

Web Service Development
Oracle FLEXCUBE Investor Servicing
Release 14.5.0.0.0
[May] [2021]



Table of Contents

1. PREFACE	1-1
1.1 AUDIENCE	1-1
1.2 RELATED DOCUMENTS.....	1-1
1.3 CONVENTIONS	1-1
1.4 HYPOTHETICAL EXAMPLE USED IN THE DOCUMENT	1-2
2. INTRODUCTION	2-1
2.1 HOW TO USE THIS GUIDE	2-1
3. WEB SERVICE - INTRODUCTION	3-1
3.1 COMPONENTS OF WEB SERVICE	3-1
3.2 FLEXCUBE SOAP FORMAT.....	3-2
3.3 FLEXCUBE WSDL FORMAT.....	3-5
3.3.1 WSDL parts.....	3-5
3.3.2 FCUBS WSDL example	3-6
3.4 FLEXCUBE WEB SERVICE DATA FLOW	3-9
4. PRE REQUISITES FOR WEB SERVICE DEVELOPMENT	4-1
5. WEB SERVICE DEVELOPMENT	5-1
5.1 WEB SERVICE SPECIFICATION.....	5-1
5.2 GENERATE / DEVELOP XSDS.....	5-2
5.2.1 Open Development generated.....	5-2
5.2.2 Hand coded.....	5-2
6. ANNEXURE A – FCMOBILESERVICE – SAMPLE FILES	6-7
6.1 XSD FILES	6-7
6.1.1 CS-Mobil-Types.xsd.....	6-7
6.1.2 CS-CreateMobil-Req-Full-MSG.xsd.....	6-18
6.1.3 CS-CreateMobil-Req-IO-MSG.xsd	6-19
6.1.4 CS-CreateMobil-Res-Full-MSG.xsd	6-20
6.1.5 CS-CreateMobil-Res-PK-MSG.xsd.....	6-21
6.2 WSDL FILE	6-22
6.2.1 FCMobileService.wsdl.....	6-22
6.3 IMPLEMENTATION JAVA FILES	6-25
6.3.1 FCMobileServiceImpl.java	6-25
6.4 GATEWAY STATIC DATA FROM OPEN DEVELOPMENT.....	6-35
6.4.1 GWTB_AMEND_FIELDS__JKDMOBIL.INC.....	6-35
6.4.2 GWTB_AMEND_NODES__JKDMOBIL.INC.....	6-35
6.4.3 GWTM_AMEND_FIELDS__JKDMOBIL.INC.....	6-36
6.4.4 GWTM_AMEND_MASTER__JKDMOBIL.INC	6-36
6.4.5 GWTM_AMEND_NODES__JKDMOBIL.INC	6-37

1. Preface

This document describes the concepts and helps reader to develop Web service for FLEXCUBE IS function IDs. Web Services developed for FLEXCUBE IS are deployed on Gateway framework.

1.1 Audience

This Web service Development book is intended for FLEXCUBE Application Developers/Users who perform the following tasks:

- Develop the new web service for existing FLEXCUBE Function IDs
- Modify the existing web service to add/remove fields

1.2 Related documents

For more information on FLEXCUBE framework, refer the below documents:

- Development Overview Guide
- Getting Started

To Use this manual, you need conceptual and working knowledge of the below:


Proficiency	Resources
FLEXCUBE IS Development overview	Development Overview Guide
OPEN DEVELOPMENT function ID development getting started	Getting Started
Open Development screen development	Function ID Development

1.3 Conventions

The following text conventions are used in this document:

Convention Meaning

boldface	Boldface type indicates graphical user interface elements (for example, menus and menu items, buttons, tabs, dialog controls), including options that you select.
<i>italic</i>	italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates language and syntax elements, directory and File name, URLs, text that appears on the screen, or text that

	you enter.
	Indicates important information

1.4 Hypothetical Example used in the document

FLEXCUBE IS has the function ID JKDMOBILE. This is a maintenance function ID. This function ID is used map the customer number with mobile number. This document covers the web service development for this function ID.

2. Introduction

2.1 How to use this Guide

The information in this guide includes:

- [Chapter 2, “Introduction”](#)
This is an introduction section.
- [Chapter 3, “Web Service - Introduction”](#)
This section discuss the Web service introduction and FLEXCUBE IS web service concepts
- [Chapter 4, “Pre requisites for Web service development”](#)
This section discuss the pre-request check list for web service development
- [Chapter 5, “Web services Development”](#)
This section discusses steps for web service development and testing.
- [Chapter 6, “Annexure A – FCMobileService sample files”](#)
This section discuss the sample files

3. Web Service - Introduction

Web Services are self-contained, self-describing, modular applications that can be published, located and invoked across the web. A Web Service is an interface that describes a collection of operations that are network accessible through standardized XML messaging. The technology is called “Web service” because it integrates services(applications) using web technologies(the internet and its standards). Web service plays vital role in creating Service Oriented Architecture based software applications.

Web services use the following technologies

- XML
XML is used to describe the web service specifications
- SOAP
Simple Object Access Protocol is a network, transport, and programming language, platform independent protocol that allows client to call a remote service. The message format is XML
- WSDL
Web Service Description Language is an XML based interface and implementation description language The service provider uses WSDL document in order to specify the operations of web service and parameters and data types of these operations

3.1 Components of web service

Following are the components of FLEXCUBE IS Web service:

- XSD files
 - Message XSDs
This represents the “Message” portion on WSDL file that used to define the message communicated in web service. A message can have one or more data types.
 - Type XSDs
This represents the “Types” portion on WSDL file that used to define the data types that are part of messages.
- WSDL file
- Implementation Java files
- Gateway static data

Example:

JKDMOBILE function ID – web service development results in following files. Refer the annexure A for these file contents

- XSD files
 - CS-Mobil-Types.xsd
 - CS-CreateMobil-Req-Full-MSG.xsd
 - CS-CreateMobil-Req-IO-MSG.xsd
 - CS-CreateMobil-Res-Full-MSG.xsd
 - CS-CreateMobil-Res-PK-MSG.xsd
- WSDL file
 - FCMobileService.wsdl
- Implementation Java files
 - FCMobileServiceImpl.java
- Gateway static data
 - GWTB_AMEND_FIELDS__JKDMOBIL.INC
 - GWTB_AMEND_NODES__JKDMOBIL.INC
 - GWTM_AMEND_FIELDS__JKDMOBIL.INC
 - GWTM_AMEND_MASTER__JKDMOBIL.INC
 - GWTM_AMEND_NODES__JKDMOBIL.INC

3.2 **FLEXCUBE SOAP format**

FLEXCUBE IS uses the following SOAP format

SOAP component	FCUBS component	Example
SOAP Envelope	OPERATION.FUNCTIONID_PATTERN_REQ/RESP	CREATEMOBILE_FSFS_REQ
SOAP Header	<FCUBS_HEADER>	<FCUBS_HEADER>
SOAP Body	<FCUBS_BODY>	<FCUBS_BODY>

Example:

```
<CREATEMOBILE_FSFS_REQ xmlns="http://fcubs.ofss.com/service/FCMobileService">
  <FCUBS_HEADER>
    <SOURCE>FLEXCUBE </SOURCE>
    <UBSCOMP>FCIS</UBSCOMP>
    <MSGID></MSGID>
    <CORRELID></CORRELID>
    <USERID>PIYUSHB</USERID>
    <BRANCH>000</BRANCH>
    <MODULEID>CS</MODULEID>
    <SERVICE>FCMobileService</SERVICE>
```

```

<OPERATION>CreateMobile</OPERATION>

<SOURCE_OPERATION></SOURCE_OPERATION>

<SOURCE_USERID></SOURCE_USERID>

<!--Optional:-->

<DESTINATION></DESTINATION>

<!--Optional:-->

<MULTITRIPID></MULTITRIPID>

<!--Optional:-->

<FUNCTIONID></FUNCTIONID>

<!--Optional:-->

<ACTION></ACTION>

<!--Optional:-->

<MSGSTAT>SUCCESS</MSGSTAT>

<!--Optional:-->

<ADDL>

  <!--Zero or more repetitions:-->

  <PARAM>

    <NAME>abc</NAME>

    <VALUE>123</VALUE>

  </PARAM>

</ADDL>

</FCUBS_HEADER>

<FCUBS_BODY>

<Jktm-Junk-Mobile-Full>

  <CUSTOMER_NO>000000001</CUSTOMER_NO>

  <CELL_NO>1234567890</CELL_NO>

  <!--Optional:-->

  <MAKER></MAKER>

  <!--Optional:-->

  <MAKERSTAMP></MAKERSTAMP>

```



```

<!--Optional:-->
<CHECKER></CHECKER>
<!--Optional:-->
<CHECKERSTAMP></CHECKERSTAMP>
<!--Optional:-->
<MODNO></MODNO>
<!--Optional:-->
<TXNSTAT></TXNSTAT>
<!--Optional:-->
<AUTHSTAT></AUTHSTAT>
<!--Zero or more repetitions:-->
<Jktm-Junk-Mobil-Dtl>
  <CUSTOMER_NO>010101</CUSTOMER_NO>
  <!--Optional:-->
  <ACCOUNT_NO></ACCOUNT_NO>
  <!--Optional:-->
  <NAME></NAME>
  <!--Optional:-->
  <TXNDETL></TXNDETL>
  <!--Optional:-->
  <STATUS></STATUS>
  <!--Optional:-->
  <CELL_NO></CELL_NO>
</Jktm-Junk-Mobil-Dtl>
<!--Optional:-->
<Jktm-Junk-Mobile-Dtl2>
  <CUSTOMER_NO></CUSTOMER_NO>
  <!--Optional:-->
  <CUSTOMER_NAME></CUSTOMER_NAME>
  <!--Optional:-->

```

```

    <CUSTOMER_CATEGORY></CUSTOMER_CATEGORY>

    <!--Optional:-->

    <ADDRESS></ADDRESS>

    <!--Optional:-->

    <CITY></CITY>

    <!--Optional:-->

    <COUNTRY></COUNTRY>

    </Jktm-Junk-Mobile-Dtl2>

    </Jktm-Junk-Mobile-Full>

  </FCUBS_BODY>
</CREATEMOBILE_FSFS_REQ>

```



In FCUBS BODY, the data blocks are referred as nodes. Example, *Jktm-Junk-Mobile-Full* in above XML represents the data block that sources from *JKTM_JUNK_MOBILE_FULL* table. Replace “-”(hyphen) with “_”(underscore) to get the table name and data block.

3.3 FLEXCUBE WSDL format

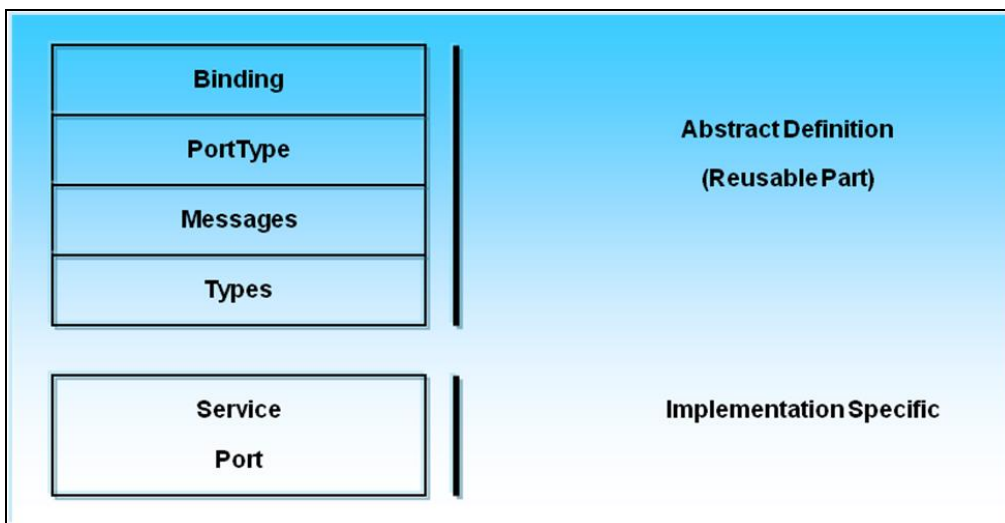
FLEXCUBE has the concept of Function ID that has multiple actions. This is translated to Web service pattern as below:

FUNCTION ID + ACTIONS = WEB SERVICE + OPERATIONS

Example

JKDMOBILE + NEW = FCMobileService + CreateMobile

3.3.1 WSDL parts



3.3.2 FCUBS WSDL example

```
<?xml version= "1.0" encoding="UTF-8"?>

<wsdl:definitions name="FCMobileService"

targetNamespace = "http://types.ws.gw.fcubs.ofss.com"

xmlns:tns = "http://types.ws.gw.fcubs.ofss.com"

xmlns:wsdlssoap = "http://schemas.xmlsoap.org/wsdlssoap/"

xmlns:wsdls = "http://schemas.xmlsoap.org/wsdls/"

xmlns:xs = "http://www.w3.org/2001/XMLSchema"

xmlns="http://schemas.xmlsoap.org/wsdls/"

xmlns:ns="http://fcubs.ofss.com/service/FCMobileService">

<wsdl:types>

  <xs:schema>

    <xs:import namespace="http://fcubs.ofss.com/service/FCMobileService"

      schemaLocation="../XSD/CS-CreateMobil-Req-Full-MSG.xsd"/>

  </xs:schema>

  <xs:schema>

    <xs:import namespace="http://fcubs.ofss.com/service/FCMobileService"

      schemaLocation="../XSD/CS-CreateMobil-Res-Full-MSG.xsd"/>

  </xs:schema>

  <xs:schema>

    <xs:import namespace="http://fcubs.ofss.com/service/FCMobileService"

      schemaLocation="../XSD/CS-CreateMobil-Req-IO-MSG.xsd"/>

  </xs:schema>

  <xs:schema>

    <xs:import namespace="http://fcubs.ofss.com/service/FCMobileService"

      schemaLocation="../XSD/CS-CreateMobil-Res-PK-MSG.xsd"/>

  </xs:schema>

</wsdl:types>
```

```

<wsdl:message name="CreateMobileFSReq">
    <wsdl:part name="RequestMsg" element="ns:CREATEMOBILE_FSFS_REQ"/>
</wsdl:message>

<wsdl:message name="CreateMobileFSRes">
    <wsdl:part name="ResponseMsg" element="ns:CREATEMOBILE_FSFS_RES"/>
</wsdl:message>

<wsdl:message name="CreateMobileIOReq">
    <wsdl:part name="RequestMsg" element="ns:CREATEMOBILE_IOPK_REQ"/>
</wsdl:message>

<wsdl:message name="CreateMobilePKRes">
    <wsdl:part name="ResponseMsg" element="ns:CREATEMOBILE_IOPK_RES"/>
</wsdl:message>

<wsdl:portType name="FCMobileServiceSEI" >
    <wsdl:operation name="CreateMobileIO">
        <wsdl:input message="tns:CreateMobileIOReq"/>
        <wsdl:output message="tns:CreateMobilePKRes"/>
    </wsdl:operation>

    <wsdl:operation name="CreateMobileFS">
        <wsdl:input message="tns:CreateMobileFSReq"/>
        <wsdl:output message="tns:CreateMobileFSRes"/>
    </wsdl:operation>

</wsdl:portType>

```

```

<wsdl:binding name="FCMobileService" type="tns:FCMobileServiceSEI">
  <wsdlsoap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="CreateMobileFS">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input>
      <wsdlsoap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <wsdlsoap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>

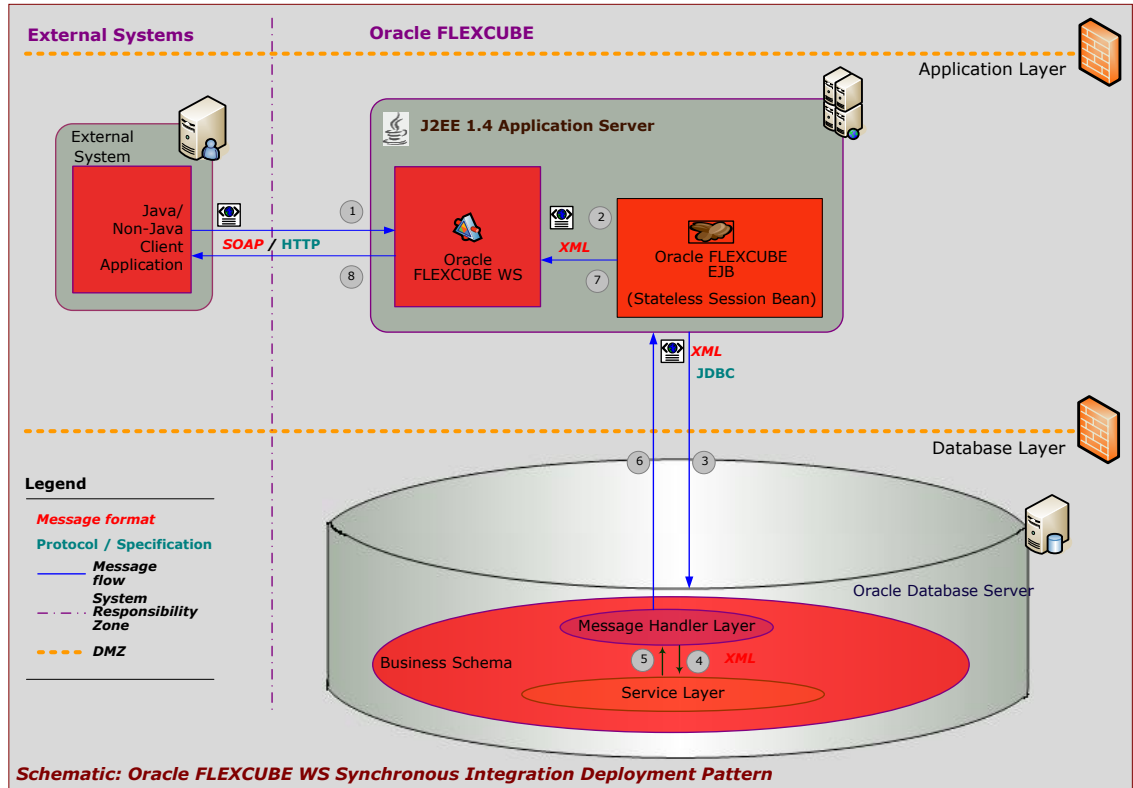
  <wsdl:operation name="CreateMobileIO">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input>
      <wsdlsoap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <wsdlsoap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>

  <wsdl:service name="FCMobileService">
    <wsdl:port name="FCMobileServiceSEI" binding="tns:FCMobileService">
      <wsdlsoap:address
location="http://localhost:9080/FCMobileService/services/FCMobileService"/>
    </wsdl:port>
  </wsdl:service></wsdl:definitions>

```

3.4 FLEXCUBE Web Service data flow

The below diagram explains the web service message flow in FLEXCUBE IS.



1. Consumer sends web service request as per the WSDL definition implemented.
2. FLEXCUBE IS Gateway(that deploys the FLEXCUBE IS Web service) received the request and forwards the request to FLEXCUBE IS EJB
3. FLEXCUBE EJB forwards the request to Database Message handler using JDBC protocol
4. Message handler identifies the Service and calls corresponding operation handler for processing
5. The response message is sent back to Message handler
6. Message handler forward the response message back to FCUBS EJB
7. FCUBS EJB sends the response back to FLEXCUBE Gateway web service
8. FLEXCUBE Gateway web service responds to the Consumer.

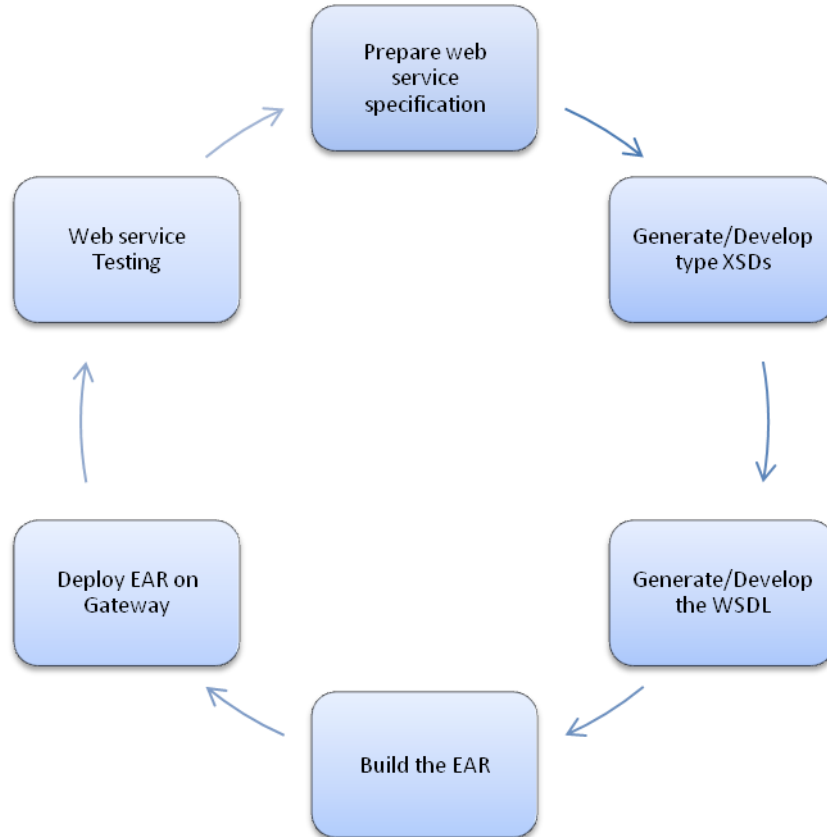
STOP Every FLEXCUBE IS Web service developed; need to be deployed on FLEXCUBE IS Gateway framework. Refer the FLEXCUBE Gateway deployment patterns for more information.

4. Pre requisites for Web service development

- Web service development Specification (explained in further sections)
- OPEN DEVELOPMENT tool -to generate the type XSDs and Gateway static data
- Target FLEXCUBE IS environment with Gateway framework implemented.

5. Web service Development

The below diagram represents the web service development cycle



5.1 Web Service specification

Specification	Example Data
What is the function ID web service required?	JKDMOBILE
What is the action of the function ID web service required?	NEW
What is the Web service name?	FCMobileService
What is the operation name?	CreateMobile
Open Development generated XSD available in the xsd folder, specified at Open Development?(Yes/No) <i>XSD files generated for the screen should be available for maintenance function IDs. For online function IDs, these type XSDs may be hand coded</i>	Yes
Property and EAR file generated from the installer? (Yes/No)	Yes

Specification	Example Data
<i>The property and EAR file are created using installer. Location for these files is specified during creation.</i>	

5.2 Generate / Develop XSDs

There are two ways to develop XSDs

- Use Open Development Tool
- Manually create XSD and maintain Gateway static data

5.2.1 Open Development generated

As a part of Function ID development, Open Development is used to generate the following web service artifacts:

- XSD folder with type XSDs
- INC folder that has following Gateway static data needed at runtime for web service.
 - GWTB_AMEND_FIELDS__JKDMOBIL.INC
 - GWTB_AMEND_NODES__JKDMOBIL.INC
 - GWTM_AMEND_FIELDS__JKDMOBIL.INC
 - GWTM_AMEND_MASTER__JKDMOBIL.INC
 - GWTM_AMEND_NODES__JKDMOBIL.INC

Refer Open Development documents for steps to get type XSDs and following INC files to populate Gateway maintenance tables.

5.2.2 Hand coded

If Open Development is not used for XSDs generation, these XSDs need to be created manually using existing samples available.

Developer expected to have proficiency in understanding complete FLEXCUBE IS web service framework and patterns to hand code the XSDs. Refer Section 6 for sample files. User can also refer existing contract online web service files for reference/samples.

If Open Development is not used, the Gateway static data need to maintain manually as follows:

1. Make the entry in GWTM_OPERATIONS_MASTER for every operation code for a given web service.

Example:

SERVICE_NAME	FCUBSCustomerService
OPERATION_CODE	ModifyMobile
SERVICE_DESC	Customer Service
OPERATION_DESCRIPTION	Modification of Mobile Number

SERVICE_NAME	FCUBSCustomerService
MODULE_CODE	ST
REPLY_REQD	N
FC_FUNCTION_ID	STDCIF
FC_ACTION	MODIFY
DEFAULT_FUNCTION	Y
SMS_FUNCTION_ID	STGCIF
SMS_ACTION	UNLOCK
FS_REQ_XSD	ST- ModifyMobile -Req-Full-MSG.xsd
PK_RES_XSD	ST- ModifyMobile -Res-PK-MSG.xsd
FS_RES_XSD	ST- ModifyMobile -Res-Full-MSG.xsd
IO_REQ_XSD	ST- ModifyMobile -Req-IO-MSG.xsd

2. Make the entry in GWTM_GATEWAY_FUNCTIONS

SERVICE_NAME	OPERATION_CODE	FUNCTION_ID	ACTION
FCUBSCustomerService	ModifyMobile	STGCIF	UNLOCK

3. Maintain external system details using function id UTDEXSYS

Example: FCAT

The screenshot displays the 'External System Details' configuration window. The 'External System' section is set to 'FCAT' with a description of 'FCAT'. The 'Correlation Pattern' is 'Request Correlation ID'. The 'Message Exchange Pattern' section shows 'Request Message' and 'Response Message' both set to 'Full Screen', with an unchecked 'XSD Validation Required' checkbox. The 'Queue' section shows 'Default Response Queue' as 'MDB_QUEUE_RESPOI' and 'Dead Letter Queue' as 'MDB_QUEUE_DLQ', with an unchecked 'Register Response Queue Message Id' checkbox. The 'External System Queues' table contains one entry: 'MDB_QUEUE' with 'MDB_QUEUE_RESPOI' as the response queue. The bottom status bar indicates the record was created by 'SYSTEM' on '2000-01-01 15:45:35' with 'Mod No 1' and 'Record Status Open'.

4. Maintain external system function using function id UTDEXFUN

External System Functions Details

New Copy Close Print Enter Query

External System * FCAT Description FCAT

Function * UTGATREP

Action * UNLOCK

Service Name FCISAuthRep

Operation Code ModifyAuthRep

Input by 216405U01 DateTime 2019-06-26 15:26:45 Mod No 1 Record Status Open

Authorized by 216405U02 DateTime 2019-06-26 15:28:14 Authorization Authorized Status

Exit

5. Make the entry in GWTB_AMEND_NODES for new operation code

NODE_NAME	AUTHREPTBL
NEW_ALLOWED	N
DELETE_ALLOWED	N
ALL_RECORDS	Y
SERVICE_NAME	FCISAuthRep
OPERATION_CODE	ModifyAuthRep

6. Make the entry in GWTB_AMEND_FIELDS for new operation code

NODE_NAME	AUTHREPTBL
FIELD_NAME	AUTHREPADDRESS
SERVICE_NAME	FCISAuthRep
OPERATION_CODE	ModifyAuthRep

7. Make Mobile number fields as Amendable for External system using function id UTDAMDMT

Amendment Details

New Copy Close Unlock Print Enter Query

Amend Nodes

1 of 1

Node Name *	New Allowed	Deleted Allowed	All Records
<input checked="" type="checkbox"/> AUTHREPTBL	No	No	Yes

Amend Fields

1 of 2

Field Name *

AUTHREADDRESS

Input by SYSTEM DateTime 2010-12-27 Mod No 1 Record Status Open
 Authorized by SYSTEM DateTime 2010-12-27 Authorization Status Authorized

Exit

8. Maintain external Source using function id SMDSORCE

Upload Source Maintenance

New Unlock Enter Query

Source Code * FCAT

Description FCAT

Authentication Required

Input by A29493A02 DateTime 2007-11-27 20:27:27 Mod No 1 Record Status Open
 Authorized by A29493M01 DateTime 2007-11-27 20:28:43 Authorization Status Authorized

Exit

9. Maintain external Source parameters preference using function id CODUPLDM

6. Annexure A – FCMobileService – Sample files

6.1 XSD files

6.1.1 CS-Mobil-Types.xsd

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <xs:annotation>

    <xs:documentation xml:lang="en">All Types of JKDMOBIL</xs:documentation>

  </xs:annotation>

  <xs:complexType name="Mobil-PK-Type">

    <xs:sequence>

      <xs:element name="CELL_NO" type="xs:decimal"/>

      <xs:element name="CUSTOMER_NO" minOccurs="1">

        <xs:simpleType>

          <xs:restriction base="xs:string">

            <xs:maxLength value="9"/>

          </xs:restriction>

        </xs:simpleType>

      </xs:element>

    </xs:sequence>

  </xs:complexType>

  <xs:complexType name="Mobil-Full-Type">

    <xs:sequence>

      <xs:element name="CUSTOMER_NO" minOccurs="1">
```

```

<xs:simpleType>
  <xs:restriction base="xs:string">
    <xs:maxLength value="9"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="CELL_NO" minOccurs="1" type="xs:decimal"/>
<xs:element name="MAKER" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="12"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="MAKERSTAMP" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="35"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="CHECKER" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="12"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="CHECKERSTAMP" minOccurs="0">
  <xs:simpleType>

```

```

        <xs:restriction base="xs:string">
            <xs:maxLength value="35"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="MODNO" minOccurs="0" type="xs:decimal"/>
<xs:element name="TXNSTAT" minOccurs="0">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="1"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="AUTHSTAT" minOccurs="0">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="1"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="Jktm-Junk-Mobil-Dtl" minOccurs="0"
maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="CUSTOMER_NO" minOccurs="1">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:maxLength value="9"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>

```



```

</xs:element>

<xs:element name="ACCOUNT_NO" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="20"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="NAME" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="105"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="TXNDETL" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="7"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="STATUS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="10"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="CELL_NO" minOccurs="0" type="xs:decimal"/>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Jktm-Junk-Mobile-Dtl2" minOccurs="0">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="CUSTOMER_NO" minOccurs="1">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:maxLength value="9"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="CUSTOMER_NAME" minOccurs="0">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:maxLength value="105"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="CUSTOMER_CATEGORY" minOccurs="0">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:maxLength value="10"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="ADDRESS" minOccurs="0">
                <xs:simpleType>
                    <xs:restriction base="xs:string">

```

```

        <xs:maxLength value="105"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="CITY" minOccurs="0">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="30"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="COUNTRY" minOccurs="0">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="25"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

```

```

<xs:complexType name="Mobil-Query-IO-Type">
    <xs:sequence>
        <xs:element name="CUSTOMER_NO" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:string">

```

```

        <xs:maxLength value="9"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
    <xs:element name="CELL_NO" minOccurs="1" type="xs:decimal"/>
</xs:sequence>
</xs:complexType>

<xs:complexType name="Mobil-Create-IO-Type">
    <xs:sequence>
        <xs:element name="CUSTOMER_NO" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:maxLength value="9"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="CELL_NO" minOccurs="1" type="xs:decimal"/>
        <xs:element name="Jktm-Junk-Mobil-Dtl" minOccurs="0"
maxOccurs="unbounded">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="CUSTOMER_NO" minOccurs="1">
                        <xs:simpleType>
                            <xs:restriction base="xs:string">
                                <xs:maxLength value="9"/>
                            </xs:restriction>
                        </xs:simpleType>
                    </xs:element>

```

```

<xs:element name="ACCOUNT_NO" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="20"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="NAME" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="105"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="TXNDETL" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="7"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="STATUS" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="10"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="CELL_NO" minOccurs="0" type="xs:decimal"/>
</xs:sequence>

```

```

    </xs:complexType>
</xs:element>
<xs:element name="Jktm-Junk-Mobile-Dtl2" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="CUSTOMER_NO" minOccurs="1">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="9"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="CUSTOMER_NAME" minOccurs="0">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="105"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="CUSTOMER_CATEGORY" minOccurs="0">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="10"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="ADDRESS" minOccurs="0">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:maxLength value="105"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="CITY" minOccurs="0">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="30"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="COUNTRY" minOccurs="0">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="25"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

<xs:complexType name="Mobil-Modify-IO-Type">
    <xs:sequence>
        <xs:element name="CUSTOMER_NO" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:maxLength value="9"/>

```

```

        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="CELL_NO" minOccurs="1" type="xs:decimal"/>
<xs:element name="MAKER" minOccurs="0">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="12"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="MAKERSTAMP" minOccurs="0">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="35"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>

```

```

<xs:complexType name="Mobil-Authorize-IO-Type">
    <xs:sequence>
        <xs:element name="CUSTOMER_NO" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:maxLength value="9"/>
                </xs:restriction>
            </xs:simpleType>

```



```

</xs:element>

<xs:element name="CELL_NO" minOccurs="1" type="xs:decimal"/>

<xs:element name="CHECKER" minOccurs="0">

  <xs:simpleType>

    <xs:restriction base="xs:string">

      <xs:maxLength value="12"/>

    </xs:restriction>

  </xs:simpleType>

</xs:element>

<xs:element name="CHECKERSTAMP" minOccurs="0">

  <xs:simpleType>

    <xs:restriction base="xs:string">

      <xs:maxLength value="35"/>

    </xs:restriction>

  </xs:simpleType>

</xs:element>

<xs:element name="MODNO" minOccurs="0" type="xs:decimal"/>

</xs:sequence>

</xs:complexType>

</xs:schema>

```

6.1.2 **CS-CreateMobil-Req-Full-MSG.xsd**

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://fcubs.ofss.com/service/FCMobileService"
  targetNamespace="http://fcubs.ofss.com/service/FCMobileService"
  elementFormDefault="qualified">

  <xs:include schemaLocation="IS-Messaging.xsd"/>

  <xs:include schemaLocation="CS-Mobil-Types.xsd"/>

  <xs:element name="CREATEMOBIL_FSFS_REQ">

    <xs:complexType>

      <xs:sequence>

```

```

    <xs:element name="FCUBS_HEADER" type="FCUBS_HEADERType"/>

    <xs:element name="FCUBS_BODY">

        <xs:complexType>

            <xs:sequence>

                <xs:annotation>

                    <xs:documentation xml:lang="en">This XSD Gives
Information about Full Request

                    </xs:documentation>

                </xs:annotation>

                <xs:element name="Jktm-Junk-Mobile-Full" type="Mobil-Full-
Type"/>

            </xs:sequence>

        </xs:complexType>

    </xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

```

6.1.3 CS-CreateMobil-Req-IO-MSG.xsd

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://fcubs.ofss.com/service/FCMobileService"
targetNamespace="http://fcubs.ofss.com/service/FCMobileService"
elementFormDefault="qualified">

    <xs:include schemaLocation="IS-Messaging.xsd"/>

    <xs:include schemaLocation="CS-Mobil-Types.xsd"/>

    <xs:element name="CREATEMOBIL_IOPK_REQ">

        <xs:complexType>

            <xs:sequence>

                <xs:element name="FCUBS_HEADER" type="FCUBS_HEADERType"/>

                <xs:element name="FCUBS_BODY">

                    <xs:complexType>

```

```

        <xs:sequence>
            <xs:annotation>
                <xs:documentation xml:lang="en">This XSD Gives
Information about IO Request
                </xs:documentation>
            </xs:annotation>
            <xs:element name="Jktm-Junk-Mobile-IO" type="Mobil-Create-
IO-Type"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

6.1.4 CS-CreateMobil-Res-Full-MSG.xsd

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://fcubs.ofss.com/service/FCMobileService"
targetNamespace="http://fcubs.ofss.com/service/FCMobileService"
elementFormDefault="qualified">

    <xs:include schemaLocation="IS-Messaging.xsd"/>

    <xs:include schemaLocation="CS-Mobil-Types.xsd"/>

    <xs:element name="CREATEMOBIL_FSFS_RES">

        <xs:complexType>

            <xs:sequence>

                <xs:element name="FCUBS_HEADER" type="FCUBS_HEADERTYPE"/>

                <xs:element name="FCUBS_BODY">

                    <xs:complexType>

                        <xs:sequence>

                            <xs:annotation>

                                <xs:documentation xml:lang="en">This XSD Gives
Information about Full Response

```

```

        </xs:documentation>

        </xs:annotation>

        <xs:element name="Jktm-Junk-Mobile-Full" type="Mobil-Full-
Type"/>

        <xs:element name="FCUBS_ERROR_RESP" type="ERRORType"
minOccurs="0" maxOccurs="unbounded"/>

        <xs:element name="FCUBS_WARNING_RESP" type="WARNINGType"
minOccurs="0" maxOccurs="unbounded"/>

        </xs:sequence>

        </xs:complexType>

        </xs:element>

    </xs:sequence>

    </xs:complexType>

    </xs:element>

</xs:schema>

```

6.1.5 CS-CreateMobil-Res-PK-MSG.xsd

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://fcubs.ofss.com/service/FCMobileService"
targetNamespace="http://fcubs.ofss.com/service/FCMobileService"
elementFormDefault="qualified">

    <xs:include schemaLocation="IS-Messaging.xsd"/>

    <xs:include schemaLocation="CS-Mobil-Types.xsd"/>

    <xs:element name="CREATEMOBIL_IOPK_RES">

        <xs:complexType>

            <xs:sequence>

                <xs:element name="FCUBS_HEADER" type="FCUBS_HEADERType"/>

                <xs:element name="FCUBS_BODY">

                    <xs:complexType>

                        <xs:sequence>

                            <xs:annotation>

```

```

        <xs:documentation xml:lang="en">This XSD Gives
Information about PK Response
        </xs:documentation>
    </xs:annotation>
    <xs:choice>
        <xs:element name="Jktm-Junk-Mobile-PK" type="Mobil-PK-
Type"/>
        <xs:element name="Jktm-Junk-Mobile-IO" type="Mobil-
Create-IO-Type"/>
    </xs:choice>
    <xs:element name="FCUBS_ERROR_RESP" type="ERRORType"
minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="FCUBS_WARNING_RESP" type="WARNINGType"
minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

6.2 WSDL file

6.2.1 FCMobileService.wsdl

```

<?xml version= "1.0" encoding="UTF-8"?>
<wsdl:definitions name="FCMobileService"
targetNamespace = "http://types.ws.gw.fcubs.ofss.com"
xmlns:tns = "http://types.ws.gw.fcubs.ofss.com"
xmlns:wsdlssoap = "http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wSDL = "http://schemas.xmlsoap.org/wsdl/"
xmlns:xs = "http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/"

```

```

xmlns:ns="http://fcubs.ofss.com/service/FCMobileService">

<wsdl:types>
  <xs:schema>
    <xs:import namespace="http://fcubs.ofss.com/service/FCMobileService"
      schemaLocation="../XSD/CS-CreateMobil-Req-Full-MSG.xsd"/>
  </xs:schema>
  <xs:schema>
    <xs:import namespace="http://fcubs.ofss.com/service/FCMobileService"
      schemaLocation="../XSD/CS-CreateMobil-Res-Full-MSG.xsd"/>
  </xs:schema>
  <xs:schema>
    <xs:import namespace="http://fcubs.ofss.com/service/FCMobileService"
      schemaLocation="../XSD/CS-CreateMobil-Req-IO-MSG.xsd"/>
  </xs:schema>
  <xs:schema>
    <xs:import namespace="http://fcubs.ofss.com/service/FCMobileService"
      schemaLocation="../XSD/CS-CreateMobil-Res-PK-MSG.xsd"/>
  </xs:schema>
</wsdl:types>

<wsdl:message name="CreateMobileFSReq">
  <wsdl:part name="RequestMsg" element="ns:CREATEMOBILE_FSFS_REQ"/>
</wsdl:message>

<wsdl:message name="CreateMobileFSRes">
  <wsdl:part name="ResponseMsg" element="ns:CREATEMOBILE_FSFS_RES"/>
</wsdl:message>

<wsdl:message name="CreateMobileIOReq">

```

```

        <wsdl:part name="RequestMsg" element="ns:CREATEMOBILE_IOPK_REQ"/>
</wsdl:message>

<wsdl:message name="CreateMobilePKRes">
    <wsdl:part name="ResponseMsg" element="ns:CREATEMOBILE_IOPK_RES"/>
</wsdl:message>

<wsdl:portType name="FCMobileServiceSEI" >
    <wsdl:operation name="CreateMobileIO">
        <wsdl:input message="tns:CreateMobileIOReq"/>
        <wsdl:output message="tns:CreateMobilePKRes"/>
    </wsdl:operation>

    <wsdl:operation name="CreateMobileFS">
        <wsdl:input message="tns:CreateMobileFSReq"/>
        <wsdl:output message="tns:CreateMobileFSRes"/>
    </wsdl:operation>

</wsdl:portType>

<wsdl:binding name="FCMobileService" type="tns:FCMobileServiceSEI">
    <wsdlsoap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="CreateMobileFS">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input>
            <wsdlsoap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <wsdlsoap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>

```

```

        </wsdl:output>
    </wsdl:operation>

    <wsdl:operation name="CreateMobileIO">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input>
            <wsdlsoap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <wsdlsoap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>

</wsdl:binding>

    <wsdl:service name="FCMobileService">
        <wsdl:port name="FCMobileServiceSEI" binding="tns:FCMobileService">
            <wsdlsoap:address
location="http://localhost:9080/FCMobileService/services/FCMobileService"/>
        </wsdl:port>
    </wsdl:service></wsdl:definitions>

```

6.3 Implementation Java files

6.3.1 FCMobileServiceImpl.java

```

/*
 * Class Name: FCMobileServiceImpl
 *
 * Date:      July 11, 2006 1:09:20 PM
 *

```


* Copyright 2005-2011 Oracle Financial Services Software Limited

* 10-11, SDF I, SEEPZ, Andheri (East),

* Mumbai - 400 096.

* India

*

* This source is part of the General Framework and is copyrighted by

* Oracle Financial Services Software Limited.

*

* All rights reserved. No part of this work may be reproduced, stored in a

* retrieval system, adopted or transmitted in any form or by any means,

* electronic, mechanical, photographic, graphic, optic recording or otherwise,

* translated in any language or computer language, without the prior written

* permission of Oracle Financial Services Software Limited.

*/

```
package com.ofss.fcubs.gw.ws.types;
```

```
import com.iflex.fcubs.gwservices.ejb.GWEJBClient;
```

```
import com.iflex.fcubs.gwutil.GWGenUtils;
```

```
import com.iflex.fcubs.gwutil.GWProperties;
```

```
import com.iflex.fcubs.gwutil.GWPropertiesConstants;
```

```
import com.iflex.fcubs.gwutil.GWPropertiesUtil;
```

```
import com.iflex.fcubs.gwutil.logging.GWLogger;
```

```
import com.iflex.fcubs.gwutil.logging.GWLoggerProperties;
```

```
import com.iflex.fcubs.gwutil.xml.GWXMLUtils;
```

```
import com.iflex.fcubs.gwutil.JAXBContextFactory;
```

```
import javax.xml.bind.JAXBContext;
```

```
import javax.xml.bind.JAXBException;
```

```
import javax.xml.bind.Marshaller;
```

```

import javax.xml.bind.Unmarshaller;

import java.io.ByteArrayInputStream;
import java.io.ByteArrayOutputStream;

import javax.annotation.PostConstruct;
import javax.annotation.PreDestroy;

import javax.jws.WebService;
import javax.xml.ws.BindingType;

@WebService(
    portName="FCMobileServiceSEI",
    serviceName="FCMobileService",
    targetNamespace="http://types.ws.gw.fcubs.ofss.com",
    endpointInterface="com.ofss.fcubs.gw.ws.types.FCMobileServiceSEI")
@BindingType("http://schemas.xmlsoap.org/wsdl/soap/http")

public class FCMobileServiceImpl implements FCMobileServiceSEI
{

//===== CLASS VARIABLES =====

    private final String g_className = "FCMobileServiceImpl";
    private final String g_serviceName = "FCMobileService";
    private String EJB_CALL_TYPE = "LOCAL";
    private String g_operationName = null;
    private GWProperties g_GWProperties = null;
    private GWLoggerProperties g_GWLoggerProp = null;
    private JAXBContext context = null;

```

```

//===== INITIALIZE METHODS =====

@PostConstruct

public void init()

{

    try{

        initializeComp();

    }

    catch(Exception ex){

        ex.printStackTrace();

    }

}

@PreDestroy

public void destroy() {

    g_GWProperties = null;

    g_GWLoggerProp = null;

}

public void initializeComp() throws Exception {

    String l_confp = GWPropertiesUtil.getGWPropPath() +
GWPropertiesConstants.GW_WS_PROP_FILENAME;

    g_GWProperties = new GWProperties(l_confp);

    String loggerPath =
g_GWProperties.getProperty(GWPropertiesConstants.LOGGER_PATH);

    g_GWLoggerProp = new GWLoggerProperties(loggerPath);

    EJB_CALL_TYPE =
g_GWProperties.getProperty(GWPropertiesConstants.EJB_CALL_TYPE,GWPropertiesCons
tants.EJB_LOCAL_CALL);

}

```

```

public CREATEMOBILEFSFSRES createMobileFS(CREATEMOBILEFSFSREQ requestMsg)
{
    g_operationName = "CreateMobile";

    String l_methodName = "createMobileFS(CREATEMOBILEFSFSREQ requestMsg)";

    CREATEMOBILEFSFSRES responseMsg = null;

    String l_req_msg = null;

    String l_resp_msg = null;

    GWLogger g_gwLog = null;

    try {
        if(g_gwLog == null)
        {
            g_gwLog = new GWLogger (
g_GWLoggerProp,GWPropertiesConstants.WS_GATEWAY + "_");
        }

        g_gwLog.println (g_className,l_methodName, " - ---START - ---");

        //Marshalling Request Object to XML

        context =
JAXBContextFactory.getInstance().getContext("com.ofss.fcubs.gw.ws.types");

        Marshaller marsh = context.createMarshaller();

        marsh.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, true);

        ByteArrayOutputStream output = new ByteArrayOutputStream();

        marsh.marshal(requestMsg, output);

        l_req_msg = output.toString("utf-8");

        g_gwLog.println (g_className,l_methodName,"Request XML Message:
\n" + l_req_msg + "\n");

        l_resp_msg = callEJB(l_req_msg, g_gwLog);

        g_gwLog.println (g_className,l_methodName,"Response XML Message:
\n" + l_resp_msg + "\n");
    }
}

```

```

        //Unmarshalling Response XML to Object

        ByteArrayInputStream input = new ByteArrayInputStream
(l_resp_msg.getBytes());

        Unmarshaller unmarsh = context.createUnmarshaller();

        responseMsg = (CREATEMOBILEFSRES)unmarsh.unmarshal(input);

        g_gwLog.println (g_className,l_methodName, "----END----");

    }

    catch (JAXBException e) {

        System.out.println("JAXBException: " + e.getMessage());

        e.printStackTrace();

    }

    catch(Exception ex) {

        g_gwLog.printStackTrace (g_className, l_methodName, ex);

    }

    finally {

        flushLogger(g_gwLog);

    }

    return responseMsg;

}

public CREATEMOBILEIOPKRES createMobileIO(CREATEMOBILEIOPKREQ requestMsg)

{

    g_operationName = "CreateMobile";

    String l_methodName = "createMobileIO(CREATEMOBILEIOPKREQ requestMsg)";

    CREATEMOBILEIOPKRES responseMsg = null;

    String l_req_msg = null;

    String l_resp_msg = null;

```

```

GWLogger g_gwLog = null;

    try {

        if(g_gwLog == null)

            {

                g_gwLog = new GWLogger (
g_GWLoggerProp,GWPropertiesConstants.WS_GATEWAY + "_");

            }

        g_gwLog.println (g_className,l_methodName, " - ---START - ---");

        //Marshalling Request Object to XML

        context =
JAXBContextFactory.getInstance().getContext("com.ofss.fcubs.gw.ws.types");

        Marshaller marsh = context.createMarshaller();

        marsh.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, true);

        ByteArrayOutputStream output = new ByteArrayOutputStream();

        marsh.marshal(requestMsg, output);

        l_req_msg = output.toString("utf-8");

        g_gwLog.println (g_className,l_methodName,"Request XML Message:
\n" + l_req_msg + "\n");

        l_resp_msg = callEJB(l_req_msg, g_gwLog);

        g_gwLog.println (g_className,l_methodName,"Response XML Message:
\n" + l_resp_msg + "\n");

        //Unmarshalling Response XML to Object

        ByteArrayInputStream input = new ByteArrayInputStream
(l_resp_msg.getBytes());

        Unmarshaller unmarsh = context.createUnmarshaller();

        responseMsg = (CREATEMOBILEIOPKRES)unmarsh.unmarshal(input);

        g_gwLog.println (g_className,l_methodName, "----END----");

    }

```

```

        catch (JAXBException e) {

            System.out.println("JAXBException: " + e.getMessage());

            e.printStackTrace();

        }

        catch(Exception ex) {

            g_gwLog.printStackTrace (g_className, l_methodName, ex);

        }

        finally {

            flushLogger(g_gwLog);

        }

        return responseMsg;

    }

private String calleJB(String p_reqMsg, GWLogger g_gwLog) throws Exception
{

    String l_methodName = "calleJB(String p_reqMsg,String g_gwLog)";

    String l_reqMsg = p_reqMsg;

    String l_respMsg = null;

    String l_replNS = "xmlns=\"http://fcubs.ofss.com/service/" + g_serviceName
+ "\"";

    g_gwLog.println (g_className,l_methodName,"----START----");

    String l_reqMsg_rootElement = GWGenUtils.getRootElement(l_reqMsg);

    String l_respMsg_rootElement =
l_reqMsg_rootElement.replaceAll("_REQ","_RES");

    String l_controlMetaData = " <FCUBS_CONTROL_METADATA> " +

        "\n\t<COMM_DET> " +

```

```

        "\n\t\t<GATEWAY>" +
GWPropertiesConstants.WS_GATEWAY + "</GATEWAY>" +

        "\n\t</COMM_DET>" +

        "\n\t<MSG_PROC_DET> " +

        "\n\t\t<SERVICE>" + g_serviceName +

"</SERVICE>" +

        "\n\t\t<OPERATION>" + g_operationName +

"</OPERATION>" +

        "\n\t</MSG_PROC_DET>" +

        "\n</FCUBS_CONTROL_METADATA>";

    g_gwLog.println (g_className, l_methodName, "l_controlMetaData = \n" +
l_controlMetaData + "\n");

    GWEJBClient l_GWEJBClient = new GWEJBClient(g_GWProperties);

    if ((GWPropertiesConstants.EJB_LOCAL_CALL).equals(EJB_CALL_TYPE)) {

        g_gwLog.println (g_className, l_methodName, "Calling
GWEJBClient.callLocalEJB");

        l_respMsg = l_GWEJBClient.callLocalEJB(l_reqMsg, l_controlMetaData);

        g_gwLog.println (g_className, l_methodName, "After calling
GWEJBClient.callLocalEJB");

    }

    else if ((GWPropertiesConstants.EJB_REMOTE_CALL).equals(EJB_CALL_TYPE)) {

        g_gwLog.println (g_className, l_methodName, "Calling
GWEJBClient.callRemoteEJB");

        l_respMsg = l_GWEJBClient.callRemoteEJB(l_reqMsg, l_controlMetaData);

        g_gwLog.println (g_className, l_methodName, "After calling
GWEJBClient.callRemoteEJB");

    }

    g_gwLog.println (g_className, l_methodName, "Response message from EJB: \n" +
l_respMsg + "\n");

    g_gwLog.println (g_className, l_methodName, "Calling GWGenUtils.replace");

```



```

        if( l_respMsg.indexOf("<" + l_respMsg_rootElement) != -1) {

            l_respMsg = GWGenUtils.replace(l_respMsg, "<" + l_respMsg_rootElement,
">", "<" + l_respMsg_rootElement + "\t" + l_replNS , false);

        }

        else {

            String l_errMsg = "Correct root tag (<" + l_respMsg_rootElement + ">) "+"
missing in the Request Message";

            g_gwLog.println (g_className, l_methodName, l_errMsg + ":" +
l_respMsg);

            throw new Exception(l_errMsg);

        }

        g_gwLog.println (g_className, l_methodName, "----After Calling
GWGenUtils.replace----");

        g_gwLog.println (g_className, l_methodName, "Response message to be sent:
\n" + l_respMsg + "\n");

        g_gwLog.println (g_className, l_methodName, "----END----");

        return l_respMsg;

    }

private void flushLogger(GWLogger g_gwLog)

{

    if (g_gwLog != null)

    {

        g_gwLog.flushLogger ();

        g_gwLog = null;

    }

}

}

```

6.4 Gateway static data from Open Development

6.4.1 GWTB AMEND FIELDS JKDMOBIL.INC

```
DELETE GWTB_AMEND_FIELDS A WHERE A.RAD_FUNCTION_ID = 'JKDMOBIL' ;

INSERT INTO
GWTB_AMEND_FIELDS (NODE_NAME, FIELD_NAME, SERVICE_NAME, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE', 'MAKER_ID', NULL, 'JKDMOBIL_MODIFY', 'JKDMOBIL');

INSERT INTO
GWTB_AMEND_FIELDS (NODE_NAME, FIELD_NAME, SERVICE_NAME, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE', 'MAKER_DT_STAMP', NULL, 'JKDMOBIL_MODIFY', 'JKDMOBIL');

COMMIT;
```

6.4.2 GWTB AMEND NODES JKDMOBIL.INC

```
DELETE GWTB_AMEND_NODES A WHERE A.RAD_FUNCTION_ID = 'JKDMOBIL' ;

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME, NEW_ALLOWED, DELETE_ALLOWED, ALL_RECORDS, SERVICE_NAME,
, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE', 'N', 'N', 'Y', NULL, 'JKDMOBIL_EXECUTEQUERY', 'JKDMOBIL')
;

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME, NEW_ALLOWED, DELETE_ALLOWED, ALL_RECORDS, SERVICE_NAME,
, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE_DTL', 'N', 'N', 'Y', NULL, 'JKDMOBIL_EXECUTEQUERY', 'JKDMOBIL');

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME, NEW_ALLOWED, DELETE_ALLOWED, ALL_RECORDS, SERVICE_NAME,
, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE_DTL2', 'N', 'N', 'Y', NULL, 'JKDMOBIL_EXECUTEQUERY', 'JKDMOBIL');

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME, NEW_ALLOWED, DELETE_ALLOWED, ALL_RECORDS, SERVICE_NAME,
, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE', 'N', 'N', 'Y', NULL, 'JKDMOBIL_NEW', 'JKDMOBIL');

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME, NEW_ALLOWED, DELETE_ALLOWED, ALL_RECORDS, SERVICE_NAME,
, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE_DTL', 'N', 'N', 'Y', NULL, 'JKDMOBIL_NEW', 'JKDMOBIL');

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME, NEW_ALLOWED, DELETE_ALLOWED, ALL_RECORDS, SERVICE_NAME,
, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE_DTL2', 'N', 'N', 'Y', NULL, 'JKDMOBIL_NEW', 'JKDMOBIL');

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME, NEW_ALLOWED, DELETE_ALLOWED, ALL_RECORDS, SERVICE_NAME,
, OPERATION_CODE, RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE', 'N', 'N', 'Y', NULL, 'JKDMOBIL_MODIFY', 'JKDMOBIL');
```

```

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,SERVICE_NAME
,OPERATION_CODE,RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE_DTL','N','N','Y',NULL,'JKDMOBIL_MODIFY','JKDMOBIL');

```

```

INSERT INTO
GWTB_AMEND_NODES (NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,SERVICE_NAME
,OPERATION_CODE,RAD_FUNCTION_ID)
VALUES ('JKTM_JUNK_MOBILE_DTL2','N','N','Y',NULL,'JKDMOBIL_MODIFY','JKDMOBIL');

```

```

COMMIT;

```

6.4.3 GWTM AMEND FIELDS JKDMOBIL.INC

```

DELETE GWTM_AMEND_FIELDS A WHERE A.EXT_SYSTEM = 'FLEXCUBE' AND
RAD_FUNCTION_ID = 'JKDMOBIL' ;

```

```

INSERT INTO
GWTM_AMEND_FIELDS (EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,FIELD_NAME,RAD_FUNCIO
N_ID)
VALUES ('FLEXCUBE','JKDMOBIL_MODIFY','JKTM_JUNK_MOBILE','MAKER_ID','JKDMOBIL');

```

```

INSERT INTO
GWTM_AMEND_FIELDS (EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,FIELD_NAME,RAD_FUNCIO
N_ID)
VALUES ('FLEXCUBE','JKDMOBIL_MODIFY','JKTM_JUNK_MOBILE','MAKER_DT_STAMP','JKDMO
BIL');

```

```

COMMIT;

```

6.4.4 GWTM AMEND MASTER JKDMOBIL.INC

```

DELETE GWTM_AMEND_MASTER A WHERE A.EXT_SYSTEM = 'FLEXCUBE' AND
RAD_FUNCTION_ID = 'JKDMOBIL' ;

```

```

INSERT INTO
GWTM_AMEND_MASTER (EXT_SYSTEM,SOURCE_OPERATION,SERVICE_NAME,OPERATION_CODE,MAKE
R_ID,CHECKER_ID,MAKER_DT_STAMP,CHECKER_DT_STAMP,ONCE_AUTH,AUTH_STAT,RECORD_STA
T,MOD_NO,RAD_FUNCTION_ID)
VALUES ('FLEXCUBE','JKDMOBIL_EXECUTEQUERY',NULL,'JKDMOBIL_EXECUTEQUERY','RADTOO
L','RADTOOL',TO_DATE('18-MAR-2011','DD-MON-RRRR'),TO_DATE('18-MAR-2011','DD-
MON-RRRR'),'Y','A','O',1,'JKDMOBIL');

```

```

INSERT INTO
GWTM_AMEND_MASTER (EXT_SYSTEM,SOURCE_OPERATION,SERVICE_NAME,OPERATION_CODE,MAKE
R_ID,CHECKER_ID,MAKER_DT_STAMP,CHECKER_DT_STAMP,ONCE_AUTH,AUTH_STAT,RECORD_STA
T,MOD_NO,RAD_FUNCTION_ID)
VALUES ('FLEXCUBE','JKDMOBIL_NEW',NULL,'JKDMOBIL_NEW','RADTOOL','RADTOOL',TO_DA
TE('18-MAR-2011','DD-MON-RRRR'),TO_DATE('18-MAR-2011','DD-MON-
RRRR'),'Y','A','O',1,'JKDMOBIL');

```

```

INSERT INTO
GWTM_AMEND_MASTER (EXT_SYSTEM,SOURCE_OPERATION,SERVICE_NAME,OPERATION_CODE,MAKE
R_ID,CHECKER_ID,MAKER_DT_STAMP,CHECKER_DT_STAMP,ONCE_AUTH,AUTH_STAT,RECORD_STA
T,MOD_NO,RAD_FUNCTION_ID)
VALUES ('FLEXCUBE','JKDMOBIL_MODIFY',NULL,'JKDMOBIL_MODIFY','RADTOOL','RADTOOL'
,TO_DATE('18-MAR-2011','DD-MON-RRRR'),TO_DATE('18-MAR-2011','DD-MON-
RRRR'),'Y','A','O',1,'JKDMOBIL');

```

```

COMMIT;

```

6.4.5 GWTM AMEND_NODES JKDMOBIL.INC

```
DELETE GWTM_AMEND_NODES A WHERE A.EXT_SYSTEM = 'FLEXCUBE' AND RAD_FUNCTION_ID = 'JKDMOBIL' ;
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_EXECUTEQUERY','JKTM_JUNK_MOBILE','N','N','Y','JKDMOBIL');
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_EXECUTEQUERY','JKTM_JUNK_MOBILE_DTL','N','N','Y','JKDMOBIL');
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_EXECUTEQUERY','JKTM_JUNK_MOBILE_DTL2','N','N','Y','JKDMOBIL');
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_NEW','JKTM_JUNK_MOBILE','N','N','Y','JKDMOBIL');
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_NEW','JKTM_JUNK_MOBILE_DTL','N','N','Y','JKDMOBIL');
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_NEW','JKTM_JUNK_MOBILE_DTL2','N','N','Y','JKDMOBIL');
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_MODIFY','JKTM_JUNK_MOBILE','N','N','Y','JKDMOBIL');
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_MODIFY','JKTM_JUNK_MOBILE_DTL','N','N','Y','JKDMOBIL');
```

```
INSERT INTO
GWTM_AMEND_NODES(EXT_SYSTEM,SOURCE_OPERATION,NODE_NAME,NEW_ALLOWED,DELETE_ALLOWED,ALL_RECORDS,RAD_FUNCTION_ID)
VALUES('FLEXCUBE','JKDMOBIL_MODIFY','JKTM_JUNK_MOBILE_DTL2','N','N','Y','JKDMOBIL');
```

```
COMMIT;
```



Web Service Development
[May] [2021]
Version 14.5.0.0.0

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], [2021], 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.