Contents

1 Preface .................................................................................................................................................................................. 3
  1.1 Audience ........................................................................................................................................................................ 3
  1.2 Related documents ............................................................................................................................................................ 4
  1.3 Conventions ........................................................................................................................................................................ 4
2 Introduction ............................................................................................................................................................................. 4
  2.1 How to use this Guide ......................................................................................................................................................... 4
3 Extensibility Introduction ......................................................................................................................................................... 5
  3.1 What is extensibility ............................................................................................................................................................ 5
  3.2 Industry Pattern .................................................................................................................................................................... 5
  3.3 Industry Approach ............................................................................................................................................................... 5
4 FLEXCUBE UBS Extensibility .............................................................................................................................................. 5
  4.1 Business Areas .................................................................................................................................................................... 5
  4.2 FLEXCUBE UBS Extensibility approach ............................................................................................................................ 6
  4.3 FLEXCUBE UBS Extensibility user roles .......................................................................................................................... 6
5 FLEXCUBE UBS Extensible features .................................................................................................................................... 8
  5.1 Screen changes ..................................................................................................................................................................... 8
    5.1.1 New Screens ............................................................................................................................................................... 8
    5.1.2 Screen Modifications ................................................................................................................................................... 8
    5.1.3 Amend field level attributes ...................................................................................................................................... 9
    5.1.4 Style Sheet changes ................................................................................................................................................... 9
    5.1.5 Language conversion of screens ................................................................................................................................ 9
  5.2 Functional ............................................................................................................................................................................ 9
    5.2.1 User Defined fields at Maintenance ............................................................................................................................ 9
    5.2.2 New SDE and rule for IC calculation .......................................................................................................................... 10
    5.2.3 Configurable workflow of Branch Screens ............................................................................................................. 10
  5.3 Processing logic .................................................................................................................................................................. 10
    5.3.1 Additional validation logic for a field or group of fields .......................................................................................... 10
    5.3.2 Modify defaulting logic for fields ............................................................................................................................ 11
    5.3.3 Online contract extensibility .................................................................................................................................... 12
    5.3.4 Batch extensibility ................................................................................................................................................... 12
  5.4 Notifications ...................................................................................................................................................................... 12
    5.4.1 Event based notifications ........................................................................................................................................ 12
  5.5 Reports .............................................................................................................................................................................. 13
    5.5.1 New BIP Reports ....................................................................................................................................................... 13
    5.5.2 New OBIEE based reports ........................................................................................................................................ 13
  5.6 User defined Advice tags .................................................................................................................................................. 13
    5.6.1 New tag in Advice message ..................................................................................................................................... 13
  5.7 Interface ............................................................................................................................................................................ 13
    5.7.1 Switch Interface ISO8583 configuration .................................................................................................................... 13
    5.7.2 Configurable Generic Interface for upload/handoff ............................................................................................ 13
    5.7.3 Upload adapter framework ...................................................................................................................................... 14
  6 Extensibility development life cycle .................................................................................................................................... 14
    6.1 Define Extensibility Requirement .................................................................................................................................. 14
    6.2 Identify the Business area of extensibility ....................................................................................................................... 15
    6.3 Identify the tools/framework to be used .......................................................................................................................... 15
    6.4 Identify the file types & layers applicable ..................................................................................................................... 16
    6.5 Develop changes ............................................................................................................................................................. 16
    6.6 Test it in FLEXCUBE environment ................................................................................................................................ 16
7 Resources .................................................................................................................................................................................. 16
1 Preface

This document describes the concepts and helps reader to get started using Extensible framework of FLEXCUBE UBS Application, to develop additional functionalities.

1.1 Audience

The Extensibility getting started book is intended for FLEXCUBE Application Developers/Users who are authorized to perform the following tasks:

- Modify the layouts of existing FLEXCUBE Screens
- Modify the existing functionality by adding new fields/tabs/data blocks
- Extend the existing screen to have fields based on customer specific table/fields
- Add customer specific validations at extension hooks
- Add customer specific processing logics in batch processing
- Add customer specific notifications
- Add customer specific calculation elements
- Add customer specific reports

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

<table>
<thead>
<tr>
<th>Proficiency</th>
<th>Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLEXCUBE UBS Development overview</td>
<td>FCUBS-FD01-01-01-Development</td>
</tr>
<tr>
<td></td>
<td>Overview Guide</td>
</tr>
<tr>
<td>RAD function ID development getting started</td>
<td>FCUBS-FD02-01-01-RAD Getting</td>
</tr>
<tr>
<td></td>
<td>Started</td>
</tr>
</tbody>
</table>

1.2 Related documents

For more information on RAD development and extensibility, refer the below documents:

- FCUBS-FD01-01-01-Development Overview Guide
- FCUBS-FD02-02-01-RAD Function ID Development
- FCUBS-FD02-03-01-RAD Web Service Development
- FCUBS-FD02-04-01-RAD BIP Report Integration
- FCUBS-FD02-05-01-RAD Notification Development
- FCUBS-FD05-02-01-RAD-Reference
- FCUBS-FD03-02-01-Extensibility Reference Guide
- FCUBS-FD03-03-01-Extensibility By Example Volume 1
- FCUBS-FD03-03-02-Extensibility By Example Volume 2
- FCUBS-FD04-02-01-Generic Interface Configuration Guide
- FCUBS-FD04-03-01-Upload Adapter Development Guide
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.

- **italic** 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 names, URLs, text that appears on the screen, or text that you enter.

2 Introduction

2.1 How to use this Guide

The information in this guide includes:

- Chapter 2, “Introduction”
- Chapter 3, “Extensibility Introduction”
- Chapter 4, “FLEXCUBE UBS Extensibility”
- Chapter 5, “FLEXCUBE Extensible features”
- Chapter 6, “Extensible Development Life Cycle”
- Chapter 7, “Resources”
3 Extensibility Introduction

3.1 What is extensibility

Extensibility is an ability of the software system to allow and accept the significant extension of its capabilities without major rewriting of code or changes in its basic architecture. Extensible systems provide technology, tools, languages that designed so that developers can expand or add to its capabilities.

3.2 Industry Pattern

Following are the industry pattern to address the extensibility in software architecture
- Frameworks
- Configuration files
- Extension using scripts
- User specific extension software packages
- Object based programming where inheritance is used for extensibility

3.3 Industry Approach

Industry approaches to extensibility typically includes following:

- Tools to allow to extend the functionality of base product
- Program hooks to allow developers to insert their program routines
- Ability define new business events to address change in process
- Ability to create regional specific software changes
- Ability to add/remove fields at business messages
- Ability to configure interface protocols without software change

4 FLEXCUBE UBS Extensibility

4.1 Business Areas

One of the primary goals of the FLEXCUBE UBS architecture is that system should be able to be extendable in required business specific areas. Following are such areas where extensibility is required:

<table>
<thead>
<tr>
<th>Business Area</th>
<th>Why extensibility required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Screen changes</td>
<td>User may want to keep some screens simple to improve training &amp; operational efficiency</td>
</tr>
<tr>
<td>Language of Screens</td>
<td>User may wish to provide screens other than the default language of software</td>
</tr>
<tr>
<td>Business / legal requirements</td>
<td>Certain processing/calculation logic may be applied specific to region/country judiciary.</td>
</tr>
<tr>
<td>Events &amp; eco system participation</td>
<td>The software has to be part of bigger eco</td>
</tr>
</tbody>
</table>
User configurable messages / reports
- System by providing other integration/notification mechanism
- Software should provide mechanism to extract required information. System to be open to provide the same

Ad hoc exchange of information between systems
- System should provide mechanism to exchange information with ad-hoc systems over the period time.

### 4.2 FLEXCUBE UBS Extensibility approach

FLEXCUBE UBS provides the following approach to address the extensible requirement.

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Industry Approach</th>
<th>FLEXCUBE Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Framework</td>
<td>Tools and framework to extend the base product</td>
<td>RAD framework</td>
</tr>
<tr>
<td>Configuration files</td>
<td>XML files / Text files to configure</td>
<td>XML configuration files and Text based configuration files</td>
</tr>
<tr>
<td>Extension using Scripts</td>
<td>Scripts</td>
<td>Java Script based extensions to enable extension at user interface layer</td>
</tr>
<tr>
<td>User specific extension of software package</td>
<td>Program hooks to allow extension logic call outs</td>
<td>FLEXCUBE call outs based on release type CUSTOM, CLUSTER</td>
</tr>
<tr>
<td>User defined events</td>
<td>Ability to define new events/message types</td>
<td>FEXCUBE notification and messaging architecture to define new XML message types</td>
</tr>
<tr>
<td>Protocol tweaking</td>
<td>Configuration of protocol without software change</td>
<td>FLEXCUBE ISO8583 protocol definition in XML file that can be modifiable.</td>
</tr>
<tr>
<td>User/Regional specific processing logic</td>
<td>Ability to extend the core processing logic</td>
<td>FLEXCUBE UDF, SDE (IC and CL) extensions</td>
</tr>
</tbody>
</table>

### 4.3 FLEXCUBE UBS Extensibility user roles

FLEXCUBE UBS Extensibility development can be classified into 4 types based on the complexity and user competencies required:

- Application maintenance/definition of components
  - User expected to login into FLEXCUBE application and use certain function IDs to define the new components. This is typically applicable to Bank business user who requires new functionality.
Example, user need to use function ID UDDMAINT to define new UDF field

- Configuration files
  User expected to modify some of the parameters in configuration files. This may require restart of application or relevant applications. Typically this is required for application administrators in bank.

  Example, user may need to modify the ISO8583 protocol definition

- Tools based development
  User expected to understand the given function ID working and required to extend the functionality by adding new data sources and fields. This is typically required by IT developer in bank.

  Example: User needs to change screen layout, to add new data blocks based on new tables added in database.

- Programming
  User expected to achieve granular control and validations using programming extensions. User expected to know the language used thoroughly in this context. This is typically required by advanced developers in bank.

  Example, bank required to modify the defaulting and validation or modify the processing flow at specific call out points.

Developer role and extensible approach matrix

Given below matrix depicts the developer role and possible extensible approaches to apply:

<table>
<thead>
<tr>
<th>Developer role</th>
<th>Maintenance/Definition</th>
<th>Configuration</th>
<th>Tools</th>
<th>Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>Implementer</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

*Implementer could be OFSS staff or customer / partner staff who implements FLEXCUBE*
5 FLEXCUBE UBS Extensible features

This section describes the extensible features available in FLEXCUBE UBS

5.1 Screen changes

This section describes features that are specific to Function ID (screens) extensibility. RAD tool is used for function ID extensibility.

5.1.1 New Screens

RAD tool used to develop the new screens depending upon the bank requirement. The screens are based on existing or new tables added in database.

Refer following documents for more information on working with screens.

- FCUBS-FD02-01-01-RAD Getting Started
- FCUBS-FD05-02-01-RAD-Reference
- FCUBS-FD03-02-01-Extensibility Reference Guide

5.1.2 Screen Modifications

Existing screens layouts can be modified using RAD tool to suite as follows:

- Hide fields that are not relevant to a given implementation
- Modify the placement of the fields (example moving from one tab to other tab)
- Add LOV to a given field
- Changing the data type
- Adding enumerations to a given field to restrict user inputs
- To increase the set fields (example adding the address line 5)

Refer FCUBS-FD03-03-01-Extensibility By Example Volume 1 for examples.

5.1.3 Amend field level attributes

Existing file level attributes can be modified to add below:

- Defaulting some value to reduce user input/errors.
- Restricting the maximum and minimum value
- Precision settings

Refer FCUBS-FD03-03-01-Extensibility By Example Volume 1 for examples.

5.1.4 Style Sheet changes

FLEXCUBE UBS provides style editor to enable CSS changes to have following user specific UI elements design:

- Page template changes
- Dialog template changes
- Form elements look and feel
- Text fonts
- Tables look and feel
- Colors changes

Refer FS_StyleDesigner_for_FC_UBS_10.3.0.0.0.0.0.zip for more information

5.1.5 Language conversion of screens

FLEXCUBE screens can be extended to support languages other than English.

5.2 Functional

5.2.1 User Defined fields at Maintenance
UDF framework enables the bank user to add the new field without changing any table structure. This is used in maintenance function IDs where new field required by bank user.

Refer FCUBS-FD03-03-01-Extensibility By Example Volume 1 for examples

5.2.2 New SDE and rule for IC calculation

IC module SDE framework enables user to add the user specific system data element for which user can write the data fetch logic. This SDE can be used further in building the interest calculation logic.

Refer FCUBS-FD03-03-01-Extensibility By Example Volume 1 for examples

5.2.3 Configurable workflow of Branch Screens

Configurable stages available for FLEXCUBE UBS Branch function IDs. Branch function ID can be identified the module type WB in menu static data. User can define the function ID and applicable stages.

Refer Savings.zip user manual section 3 for workflow definition features.

5.3 Processing logic

5.3.1 Additional validation logic for a field or group of fields

FLEXCUBE UBS provides the extension call outs in database layer. These extension call outs are extensible package and pre-named procedures to be used for extensibility. The base product will call this call outs during runtime with required PLSQL data type as parameters.

Example:

User wanted extends STDCIF function to add capital letter validation for the field “card holder name”. This can be achieved as follows:

Edit the STPKS_STDCIF_CUSTOME.Fn_Pre_Default.Validate as below

```sql
FUNCTION Fn_Pre_Default_And_Validate
(p_Source           IN VARCHAR2,
p_Source_Operation IN VARCHAR2,
p_Function_Id      IN VARCHAR2,
p_Action_Code      IN VARCHAR2,
p_Child_Function   IN VARCHAR2,
p_stdcif           IN stpks_stdcif_Main.ty_stdcif,
p_Prev_stdcif      IN OUT stpks_stdcif_Main.ty_stdcif,
p_Wrk_stdcif       IN OUT stpks_stdcif_Main.ty_stdcif,
p_Err_Code         IN OUT VARCHAR2,
p_Err_Params       IN OUT VARCHAR2)
RETURN BOOLEAN IS
BEGIN
```

FD03-01-01 Extensibility Getting Started 10
Dbg('In Fn_Pre_Default_And_Validate..');

--extensibility code start
p_Wrk_stdccif:= p_stdcif;

IF  p_wrk_stdcif.v_sttms_customer.CARD_HOLDER_NAME NOT IN (upper(p_wrk_stdcif.v_sttms_customer.CARD_HOLDER_NAME)) THEN
  p_err_code    := 'ST-OTHR-097';
  p_err_params  := NULL;
  Dbg('Out of validation code-Sarva');
  RETURN FALSE;
END IF;
--extensibility code ends

Dbg('Returning Success From fn_pre_default_and_validate..');

RETURN TRUE;
EXCEPTION
WHEN OTHERS THEN
  Debug.Pr_Debug('**', 'In When Others of stpks_stdcif_Custom.Fn_Pre_Default_And_Va
  **', SQLERRM);
  p_Err_Code   := 'ST-OTHR-001';
  p_Err_Params := NULL;
  RETURN FALSE;
END Fn_Pre_Default_And_Va

Note:
Open RAD XML for a given function ID using RAD tool to understand the data block and field name. This would give above complete path to access the field name. you can prefix “p_” to get function ID data type and “v_” to data block to get data block name.

Example: to know the card holder name element at runtime, use following template:

[p function_id type].[data block name].[field name]
p_wrk_stdcif.v_sttms_customer.CARD_HOLDER_NAME

Refer FCUBS-FD03-03-01-Extensibility By Example Volume 1 for detailed steps involved in this exercise.

5.3.2 Modify defaulting logic for fields
FLEXCUBE UBS call outs allows to change defaulting logic for elements using PLSQL data types.

Note:
Refer example given in section 5.3.1 to know how to identify the element name
5.3.3 Online contract extensibility

FLEXCUBE UBS allows to modify or enrich the online processing logics at given call out functions.

**Note:** How to identify package name?

Refer the RAD generated packages for CUSTOM and CLUSTER types to know the possible call outs available which has PLSQL data type parameters. To arrive at the package name using following template.

Template:
<Module code>PKS_<Function ID>_<Release type>

Example:
To get the CUSTOM release of function ID FTDTRONL which belongs to FT module, package would be

*FTPKS_FTDTRONL_CUSTOM*

5.3.4 Batch extensibility

Apart from RAD generated function ID based database packages, FLEXCUBE UBS allows to modify below core service packages. Note that these are core packages which don’t have any function ID associated.

ACPKS package is extensible using following packages:

- ACPKS_CUSTOM
- ACPKS_CLUSTER

WRP_BATCH batch running package is extensible using following packages:

- WRP_BATCH_CLUSTER
- WRP_BATCH_CUSTOM

Refer the *FCUBS-FD03-03-02-Extensibility By Example Volume 2* for example.

5.4 Notifications

5.4.1 Event based notifications

FLEXCUBE UBS supports events based notification framework, where notification triggers can be developed as per user requirement. Once the event occurs, the framework pushes the required data to external systems.
5.5 Reports
FLEXCUBE provides factory shipped BIP canned reports and OBIEE repositories. User can extend the reports or repositories to suite the local requirements.

Refer FCUBS-FD07-01-01-Report Getting started for more information

5.5.1 New BIP Reports
User can develop the new report or modify the existing report to change report query, result columns or filter criteria.

Refer FCUBS-FD07-02-01-BIP Report Development Guide for more information on BIP report development

5.5.2 New OBIEE based reports
User can develop the new OBIEE repositories or work with existing OBIEE repositories.

Refer FCUBS-FD07-03-01-OBIEE repository Development Guide for more information on OBIEE repositories development.

5.6 User defined Advice tags

5.6.1 New tag in Advice message
FLEXCUBE provides rich set of advices with pre-defined tags for each message type. During run time, the contents are fetched and filled in advices. User can define new TAG and add code to fetch the logic.

5.7 Interface

5.7.1 Switch Interface ISO8583 configuration
- FLEXCUBE user can configure the version and protocol fields of ISO8583 based SWITCH interface gateway.
- User can define the mapping of ISO processing code and FLEXCUBE internal transaction code.

5.7.2 Configurable Generic Interface for upload/handoff
User can define following interfaces
- Incoming – to get data into FLEXCUBE
- Outgoing – to get data out of FLEXCUBE

Refer FCUBS-FD04-02-01-Generic Interface Configuration Guide on how to define generic interface.

5.7.3 Upload adapter framework
FLEXCUBE UBS provides factory shipped adapters (spread sheet based upload) for incoming Interface upload purpose. User can extend by developing new adapter using upload adapter framework.

Refer FCUBS-FD04-03-01-Upload Adapter Development Guide

6 Extensibility development life cycle
Extensibility development involves following stages. These stages are explained in detail further down the line.

6.1 Define Extensibility Requirement
Extensibility Requirements need to be clearly defined and documented. This requirement should describe the module, function ID (if applicable) and intended functionality required. This requirement should have justification of why extensibility needed compared with base functionality. It also should cover other alternatives to achieve the functionality without extensibility.
6.2 Identify the Business area of extensibility
Depending upon the Requirement, user needs to identify the FLEXCUBE business area that requires extensibility development. This includes:
- Function ID (New, modify existing, add fields, hide fields)
- Processing logic (defaulting, enriching, validating)
- UDF (New UDF fields for identified function IDs)
- SDE (new SDE for calculation purpose)
- Accounting
- Batch (New batch function during EOD time or intraday)
- Notification (New event notification)
- Report (new report or modify existing report query)
- Interface (New incoming or outgoing)
- Adapter (Migration / data upload into FLEXCUBE)

6.3 Identify the tools/framework to be used

<table>
<thead>
<tr>
<th>Area</th>
<th>FLEXCUBE Tools/Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function ID</td>
<td>• RAD</td>
</tr>
<tr>
<td></td>
<td>• Style sheet editor</td>
</tr>
<tr>
<td>Processing logic</td>
<td>• PLSQL programming on RAD generated packages</td>
</tr>
<tr>
<td></td>
<td>• PLSQL programming on core packages</td>
</tr>
<tr>
<td>UDF</td>
<td>• UDDMAINT function ID</td>
</tr>
<tr>
<td></td>
<td>• PLSQL programming on -Hand coding of UDF logics</td>
</tr>
<tr>
<td>SDE</td>
<td>• SDE maintenance</td>
</tr>
<tr>
<td></td>
<td>• PLSQL programming on -Hand coding of SDE data fetch logic</td>
</tr>
<tr>
<td>Accounting</td>
<td>• PLSQL programming on -Accounting extensible packages</td>
</tr>
<tr>
<td>Batch</td>
<td>• PLSQL programming on -Batch extensible</td>
</tr>
<tr>
<td>Notification</td>
<td>• RAD – Notification trigger</td>
</tr>
</tbody>
</table>
6.4 Identify the file types & layers applicable

The below table described the layer and file types developed for each extensibility business areas that involves software modification.

<table>
<thead>
<tr>
<th>Area</th>
<th>Client Layer</th>
<th>Application Layer</th>
<th>Database Layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function ID</td>
<td>Java script files</td>
<td>UIXML</td>
<td>RAD generated CUSTOM/CLUSTER packages</td>
</tr>
<tr>
<td>Processing logic</td>
<td></td>
<td></td>
<td>CORE FLEXCUBE Packages</td>
</tr>
<tr>
<td>UDF</td>
<td></td>
<td></td>
<td>UDF specific procedures and packages</td>
</tr>
<tr>
<td>SDE Accounting</td>
<td></td>
<td></td>
<td>IC and CL packages</td>
</tr>
<tr>
<td>Batch</td>
<td></td>
<td>Batch processing</td>
<td>Batch processing package</td>
</tr>
<tr>
<td>Notification</td>
<td></td>
<td>RAD generated</td>
<td>Notification Triggers</td>
</tr>
<tr>
<td>Reports</td>
<td></td>
<td>RTF file</td>
<td>RAD generated Report packages</td>
</tr>
<tr>
<td>Interface</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>

6.5 Develop changes

User can develop the required changes using respective tools documents.

Refer section 5 for development documentation help on each area of extensibility.

6.6 Test it in FLEXCUBE environment

User need to copy the developed files to target environment and can test the developed functionality. Refer the FLEXCUBE UBS installation manuals on how to deploy the changes.

7 Resources

Refer the below resources to gain further working knowledge with extensibility
<table>
<thead>
<tr>
<th>To Do</th>
<th>Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAD Getting started</td>
<td>FCUBS-FD02-01-01-RAD Getting Started</td>
</tr>
<tr>
<td>RAD complete reference guide</td>
<td>FCUBS-FD05-02-01-RAD-Reference</td>
</tr>
<tr>
<td>RAD screen development step by step procedure</td>
<td>FCUBS-FD02-02-01-RAD Function ID Development</td>
</tr>
<tr>
<td>RAD web service development</td>
<td>FCUBS-FD02-03-01-RAD Web Service Development</td>
</tr>
<tr>
<td>BIP report integration with RAD screen</td>
<td>FCUBS-FD02-04-01-RAD BIP Report Integration</td>
</tr>
<tr>
<td>Outbound Notification trigger development</td>
<td>FCUBS-FD02-05-01-RAD Notification Development</td>
</tr>
<tr>
<td>Extensibility Reference guide</td>
<td>FCUBS-FD03-02-01-Extensibility Reference Guide</td>
</tr>
<tr>
<td>Extensibility use case development examples</td>
<td>FCUBS-FD03-03-01-Extensibility By Example</td>
</tr>
<tr>
<td>Branch work flow definition</td>
<td>Savings.zip (WB user manual)</td>
</tr>
<tr>
<td>Style sheet editor</td>
<td>FS_StyleDesigner_for_FC_UBS_10.3.0.0.0.0.zip</td>
</tr>
<tr>
<td>Generic Interface configuration Guide</td>
<td>FCUBS-FD04-02-01-Generic Interface Configuration Guide</td>
</tr>
<tr>
<td>FLEXCUBE Upload adapter development</td>
<td>FCUBS-FD04-03-01-Upload Adapter Development Guide</td>
</tr>
</tbody>
</table>