileRequest::<br>Envelope::<br>PdEmissionTime  | Time of emission<br>of a message                     | Optional                        | Multiple occur-<br>rences possible       |
| BaExchangeFileRequest::<br>Envelope::<br>CompressionReq  | Indicates whether<br>Compression<br>Required or Not  | Optional                        | [TRUE, FALSE]                            |

| <b>BaExchangeFileRequest<br/>Field Tag</b>               | <b>Field Description</b>                              | <b>Optional /<br/>Mandatory</b> | <b>Restrictions</b> |
|--|---|---------------------------------|---------------------|
| BaExchangeFileRequest::<br>Envelope::<br>CompressionAlgo | Indicates which<br>Compression algo<br>has to be used | Optional                        | [ZIP,GZIP,NONE]     |
| BaExchangeFileRequest::<br>Envelope::<br>CompressionAlgo | Indicates which<br>Compression algo<br>has to be used | Optional                        | [ZIP,GZIP,NONE]     |

## Message format for IsExchangeFileRequest

| Field Tag  | Field Description   | Optional / Mandatory | Restrictions  |
|--|---|----------------------|---|
| IsExchangeFileRequest::Envelope ::<br>TransactionRef         | Transaction reference for a given transaction. Same as in request | Mandatory            | Maximum length = 30   |
| IsExchangeFileRequest::Envelope ::<br>TransferRef            | The unique reference of the file transfer                         | Mandatory            | Maximum length = 30   |
| IsExchangeFileRequest::Envelope ::<br>LogicalName            | Logical file name   | Mandatory            | Maximum length = 254  |
| IsExchangeFileRequest::Envelope ::<br>PhysicalName           | Physical file name  | Optional             | Maximum length = 254  |
| IsExchangeFileRequest::Envelope ::<br>Crypto::MemberRef      | Elements that were signed   | Optional             | Multiple occurrences possible                                 |
| IsExchangeFileRequest::Envelope ::<br>Crypto::EncryptDn      | The distinguished name of the decrypter.                          | Optional             | Maximum length = 100  |
| IsExchangeFileRequest::Envelope ::<br>Crypto::Cryptouserinfo | Application-to-application information                            | Optional             |   |
| IsExchangeFileRequest::Envelope ::<br>GblStatus::Severity    | Result of the swCall function call                                | Optional             | Possible values = [Fatal, Transient, Logic, Success, Warning] |
| IsExchangeFileRequest::Envelope ::<br>GblStatus::Code        | Status Code.  | Optional             |   |
| IsExchangeFileRequest::Envelope ::<br>GblStatus::Parameter   | All error Parameters  | Optional             | Multiple occurrences possible                                 |

| <b>Field Tag</b>   | <b>Field Description</b>        | <b>Optional / Mandatory</b> | <b>Restrictions</b>                |
|--|---------------------------------|-----------------------------|------------------------------------|
| IsExchangeFileRequest::<br>Envelope ::<br><br>GblStatus::<br><br>Text                    | Textual description             | Optional                    |                                    |
| IsExchangeFileRequest::<br>Envelope ::<br><br>GblStatus::<br><br>action                  | Proposed correc-<br>tive action | Optional                    |                                    |
| IsExchangeFileRequest::<br>Envelope ::<br><br>GblStatus::<br><br>Details::<br><br>Code   | Status Code                     | Optional                    | Multiple occur-<br>rences possible |
| IsExchangeFileRequest::<br>Envelope ::<br><br>GblStatus::<br><br>Details::<br><br>Text   | Textual description             | Optional                    |                                    |
| IsExchangeFileRequest::<br>Envelope ::<br><br>GblStatus::<br><br>Details::<br><br>Action | Proposed correc-<br>tive action | Optional                    |                                    |



## Message format for ACKFile

| Field Tag                                  | Field Description   | Optional / Mandatory / Required | Restrictions   |
|--|---|---------------------------------|--|
| AckFile::<br>Envelope ::<br>TransactionRef | Transaction reference for a given transaction. Same reference is sent back in response  | Mandatory                       |  |
| AckFile::<br>Envelope ::<br>FcssiRef       | Transaction reference for a given transaction generated in FC SSI.<br><br>Note: This tag is used in case of FileAct Put from Business application to SWIFTNet.                  | Optional                        |  |
| AckFile::<br>Envelope::<br>TransferRef     | The unique reference of the file transfer   | Mandatory                       | Maximum length = 30  |
| AckFile::<br>Envelope::<br>Status          | Indicates the success or failure of the operation   | Mandatory                       | Possible values = [ SUCCESS, FAILURE]                          |
| AckFile::<br>Envelope::<br>TransferAnswer  | Indicates if the file transfer was rejected or accepted by the server   | Optional                        | Possible values=[ Accepted, Rejected]                          |
| AckFile::<br>Envelope::<br>MsgId           | E2E message identifier. Business application can relate the information to its transaction using this identifier as it is the same identifier sent by the business application. | Optional                        | Maximum length = 40  |
| AckFile::<br>Envelope::<br>CreationTime    | Original request creation time  | Optional                        | Date Format:<br>YYYY-MM-DDTHH:MM:SSZ or<br>YYYY-MM-DDTHH:MM:SS |
| AckFile::<br>Envelope::<br>ackdescription  | The answer of the delivery notification. Free Text.   | Optional                        |  |

| <b>Field Tag</b>                                    | <b>Field Description</b>  | <b>Optional / Mandatory / Required</b> | <b>Restrictions</b>   |
|---|---|--|---|
| AckFile::<br>Envelope::<br>RejectDescription        | Indicates why the server application rejected the file transfer | Optional                               |   |
| AckFile::<br>Envelope::<br>RejectInfo               | Structured data about the file transfer rejection               | Optional                               |   |
| AckFile::<br>Envelope::<br>Size                     | File Size   | Optional                               |   |
| AckFile::<br>Envelope::<br>digeststatus             | Digest status computed by FC SSI                                | Optional                               | Possible values = [TRUE, FALSE]                               |
| AckFile::<br>Envelope::<br>GblStatus::<br>Severity  | Result of the swCall function call                              | Optional                               | Possible values = [Fatal, Transient, Logic, Success, Warning] |
| AckFile::<br>Envelope::<br>GblStatus::<br>Code      | Status Code.  | Optional                               |   |
| AckFile::<br>Envelope::<br>GblStatus::<br>Parameter | All error Parameters  | Optional                               | Multiple occurrences possible                                 |
| AckFile::<br>Envelope::<br>GblStatus::<br>Text      | Textual description   | Optional                               |   |
| AckFile::<br>Envelope::<br>GblStatus::<br>action    | Proposed corrective action                                      | Optional                               |   |

| <b>Field Tag</b>  | <b>Field Description</b>                    | <b>Optional / Mandatory / Required</b> | <b>Restrictions</b>           |
|---|---|--|-------------------------------|
| AckFile::<br>Envelope::<br>GblStatus::<br>Details::<br>Code   | Status Code                                 | Optional                               | Multiple occurrences possible |
| AckFile::<br>Envelope::<br>GblStatus::<br>Details::<br>Text   | Text description                            | Optional                               |                               |
| AckFile::<br>Envelope::<br>GblStatus::<br>Details::<br>Action | Proposed corrective action                  | Optional                               |                               |
| AckFile::<br>Envelope::<br>Responder                          | Responder                                   | Optional                               | Maximum length = 100          |
| AckFile::<br>Envelope::<br>ResponseRef                        | Response Ref                                | Optional                               | Maximum length = 30           |
| AckFile::<br>Envelope::<br>PdIndication                       | Indicator of possible duplicate of emission | Optional                               | Possible values=[TRUE, FALSE] |
| AckFile::<br>Envelope::<br>PdEmissionTime                     | Time of emission of a message               | Optional                               | Multiple occurrences possible |

### Message format for Error File

| Field Tag                    | Field Description  | Optional / Mandatory/ Required | Restrictions                  |
|------------------------------|--|--------------------------------|-------------------------------|
| ErrorInfo:: TransactionRef   | Transaction reference for a given transaction. Same reference is sent back in response   | Mandatory                      |                               |
| ErrorInfo:: FcssiRef         | Transaction reference for a given transaction generated by FC SSI.   | Optional                       | Maxlength=30                  |
| ErrorInfo:: IdMsg            | In case of MQ Interface, this should give the MQ Message ID of the associated message. For Folder Interface this field should contain the name of the associated message file. | Optional                       |                               |
| ErrorInfo:: Description      | Text description of the error to the possible detailed level   | Mandatory                      |                               |
| ErrorInfo:: DuplicationError | Flag indicating a duplication error  | Optional                       | Possible values=[true, false] |