Reference Guide
Oracle FLEXCUBE Investor Servicing
Release 14.0.0.0.0
[May] [2018]

Table of Contents

1. PREFACE 1-1

1.1 AUDIENCE 1-1
1.2 RELATED DOCUMENTS 1-1
1.3 CONVENTIONS 1-2

2. INTRODUCTION 2-1

2.1 HOW TO USE THIS GUIDE 2-1

3. OVERVIEW OF ORACLE FLEXCUBE MANAGEMENT AND INTEGRATION CONSOLE 3-1

3.1 RELEASE DETAILS: 3-2
3.2 ENVIRONMENT DETAILS: 3-2

4. SCREEN DESIGN 4-1

4.1 BASIC DETAILS 4-1
4.1.1 Action 4-2
4.1.2 Function Id 4-2
4.1.3 Save Xml Path 4-2
4.1.4 Function Type 4-2
4.1.5 Parent Function 4-2
4.1.6 Parent Xml 4-2
4.1.7 Function Category 4-2
4.1.8 Header Template 4-2
4.1.9 Footer Template 4-2
4.1.10 Deploy 4-3
4.1.11 Release 4-3
4.1.12 All Packages 4-3
4.2 PREFERENCES 4-3
4.2.1 Module 4-4
4.2.2 Head office Function 4-4
4.2.3 Logging Required 4-4
4.2.4 Auto AEOD Aware 4-4
4.2.5 Auto Authorization 4-4
4.2.6 Tank Modification 4-4
4.2.7 Field Log Required 4-4
4.2.8 Branch Program ID 4-4
4.2.9 Process Code 4-4
4.2.10 Main Menu 4-4
4.2.11 Sub-Menu1 4-4
4.2.12 Sub-Menu2 4-4
4.2.13 Balloon Help 4-5
4.3 DATA SOURCES 4-5
4.3.1 Alias Name 4-5
4.3.2 Data source 4-6
4.3.3 Master 4-6
4.3.4 Relation Type 4-6
4.3.5 Multi Record 4-6
4.3.6 PK cols 4-6
4.3.7 PK Types 4-7
4.3.8 Parent 4-7
4.3.9 Relation 4-7
4.3.10 Default Where Clause 4-7

4.3.11 Default order by ... *4-7*
4.3.12 Type ... *4-7*
4.3.13 Mandatory ... *4-7*
4.3.14 Data source columns ... *4-7*
4.3.15 Column Name ... *4-8*
4.3.16 Max Length ... *4-8*
4.3.17 Field Name ... *4-8*
4.3.18 Data Type ... *4-8*
4.3.19 Block Name ... *4-9*
4.4 LIST OF VALUES DEFINITION ... 4-9
4.4.1 LOV Name ... *4-9*
4.4.2 LOV Query ... *4-9*
4.4.3 Query Cols ... *4-9*
4.4.4 Data Type ... *4-9*
4.4.5 Visible ... *4-10*
4.4.6 Rdn Fld ... *4-10*
4.4.7 Rdn Fld Type ... *4-10*
4.4.8 Rdn/Col Label ... *4-10*
4.5 DATA BLOCKS ... 4-10
4.5.1 Block Name ... *4-11*
4.5.2 Parent ... *4-11*
4.5.3 XSD Node ... *4-11*
4.5.4 Multi Record ... *4-11*
4.5.5 Block Title ... *4-12*
4.5.6 Relation Type ... *4-12*
4.5.7 Block Type ... *4-12*
4.5.8 Master Block ... *4-12*
4.5.9 Block data sources ... *4-12*
4.6 BLOCK FIELDS ... 4-12
4.6.1 Field Name ... *4-13*
4.6.2 Field Label ... *4-13*
4.6.3 XSD Tag ... *4-13*
4.6.4 Display Type ... *4-13*
4.6.5 Item Type ... *4-13*
4.6.6 Text Area Rows ... *4-14*
4.6.7 Text Area Columns ... *4-14*
4.6.8 Related Block: ... *4-14*
4.6.9 Related Field ... *4-14*
4.6.10 Parent Field ... *4-14*
4.6.11 Min. Val ... *4-14*
4.6.12 Max. Val ... *4-14*
4.6.13 Max Decimals ... *4-14*
4.6.14 LOV Name ... *4-14*
4.6.15 Global LOV ... *4-15*
4.6.16 Offline LOV Name ... *4-15*
4.6.17 Offline Global LOV ... *4-15*
4.6.18 Field Set Name ... *4-16*
4.6.19 Data Type ... *4-16*
4.6.20 Data source ... *4-16*
4.6.21 Max Length ... *4-16*
4.6.22 Field Size ... *4-16*
4.6.23 Column Name ... *4-17*
4.6.24 Default Value ... *4-17*
4.6.25 Copy from Block, Copy from Field ... *4-17*
4.6.26 Access key Code ... *4-17*
4.7 CUSTOM ATTRIBUTES TAB ... 4-17

4.7.1 Attribute Name *4-18*
4.7.2 Attribute Value *4-18*
4.7.3 Selected *4-18*
4.8 EVENTS TAB 4-18
4.8.1 Event Name *4-19*
4.8.2 Function Name *4-19*
4.8.3 Event type *4-19*
4.8.4 Button Screen: *4-19*
4.8.5 Call form Name: *4-20*
4.8.6 Screen Name: *4-20*
4.9 SCREENS 4-20
4.9.1 Screen Name *4-20*
4.9.2 Screen Title *4-20*
4.9.3 Screen Type *4-21*
4.9.4 Main Screen *4-21*
4.9.5 Screen size *4-21*
4.9.6 Exit Button Type *4-21*
4.10 TABS 4-21
4.10.1 Screen Name *4-22*
4.10.2 Tab Name *4-22*
4.10.3 Tab Label *4-22*
4.10.4 Tab Access Key *4-22*
4.11 SECTION AND PARTITIONS 4-22
4.12 FIELD SETS 4-23
4.12.1 Field set Name *4-24*
4.12.2 Field set Label *4-24*
4.12.3 Data Block *4-24*
4.12.4 Multi Record *4-24*
4.12.5 View Type *4-24*
4.12.6 Field set Height *4-24*
4.12.7 Field Set Width *4-24*
4.12.8 No of Rows: *4-24*
4.12.9 Screen Name *4-24*
4.12.10 Screen Portion *4-24*
4.12.11 Tab Name *4-24*
4.12.12 Section Name *4-24*
4.12.13 Partition Name *4-25*
4.12.14 Horizontal FieldSet *4-25*
4.12.15 Read Only *4-25*
4.12.16 Navigation Button *4-25*
4.12.17 Data Block Fields *4-25*
4.12.18 Field Set Fields *4-25*
4.12.19 Sub Partition Name *4-25*
4.13 ACTIONS 4-25
4.13.1 XSD Type Identifier *4-26*
4.13.2 Service Name *4-26*
4.13.3 Operation Id *4-26*
4.13.4 Web Service *4-26*
4.13.5 Action Code *4-26*
4.13.6 Operation Code *4-27*
4.14 AMENDABLE: 4-27
4.14.1 Data blocks *4-27*
4.14.2 New allowed *4-27*
4.14.3 Delete Allowed *4-27*
4.14.4 All Records *4-28*
4.14.5 Mandatory *4-28*

4.14.6 Field Name *4-28*
4.14.7 Amendable *4-28*
4.15 CALL FORMS 4-28
4.15.1 Function ID *4-28*
4.15.2 Parent Block *4-28*
4.15.3 Parent Data source *4-29*
4.15.4 Relation *4-29*
4.15.5 Relation Type *4-29*
4.15.6 Display Type *4-29*
4.16 SCREEN ARGUMENTS: 4-29
4.16.1 Dependent Fields: *4-30*
4.17 LAUNCHFORMS 4-30
4.17.1 Function ID *4-31*
4.17.2 Screen Arguments *4-31*
4.18 SUMMARY 4-32
4.18.1 Title *4-33*
4.18.2 Data Block *4-33*
4.18.3 Data Source *4-33*
4.18.4 Default where Clause *4-33*
4.18.5 Default Order By *4-33*
4.18.6 Summary Type *4-33*
4.19 CUSTOM BUTTONS 4-33
4.19.1 Number of Rows *4-34*
4.19.2 Number of Buttons/Row *4-34*

5. EXTENDED SEARCH OPTION 5-1

5.1 LOCATE FIELD: 5-1
5.2 GOING TO BLOCK FIELD FROM PREVIEW 5-2

6. CHILD AND PARENT FUNCTIONALITY 6-1

7. OPEN DEVELOPMENT- GENERATED FILES 7-1

7.1.1 Front End Units *7-1*
7.1.2 Back End units *7-1*
7.1.3 Release Based *7-1*
7.1.4 Action Based *7-1*

8. STANDARD NAMING CONVENTIONS 8-1

8.1 SCREEN NAME 8-1
8.2 BLOCK NAME 8-1
8.3 FIELD SET NAMES 8-1
8.4 LOV NAMES 8-1

9. CONVERSION PROCEDURE 9-1

10. REFRESH 10-1

10.1 SOURCE REFRESH 10-1
10.2 CHILD REFRESH 10-1

11. SCREEN CUSTOMIZER 11-1

12. NOTIFICATION 12-1

12.1 ACTION 12-1
12.2 NOTIFICATION FUNCTION 12-1
12.3 NOTIFICATION CODE 12-1
12.4 DESCRIPTION 12-1
12.5 OPERATION 12-1
12.6 FILTER TYPE 12-2

12.7 GATEWAY OPERATION 12-2
12.8 GATEWAY SERVICE 12-2
12.9 GATEWAY I/O REQUEST 12-2
12.10 NOTIFICATION MODULE 12-2
12.11 BASE TABLE 12-2
12.12 PK COLS 12-2
12.13 PK TYPES 12-2
12.14 FULL SCREEN REPLY 12-2
12.15 HO FUNCTION 12-2
12.16 FILTER LOGIC 12-2
12.17 FIELD ORDER 12-2
12.18 XSD FIELD 12-3
12.19 TABLE FIELD 12-3
12.20 DATA TYPE 12-3

13. TRIGGER MAINTENANCE 13-1

13.1 TRIGGER CODE 13-1
13.2 DESCRIPTION 13-1
13.3 TRIGGER TIME 13-1
13.4 EACH ROW 13-1
13.5 BASE TABLE 13-1
13.6 PK COLS 13-1
13.7 PK TYPES 13-2
13.8 TRIGGER WHEN CLAUSE 13-2
13.9 SPECIFIC TRIGGERS 13-2
13.10 NOTIFICATION CODE 13-2
13.11 TRIGGER LOGIC 13-2
13.12 SUMMARY 13-2

14. BULK GENERATION 14-1

14.1 SOURCE FOLDER 14-1
14.2 DESTINATION FOLDER 14-1

15. DATA EXTRACTOR 15-1

15.1 ACTIONS 15-1
15.2 OLD OR NEW 15-1

16. VIEW CHANGES 16-1

17. FIELD LOCATOR 17-1

18. DESIGN PRACTICES 18-1

1. Preface

This document describes the features of Oracle FLEXCUBE Management and Integration Console (OPEN DEVELOPMENT tool) and guides the developers on how to use OPEN DEVELOPMENT tool.

1.1 Audience

This document is intended for FLEXCUBE Application developers/users that use OPEN DEVELOPMENT tool to develop various FLEXCUBE components.

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

Proficiency	Resources
FLEXCUBE Functional Architecture	Training programs from Oracle Financial Software Services.
FLEXCUBE Technical Architecture	Training programs from Oracle Financial Software Services.
FLEXCUBE Object Naming conventions	Development Overview Guide
Working knowledge of Web based applications	Self Acquired
Working knowledge of Oracle Database	Oracle Documentations
Working knowledge of PLSQL developer	Respective vendor documents
Working knowledge of PLSQL & SQL Language	Self Acquired
Working knowledge of XML files	Self Acquired

1.2 Related documents

For more information on OPEN DEVELOPMENT development, see these resources:

- Overview Guide
- Getting Started
- Function ID Development.doc
- Web Service Development
- Notification Development
- BIP Report Integration

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 forwhich 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

2. Introduction

2.1 How to use this Guide

The information in this document includes:

- Chapter 2 , “Introduction”
- Chapter 3 , “Overview of Oracle FLEXCUBE Management and Integration Console”
- Chapter 4 , “Screen Design”
- Chapter 5 , “Extended search option”
- Chapter 6 , “Child and Parent functionality”
- Chapter 7 , “OPEN DEVELOPMENT- generated Files”
- Chapter 8 , “Standard Naming Conventions”
- Chapter 9 , “Conversion procedure”
- Chapter 10 , “Refresh”
- Chapter 11 , “Screen customizer”
- Chapter 12 , “Notification”
- Chapter 13 , “Trigger Maintenance”
- Chapter 14 , “Bulk Generation”
- Chapter 15 , “Data Extractor”
- Chapter 16 , “View Changes”
- Chapter 17 , “Field Locator”
- Chapter 18 , “Design practices”

3. Overview of Oracle FLEXCUBE Management and Integration Console

The FLEXCUBE MICON (Management and Integration Console) has been designed to develop the different types of screens in FLEXCUBE. This tool has been designed to assist developers in designing screens with the capability of generating front end scripting files, PL/SQL Packages, Static data scripts and XSDS.

This generated code performs validations and does processing which is common across screens in FLEXCUBE, only the Business logic specific to the screen need to be added by the Developer in back end and front end units.

Oracle FLEXCUBE Management and Integration Console have to be deployed for each Release and the tool identifies the Release type and the release number.

For Example

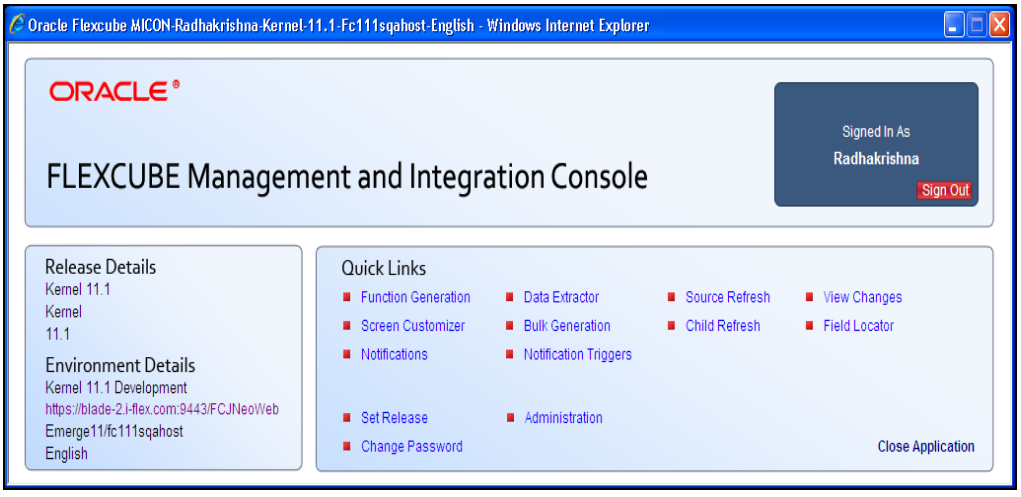
Release Name: FC 11.1

Release Type: KERNEL, CLUSTER or CUSTOM

Tool will generate all files and developers are supposed to add the business logic in designated units depending on the Release Type.

Using the Oracle FLEXCUBE Management and Integration Console user can achieve screen design, Source Refresh, Child Refresh, Data Extractor, Bulk Generation, Notification, Trigger Maintenance, Screen Customizer, View Changes and Field Locator.

Below is the screen of Oracle FLEXCUBE Management and Integration Console, which has all the links for above feature and same can be lunched from Tool link.



Title of the above screen shows the user about Release name and Release type of the development.

3.1 Release Details:

The Release details will gives the user information about the Release Name, Release Type and Release No. in which user is working.

3.2 Environment Details:

The Environment details gives the user information about the about Schema, Instance in which the user is working and Application URL and language in which the user is currently working.

4. Screen Design

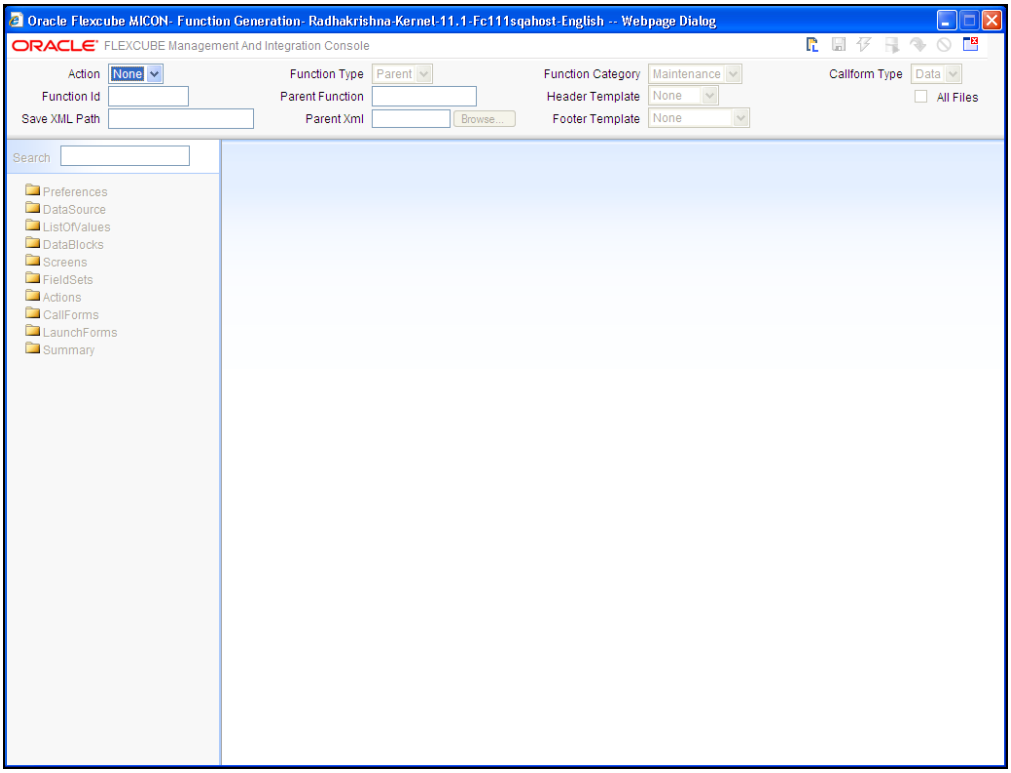
This section describes Function ID (Screen) design in FLEXCUBE MICON.

The following are the stages involved in designing the FLECUBE Screen.

- Preferences
- Data sources
- Data blocks
- Screens
- Tabs
- Sections
- Field Sets
- Actions
- Call forms
- Launch forms
- List Of Values
- Summary

4.1 Basic Details

Below is the screen used to design the FLEXCUBE screen and same can be launched by clicking on “Function Generation” link of the Tool main page.



4.1.1 Action

The field indicates that, the screen to be designed or to be modified .Action New is for new screen design and load is for screen modification.

4.1.2 Function Id

Captures function Id Name and name can be of FLEXCUBE standard naming convention for the function id.

4.1.3 Save Xml Path

This is the path where the XMLs and JS files have to be saved in case of anew Functions. In case of loading an existing Function, Oracle FLEXCUBE Management and Integration Console saves the files in the same directory.

The label description of the field will change depends on the action, if the action is load tool changes the label description of the field to Load Screen xml and attaches a Brows button to it so that user can browse the OPEN DEVELOPMENT xml and load it.

4.1.4 Function Type

Function Type can be Parent or Child (based on the screen which you are going to design). Refer child and parent functionality described in this document.

4.1.5 Parent Function

This field is applicable only if the function type is child and this field will be populated when the parent OPEN DEVELOPMENT xml is loaded.

4.1.6 Parent Xml

This field is also applicable only if the Function type is child. If the Function Type is child user has to load the OPEN DEVELOPMENT xml of the Parent Function using browse button provided to this field. It is non editable field if the action is “Load”

4.1.7 Function Category

Function category is based on your screen whether it is Maintenance or Transaction (Online), generated packages will be different for Maintenance and Transaction.

4.1.8 Header Template

A template can be selected for header. Depends on the template selected, tool will generate screen. Header Template can be Process for BPEL screens.

4.1.9 Footer Template

Tool has the options to design the audit trail block of the screen. One is select the Template and let the Tool to design audit trail of the screen depends on the selection, Maintenance, Maint Process and Process, depends on the selection tool design the audit trail block. If none is selected then user has to design audit trail block.

Transaction (online) screen Footer Template should be none and user has to design the audit trail block.

4.1.10 Deploy

Deploy Option allows the user to deploy both the front end and backed units generated by FLEXCUBE MICON. It deploys the FUNCTIONID.xml and FUNCTIONID_SYS.js in the ENG xml and JS path which is specified at the environment level .It Loads all the SPC, SQL, INC Files generated in the Scheme which is specified at the environment level.

4.1.11 Release

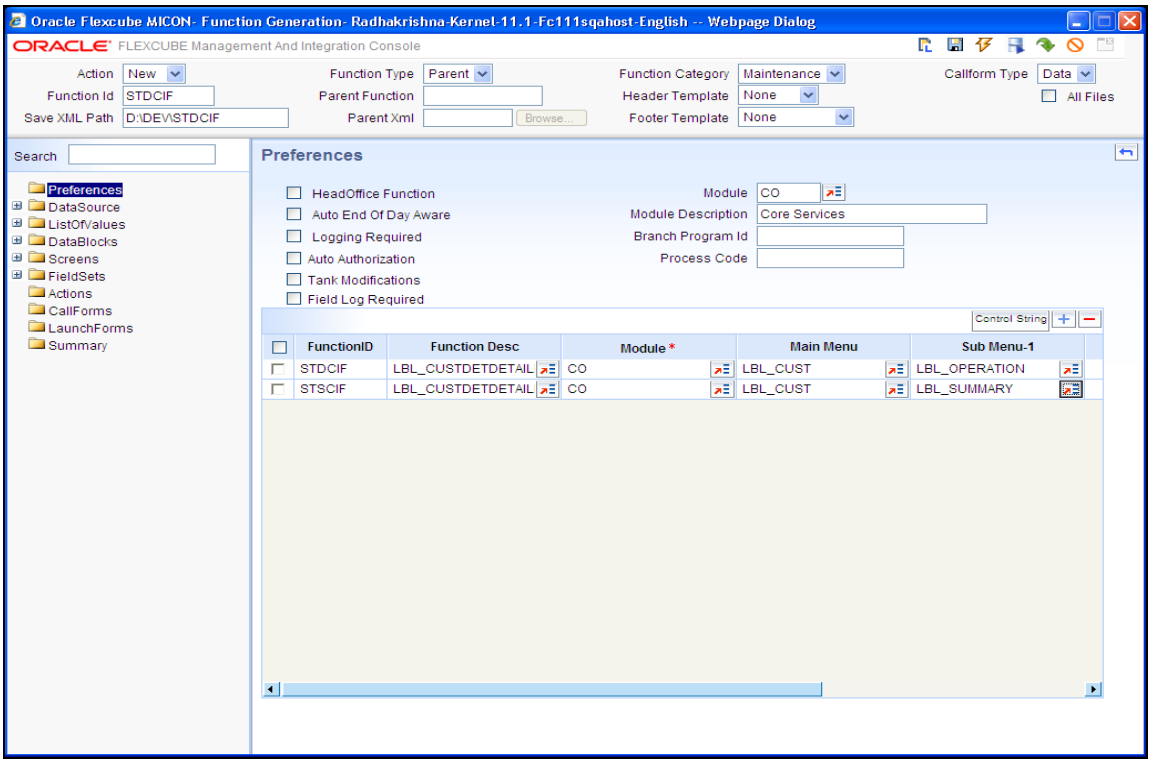
Release option gets the Check in information like check in mode, sfr no, site code etc and checks in the units.

4.1.12 All Packages

Restrictions can be done on package generation from Oracle FLEXCUBE Management and Integration Console. If this check box is selected all packages will be generated including cluster, custom and kernel else only main packages will be generated.

4.2 Preferences

Function id level preferences like module, logging required, tanking modification Main menu, Sub-Menu1, Sub-Menu2, etc. can be given here, below is the screen for the same.



4.2.1 Module

It captures the Module of the function id.

4.2.2 Head office Function

It captures whether the function is head office function or not.

4.2.3 Logging Required

It captures whether logging is required for the function or not.

4.2.4 Auto AEOD Aware

It captures whether function is Auto AEOD Aware or not.

4.2.5 Auto Authorization

It captures whether Auto Authorization is allowed for the function or not.

4.2.6 Tank Modification

It captures whether modification to be tanked for the function or not.

4.2.7 Field Log Required

It captures whether filed Logging is required for the function or not.

4.2.8 Branch Program ID

It captures branch program id for the function.

4.2.9 Process Code

It captures process code for the function.

4.2.10 Main Menu

Main menu of the function can be specified here.

4.2.11 Sub-Menu1

Sub-menu1 of the function can be specified here.

4.2.12 Sub-Menu2

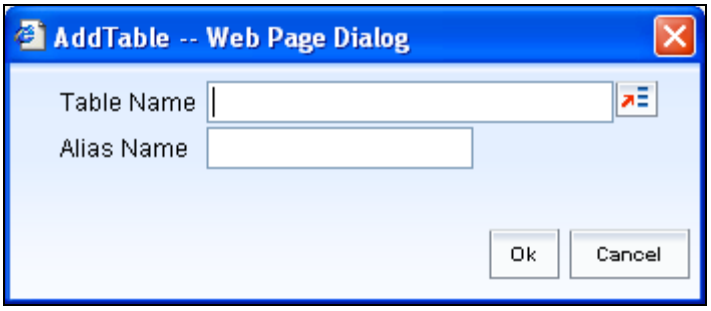
Sub-menu2 of the function can be specified here.

4.2.13 Balloon Help

Balloon Help of the function can be specified here.

4.3 Data sources

Data Sources are the Tables and views used in the function. Data sources can be added to design the screen by right clicking on the data source of Oracle FLEXCUBE Management and Integration Console Tree. Data source can be entered or can be selected from list of values which is attached to the table Name field. As shown in the below screen.

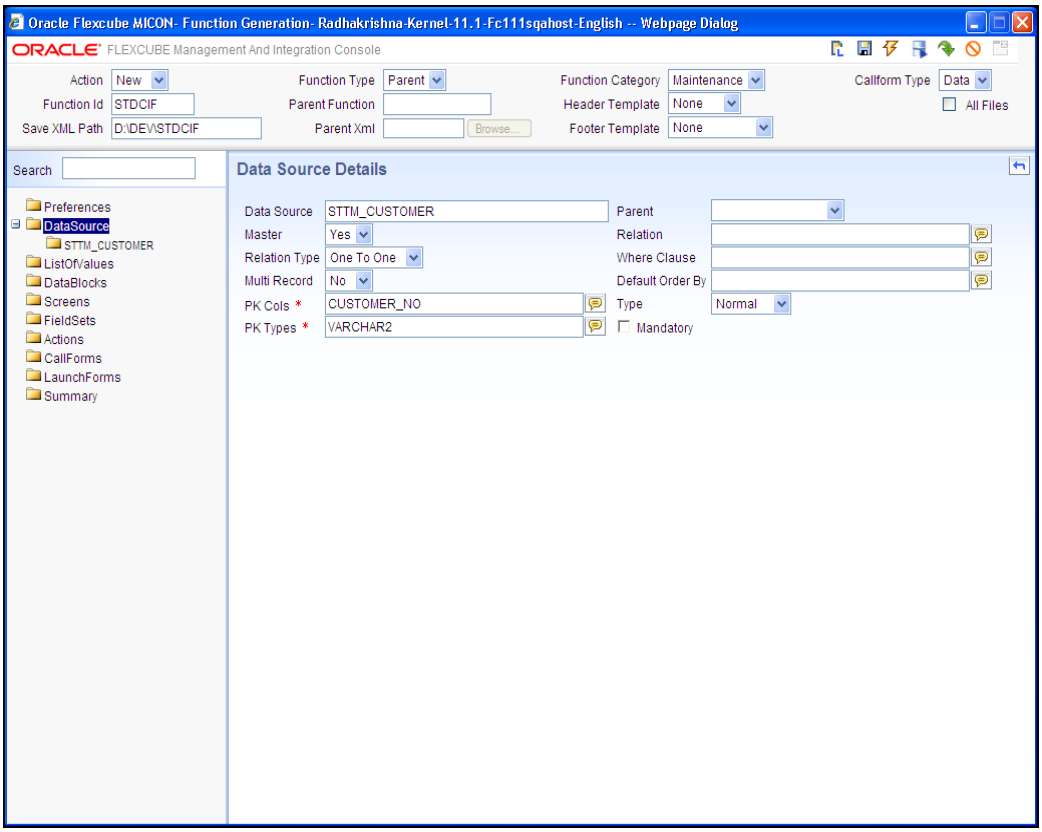


4.3.1 Alias Name

An alias can be maintained for a data source in this field.

Before adding a data source user has to maintain the primary key column for the data source in STTB_PK_COLS.

Below is the screen where you can define the properties of a data source selected



4.3.2 Data source

This is a non editable field, which shows the data source name selected.

4.3.3 Master

One master data source is mandatory for designing a screen and same can be maintained here.

4.3.4 Relation Type

Relation type can be selected for the selected data source with the parent, “One To One” or “One To Many“ , the relation should be one to many if the parent data source is Multi Record entry and child data source is single Record.

4.3.5 Multi Record

This field tells about the type of data source in the screen, whether it is multiple Entry record or the single Entry Record data source, Select the field value “Yes” or “No” accordingly.

4.3.6 PK cols

The value of this field will be picked from the STTB_PK_COLS, if it is maintained, and Tool will allow the user to override the value and same value will be used in generating the package.

4.3.7 PK Types

The value of this field will be picked from the STTB_PK_COLS, if it is maintained, and Tool will allow you to override the value and same value will be used in generating the package.

4.3.8 Parent

This field will allow the user to select a Parent Data source for the current Data source, this relation will be used in generated packages.

Same Data Source Cannot be a parent to itself.

4.3.9 Relation

Relation between current data source and its parent data source can be maintained in this field.

4.3.10 Default Where Clause

You can define the where clause for the data source. The values of where clause field will be used in generating the query statements for current data source

4.3.11 Default order by

Captures the default where clause for the data source. The values of where clause field will be used in generating the query statements for current data source in generated packages.

4.3.12 Type

Data source can be selected depends on the Type requirement of the screen design.

Below are the functionalities of the Types.

- **Normal**: Tool generates packages with insert and updates statements with this data source and infra will build the request and response.
- **Query:** The data source can be used for only querying the data; Tool will generate packages without insert or update statements with this data source.
- **InOnly:** Tool will generate the packages without insert and update with this data source and infra will build only request with this data source. Response also will be build based on the values received from the Front End.
- **Summary**: This data source can be used for summary and this data source will not be considered while generating the packages.

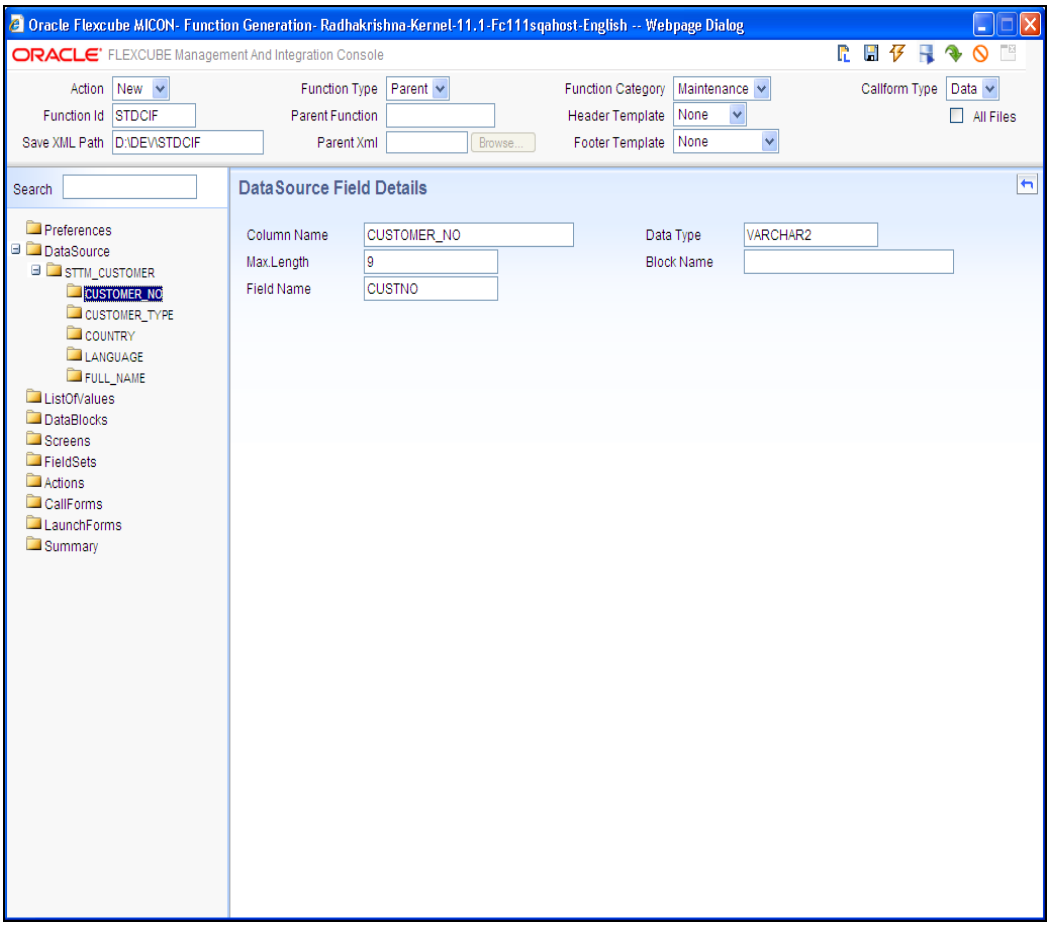
4.3.13 Mandatory

Data source should have one record/value.

4.3.14 Data source columns

Data source column will be added by right clicking on the data source.

Below is the screen where column properties can be maintained.



4.3.15 Column Name

This is an information column; tool will not allow the user to change the value.

4.3.16 Max Length

The max length of column will be defaulted while adding a column and it can be overwritten. This value will be considered for the field max length of the designed screen. It means, at the run time system will not allow the user to enter the text more than this length.

4.3.17 Field Name

Field name will be populated while adding the column, which is maintained in CSTB_DATA_DICTIONARY table. Non editable field.

4.3.18 Data Type

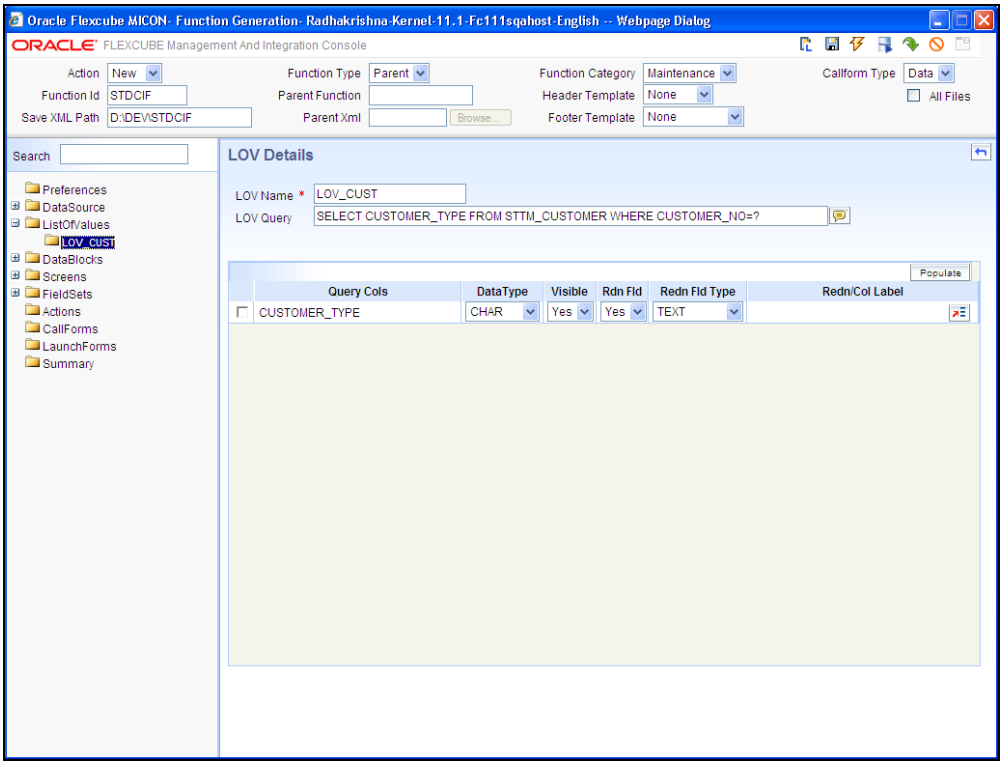
This shows about data type of the column selected and value will be populated while adding the column, this is a non editable field.

4.3.19 Block Name

It is an information field, if the field is added to a block, that block name will be populated here.

4.4 List of values definition

LOV's can be defined using the below screen, and same can be launched by right click on ListOfValues of Tree in Oracle FLEXCUBE Management and Integration Console.



4.4.1 LOV Name

Name of the LOV, a non editable field and same can be changed using rename option of the Tool.

4.4.2 LOV Query

Captures the LOV query and band variables can captured with "?".

4.4.3 Query Cols

Depends on the lOV Query columns will be populated by clicking the populate button.

It's a non editable field.

4.4.4 Data Type

Data type will be populated depends on the data type of the column in the data base, this can be overwritten.

4.4.5 Visible

A result column of the LOV can be made as invisible by using this filed.

4.4.6 Rdn Fld

A field can be made as reduction of non reduction by using this flag.

4.4.7 Rdn Fld Type

Display type of the reduction field can be maintained. Display type of the reduction field can be either of the below given list

- TEXT
- CHECK BOX
- RADIO
- SELECT

4.4.8 Rdn/Col Label

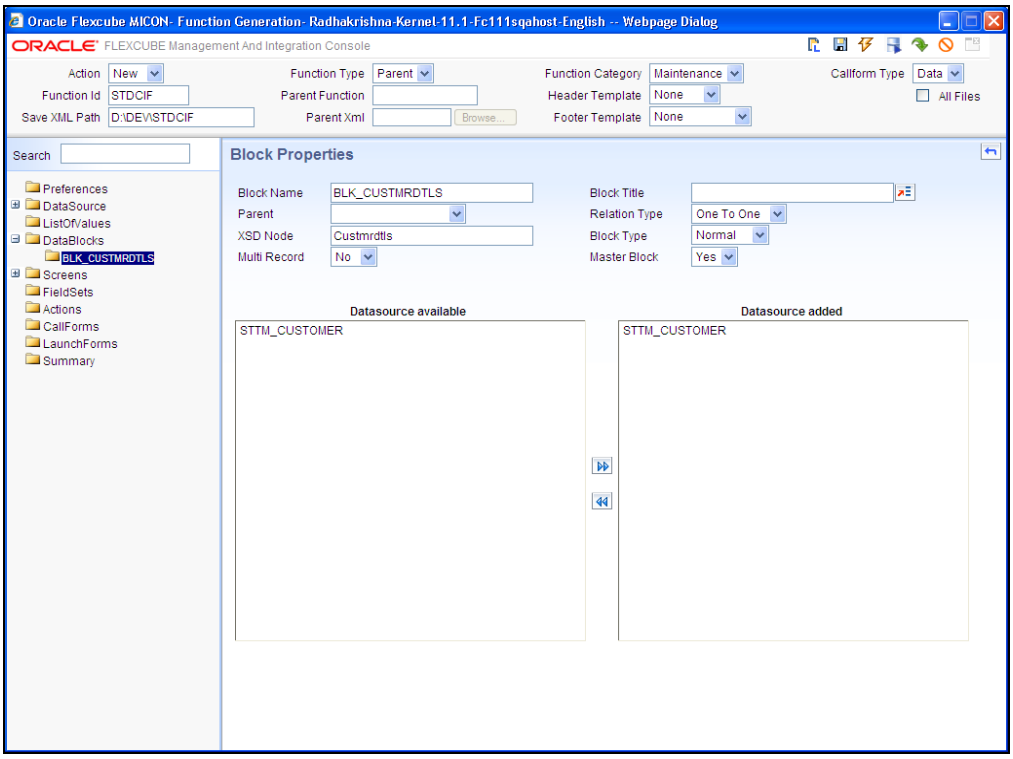
Title for each field can be maintained as Label code, and label code can be selected from the list of values attached to the field.

Bind variables and return fields of a LOV are maintained at block field properties.

4.5 Data Blocks

Data Block is a logical grouping of columns from one or more data sources. Below is the procedure of defining Data blocks in Oracle FLEXCUBE Management and Integration Console.

Below is the screen where Data Block Properties can be maintained.



4.5.1 Block Name

This field will be defaulted with block name which is added and this value cat be changed the user.

4.5.2 Parent

This field will be populated with all the data blocks which are added, and you can select the parent data block for the current block .Ultimate parent should not have the parent data block and this information will be used while generating the front units and same will be used by FLEXCUBE to build request xml.

4.5.3 XSD Node

This field captures the xsd node. Tool will use this name for generation of node in xsd's .

4.5.4 Multi Record

Using this field you can decide block type, whether the block is single entry block or multiple entry blocks.

Depends on this value, the field "data sources available" will be populated.

If the value is "Yes" the field "data source available" will be populated with data sources which have flag Multi record "Yes" and vice versa.

4.5.5 Block Title

You can select the block title here by select the label code from list of values, and this title will be displayed on the block.

4.5.6 Relation Type

This field captures the Type of the relation, One to One or One to Many and this information will be used while generating the front units and same will be used by FLEXCUBE to build request xml.

4.5.7 Block Type

Type of the block can be selected depends on the requirement, below are the description of the each type.

- **Control:** if the block is used for only UI purpose and this block data is not going for the back end to process the data at run time. Infra will not build the request and response with this block and OPEN DEVELOPMENT will not populate any data sources to select
- **Normal:** Block which is used in the back end to process that data. Infra builds the request and response with this block.
- **Summary:** Block which is used for creating the summary screen.

4.5.8 Master Block

One master block is compulsory while designing the screen and using this flag you can define a block as master. This parameter will be using in generating xsd's , packages and front end units .

4.5.9 Block data sources

Depends on the "Multi Record" flag, data sources will be populated. Then attach the data sources required to the block.

Validations for populating the data source:

Multi Record: Yes

All the data sources which have "Multi Record" flag "YES" and "Relation Type" One To One

Multi Record: No

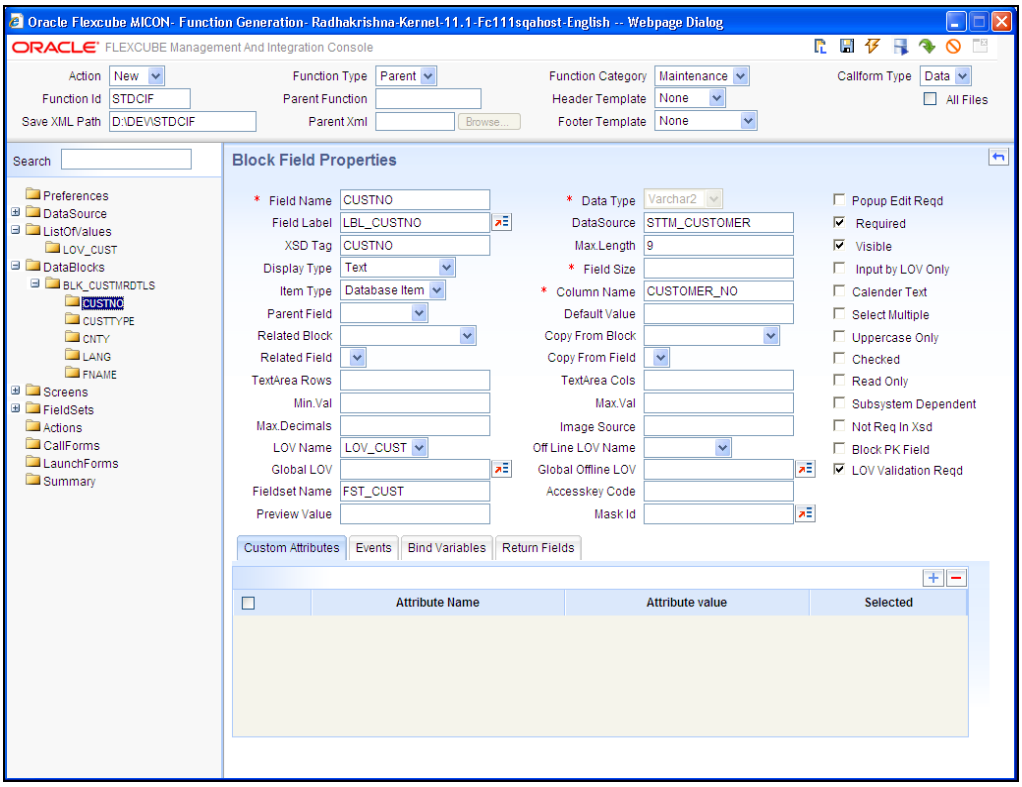
All the Data source which have "Multi Record" No

While attaching Data Sources to Blocks, multiple data sources should be selected only if such data Sources can be functionally clubber. For examples CSTM_PRODUCT and LCTM_PRODUCT_DEFINITION. Clubbing should not be arbitrary.

4.6 Block fields

Block fields can be added to a block by right clicking on block, all the data sources and respective fields will be available to add to block.

Below is the screen where field properties can be entered



4.6.1 Field Name

Name of the field and same name will be used in front and back end units generated. Name of the field can be changed by renaming the field

4.6.2 Field Label

This will be defaulted when field is added, label code also be added using the list of values. This value will be used for the label of the field in the screen.

Note: If you enter the LBL code manually, it should be with below given format

“LBL_ xsd tag Name”

4.6.3 XSD Tag

It will be defaulted field is added and whenever label code got modified. This value is used in the xsd’s as xsd tag and same will be used for web service.

4.6.4 Display Type

Pre-defined values will be available in the select box for this field. Value can be selected on requirement. This is the value which decides the display of the field in the screen where it is a check box, Drop down box or a radio so on.

4.6.5 Item Type

The value of this field will be defaulted to Database Item, if the field has taken from the data source, or default values are Control and Desc if the field is a UI field.

The field will be generated in the package and front end (SYS.js) only if the item type is Database item.

4.6.6 Text Area Rows

If the display type is text area, height of the text area can be mentioned. Using this field user can increase the height of the Text Area in the screen.

4.6.7 Text Area Columns

If the display type is text area, width of the text area can be mentioned. Using this field user can increase the width of the Text Area in the screen.

4.6.8 Related Block:

An amount field can be related to a particular currency field, and this field captures the block name where the currency field is present.

4.6.9 Related Field

Subsequent to the above field (Related Block), a field can be selected as a related field for an amount field.

Selected Block's (Related Block) fields will be available in the select box.

4.6.10 Parent Field

If the Item Type of the selected field is Desc, a parent field should be attached and the selected parent field should have a LOV attached to it. And the current field should be one of the return fields of the attached LOV.

This feature is provided to avoid Query data sources and should be used for all description Fields. If a particular field is selected as "Description Field" in OPEN DEVELOPMENT, The LOV for the Parent field should not have any Control Item as a bind variable.

4.6.11 Min. Val

Minimum value to a number field can be mentioned, tool will generate the validation of the min value for the filed in both front and back end.

4.6.12 Max. Val

Maximum value to a number field can be mentioned, tool will generate the validation of the max value for the field in both front and back end.

4.6.13 Max Decimals

Max Decimals to a number or amount field can be mentioned.

4.6.14 LOV Name

All the LOV names will be available which are defined in ListOfValues definition,

A LOV can be selected and attached to a field.

4.6.15 Global LOV

Global LOV can be maintained across function id and same can be attached to a field. This will reduce development time. LOV can be selected from list of values attached to the field .These LOV are handled by the FLEXCUBE Infra.

Global LOVs are stored in **CSTB_LOV_INFO** WITH FUCNTION ID AS **COMMON .**

4.6.16 Offline LOV Name

These LOV will be used by the Branch function id if the branch is in offline mode, offline LOV will be used. All the LOV will be populated which are defined in the LOV definition screen.

Both offline and online LOV can be maintained for the same field.

4.6.17 Offline Global LOV

Defining of these LOV remain same as Global LOV, and functionality is same as offline LOV

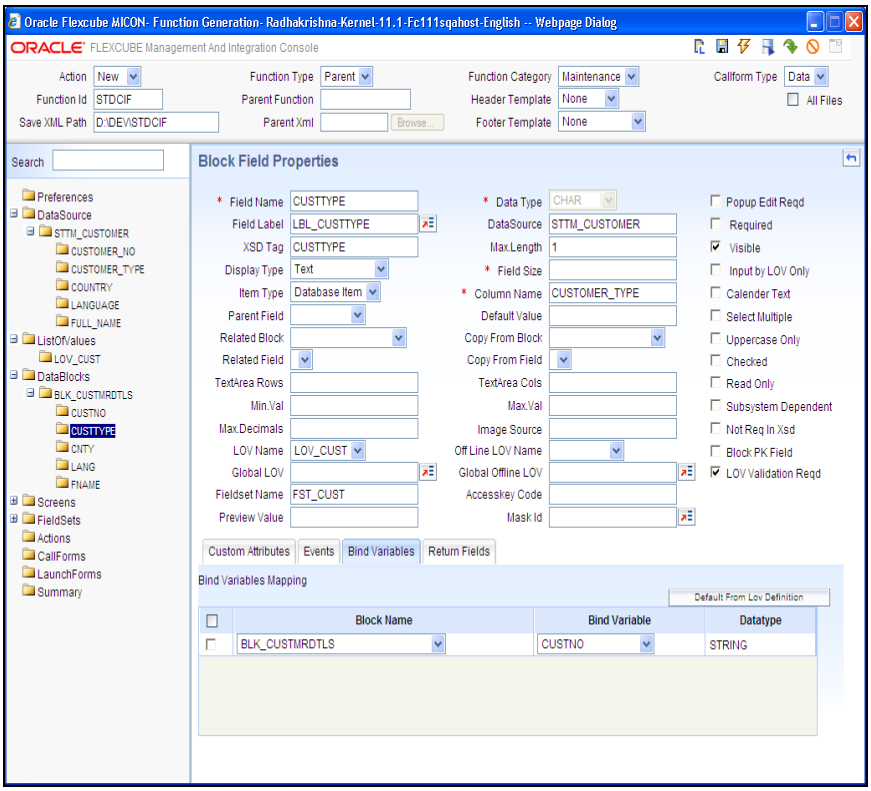
In a given point in time tool will allow the user to select either local LOV (LOV Name, Offline LOV Name) or Global LOV (Global LOV, Global Offline LOV).

Return fields and bind variables can be defined in the “Bind variables” tab and “Return fields” tab as shown below.

The button

will populate the bind variables and return fields from the LOV definition.

Any field across the blocks can be attached as LOV bind variable or return field.



4.6.18 Field Set Name

It is an info field, tells user about the name of the field set where the field is selected.non editable field.

4.6.19 Data Type

Non editable field, value will be defaulted while adding the field this values be considered while generating the packages and xsd's for web service.

4.6.20 Data source

Information field to the user, value will be defaulted while adding the field if the field is database item and value cannot be changed .

4.6.21 Max Length

The max length of column will be defaulted while adding a field and it can't be overwritten.

This value will be considered for the field max length of the designed screen. It means, at the run time system will not allow the user to enter the text more than this length.

The value of the field will be changed from data source column properties.

4.6.22 Field Size

Value will be defaulted while adding the field and same can be modified. This is the size which will reflect in the screen for the field.

4.6.23 Column Name

Non editable field, value will be defaulted while adding the field if the field is database item.

4.6.24 Default Value

A default value can be given to the selected field , system will use the default value at the run time and default value will be stored in the generated front end unit , it will not available in the generated packages.

4.6.25 Copy from Block, Copy from Field

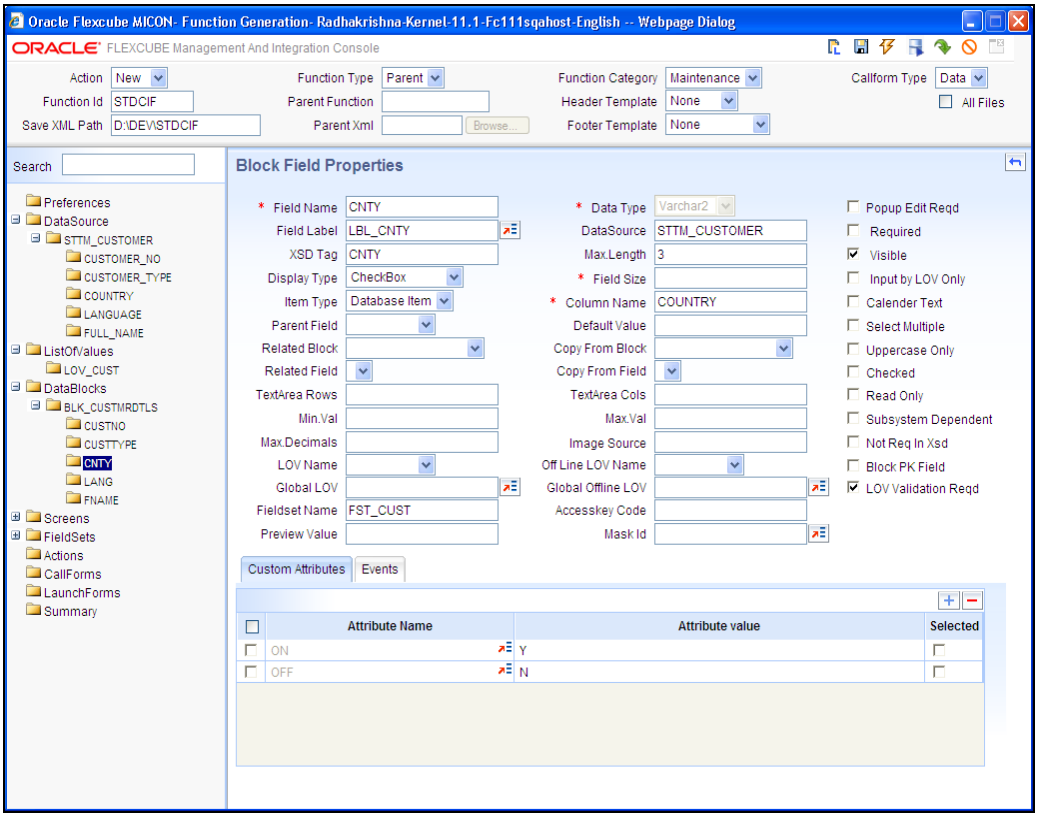
At the run time field values can be copied from one field to another field, without writing extra code, same can be achieved by selecting copy from block and copy from field.

4.6.26 Access key Code

A short cut key can be created to a field by adding the value in this field and created short cut key can be used at the run time to access the field.

4.7 Custom Attributes TAB

If a screen has the select box (Drop down box) or the radio button , name and value pair should be maintained for such fields , same can be captured under customer Attributes tab of the field properties as shows in the below screen.



4.7.1 Attribute Name

Captures the Name of the attribute and same can be selected from list of value attached to the field.

For Ex: For check box Name can be ON or OFF.

4.7.2 Attribute Value

Captures the value of the attribute and same can be selected from list of value attached to the field.

For Ex: For check box value can be Y or N.

4.7.3 Selected

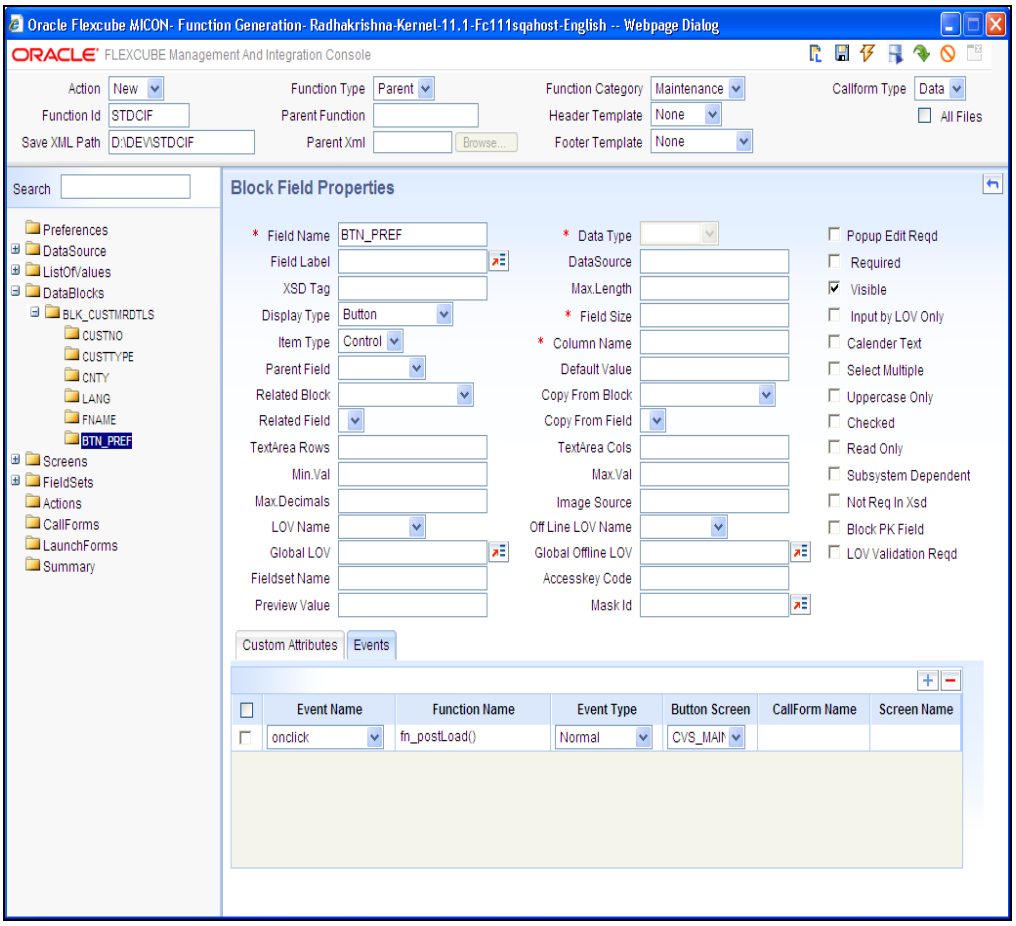
If the selected check box is checked, the particular value be default value.

If display type is check Tool will default two row in the custom attributes tab with ON and OFF names, Y and N values.

Custom attributes cab be maintained if Display Type is check box, select and Radio.

4.8 Events Tab

Field events can be maintained in the events tab and same is show in the below screen.



4.8.1 Event Name

A pre-defined java script event can be attached to the field and same can be selected.

4.8.2 Function Name

A function name should be mentioned for event, same function should be maintained in the function id java script or in the infra java script files.

Note: Not required to mention the function name if the event type is call form or sub screen.

4.8.3 Event type

A pre-define event type can be selected and event type are as below

- **Normal**: On selected event the mentioned function will be called.
- **Callform**: On selected event call form will be launched.
- **Subfunction**: On selected event another main form will be launched.
- **Subscreen**: On selected event sub screen will be launched.

4.8.4 Button Screen:

Call form button can be added to main or sub screens and same can be maintained using button screen filed. This field will be populated with all the screens created.

4.8.5 Call form Name:

If the event type is call form, call form Name should be maintained and value would be the call form function id name

4.8.6 Screen Name:

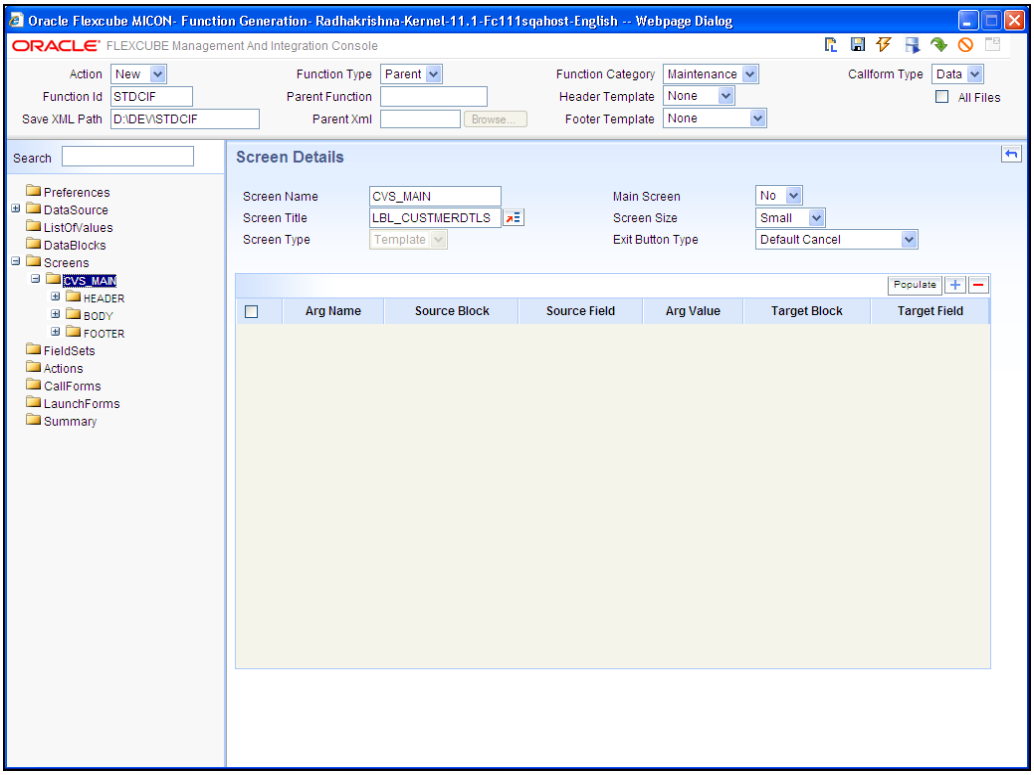
If the event type is call form or sub screen, screen name should be maintained and same screen will be launched for the event attached.

4.9 Screens

Here in this section Oracle FLEXCUBE Management and Integration Console, Screen layout has to be designed and below is the Procedure to do the same.

By right clicking on “Screens” of the Tree portion of the Oracle FLEXCUBE Management and Integration Console, screen will be added.

Below is the screen where the screen properties can be maintained



4.9.1 Screen Name

This field value will be defaulted when screen is added and it is a non editable field.

4.9.2 Screen Title

This field will have a label code for the screen title to be displayed. Label code can be selected from list of values button.

4.9.3 Screen Type

This value should be always Template, Tool have template based approach to design the screen. That is, to design the screen user has to create sections of the screen depends on the requirement and each section should have partitions and each partition will have field sets, and all the fields will fall under the field sets, this is the template and same should be followed for all the screens in the FLEXCUBE.

4.9.4 Main Screen

One main screen is compulsory to design a screen. If the selected screen is the main screen then select the value as “Yes” else “No”. If the screen is main screen, only that screen will be launched from FLEXCUBE.

4.9.5 Screen size

Depends on the number of fields of the screen size can be selected. Available sizes are Small, Medium and large.

4.9.6 Exit Button Type

This field tells about exit and ok buttons to the screen, if you want to have only cancel button then select Default cancel, none option also available.

4.10 Tabs

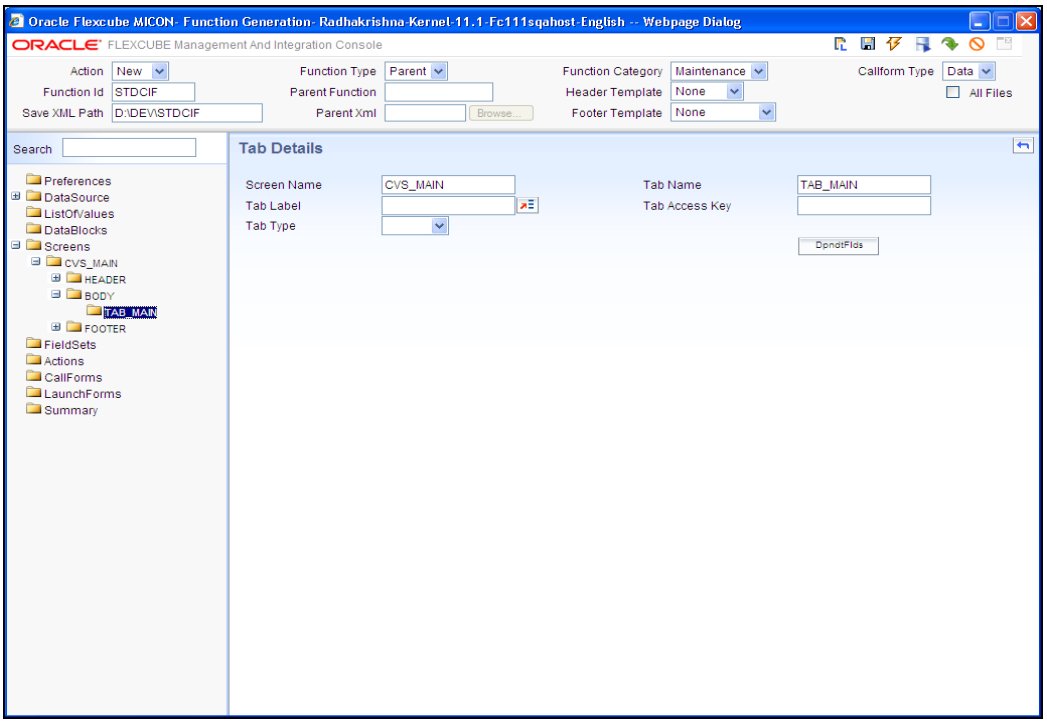
When user creates a screen in the Open Development, Tool will create three default portions in the screen

- Header
- Body
- Footer

FLEXCUBE screens are divided into three portions, Header, Body and Footer and the all portions can have tabs and inside tabs fields.

Each portion will have default tabs. Along with these default tabs you can create tabs

Below is the screen where tab properties can be defined



4.10.1 Screen Name

This is information filed and the value will be defaulted when tab is added, and name will be name of the screen under which screen, the tab has been created and it is a non editable field

4.10.2 Tab Name

This is information filed and the value will be defaulted when tab is added, it is a non editable field.

4.10.3 Tab Label

This field allows you to add a Name of the Tab, label code can be selected from the list of values.

4.10.4 Tab Access Key

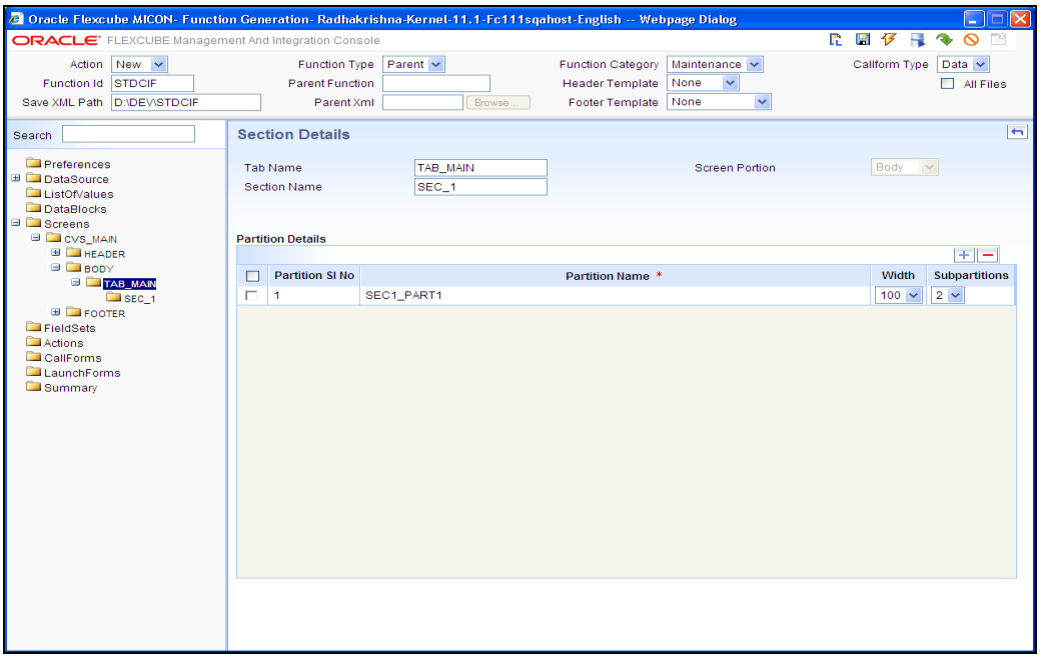
Tool will allow you to select a short cut key to the tab and same value can be used at run time to access the tab.

4.11 Section and Partitions

Section can be created under each tab. Number of section vary from screen to screen .sections can of any number.

Partitions can be added to the each section, depends on the screen size (Small, Medium and Large) tool will allow you to add partitions and sub partitions.

Partitions can be added using buttons.

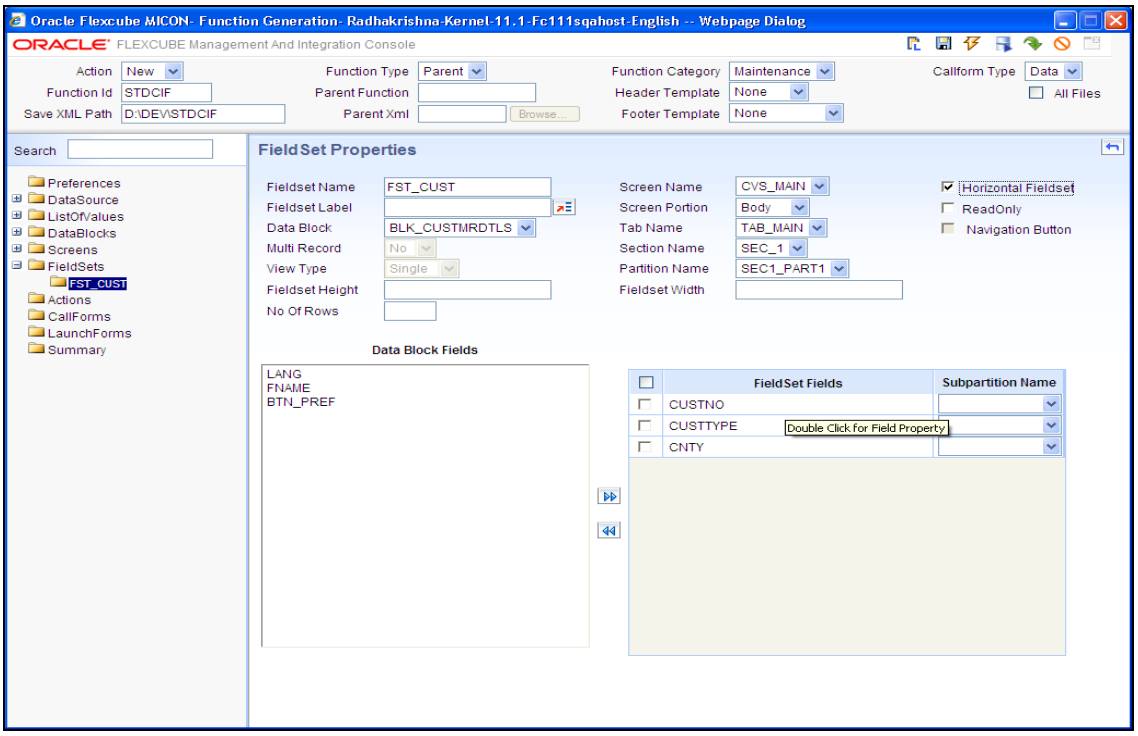


4.12 Field sets

Field set is a logical set of fields which would appear together in one partition of the screen layout.

Field sets can be created using Open Development Tree, by right clicking on Field sets.

Below is the screen where field set properties can be defined.



4.12.1 Field set Name

Non editable field, this can be changed using rename option of the Tool.

4.12.2 Field set Label

All field set can have a title to them, and same can be mentioned in this field as label code. Label code can be selected using the list of values attached to the field.

4.12.3 Data Block

A block can be selected among populated, all field will be populated which are belong to the selected block to add to the field set.

4.12.4 Multi Record

A non editable field, this will be defaulted when block selected. This is the parameter defined at the block properties level.

4.12.5 View Type

View type of the field set can be selected here depends on the requirement, single or Multiple view.

4.12.6 Field set Height

Captures the height of the fields set.

4.12.7 Field Set Width

Holds the width of the fields set.

4.12.8 No of Rows:

If view type is multiple and number of rows in the multiple table can be decided using No of rows field and this is optional field, Tool will default to 15 rows, if the field value is null.

4.12.9 Screen Name

This field captures the name of the screen where the particular field set to be added.

4.12.10 Screen Portion

This field captures the name of the screen portion where the particular field set to be added.

4.12.11 Tab Name

This field captures the name of the tab under which the selected field set to be added.

4.12.12 Section Name

This field captures the name of the section in which the selected field set to be added.

4.12.13 Partition Name

This field captures the name of the partition in which the selected field set to be added.

4.12.14 Horizontal FieldSet

A field set can be horizontal if this flag is selected.

A horizontal field set will have fields horizontally arranged, where as in normal field sets all the fields will be arranged vertically by default.

4.12.15 Read Only

Entire block can be made read only , if this check box is selected.

4.12.16 Navigation Button

This is applicable only for multi record is yes and selected view type is Single. If this check box is selected, tool will add previous and next buttons to the screen.

4.12.17 Data Block Fields

Depends on the block selected this field will be populated with all the fields which are belong the blocks and not attached to any of the field set.

4.12.18 Field Set Fields

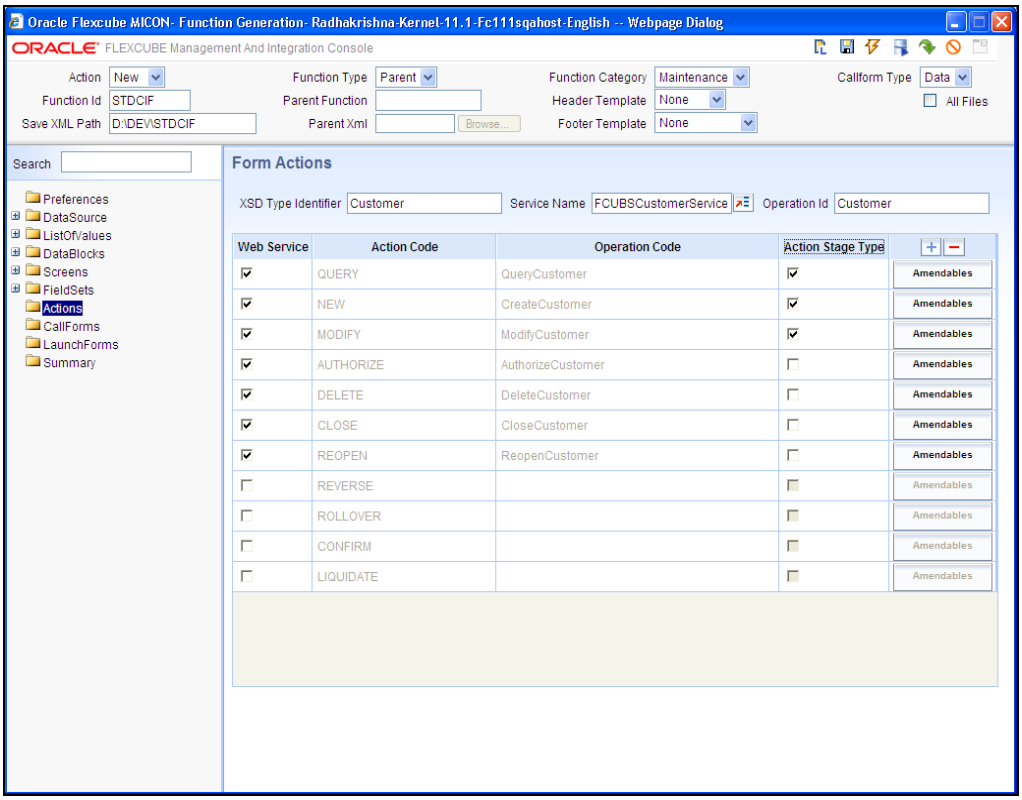
Fields can be selected to the field set from the Data block Fields.

4.12.19 Sub Partition Name

If the selected partition has any sub partitions created, then sub partition field will be populated with number of the sub partitions, field can be selected to particular sub partition.

4.13 Actions

Web Service related information and FLEXCUBE amendable fields can be captured in the actions screen show below.



4.13.1 XSD Type Identifier

This field captures the name, using this name xsd type will be generated for the function, and tool will generate xsd file names using this name. This Should be a descriptive code of the Function being developed.

4.13.2 Service Name

Web service should be registered with a server Name and that server name can be mentioned here, and service name can be selected using the list of values attached to the field.

Tool will use the server name while generating the xsd files.

4.13.3 Operation Id

Each operation code will be defined using the operation id, the name can a self descriptive.

4.13.4 Web Service

User can decide what all the operations are available for web server by selecting this check box and tool will generate xsd only for those operations.

4.13.5 Action Code

Pre-define actions will be available for the action code. These are the operations which are available in the FLEXCUBE.

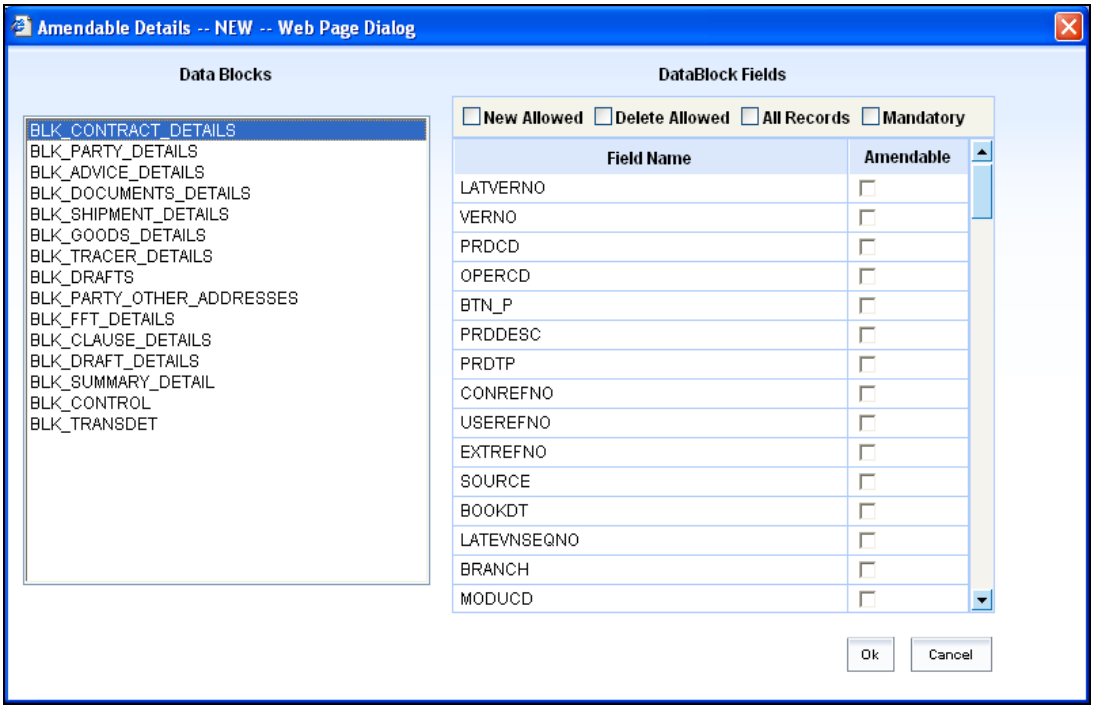
4.13.6 Operation Code

If the web service is selected, the operation code will be defaulted and operation code will be the combination of action code and Operation Id and same operation code will be used to perform operations for web service.

4.14 Amendable:

In FLEXCUBE amendable fields are pre-defined and those fields can be modified at the time of modification operation and same can be maintained using the Oracle FLEXCUBE Management and Integration Console at the time of designing the screen.

Amendable fields can be maintained at actions level, below is the screen where amendable can be maintained



4.14.1 Data blocks

This field contains all the blocks which are created for the screen.

4.14.2 New allowed

A new row can be added to a multiple entry block at the time of modification, if new allowed check box is selected.

4.14.3 Delete Allowed

A row can be deleted from a multiple entry block at the time of modification, if Delete allowed check box is selected.

4.14.4 All Records

All the rows can be deleted from a multiple entry block at the time of modification, if All Records check box is selected.

4.14.5 Mandatory

In xsd a node (element) can be mandatory, if the Mandatory check box is selected.

4.14.6 Field Name

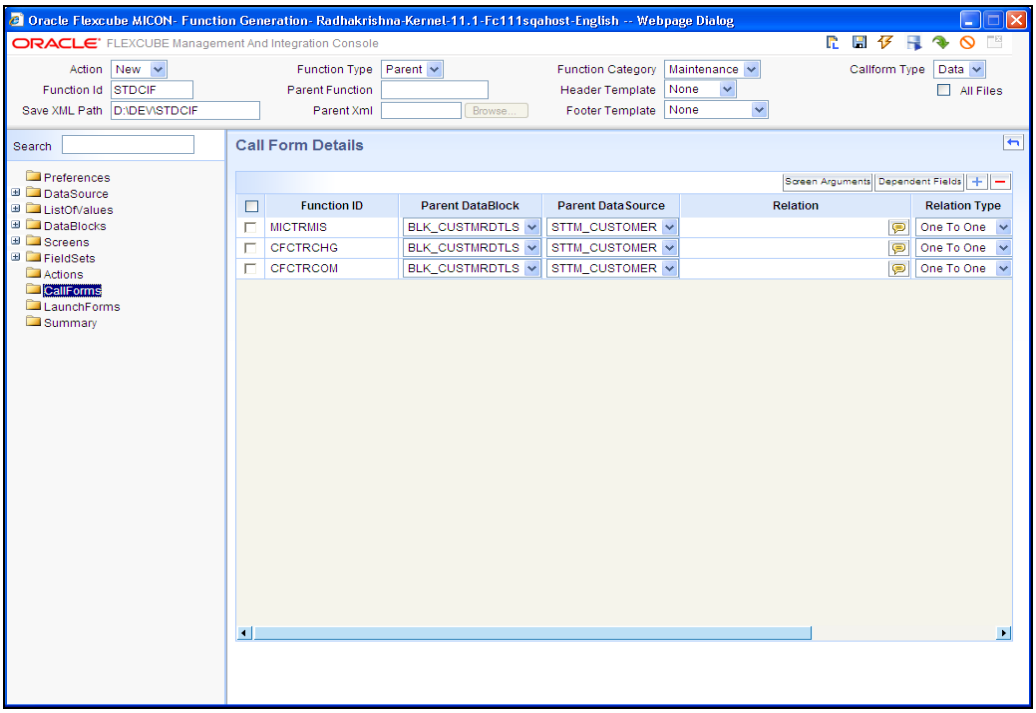
Contains all the fields of a block which is selected

4.14.7 Amendable

A field can be amendable if the check box is checked.

4.15 Call Forms

Call forms details of a function can be maintained using call forms screen .below is the screen for the same.



4.15.1 Function ID

It captures the call form function id .

4.15.2 Parent Block

Parent block can be mentioned for the call form master block.

4.15.3 Parent Data source

Parent data source can be mentioned for the call form master block.

4.15.4 Relation

There should be a relation between call form data source and main form data source and same can be maintained in the field.

4.15.5 Relation Type

Relation type can be maintained using “One To One” or “One to Many” values.

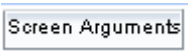
4.15.6 Display Type

A call form will be lunched from a main form by clicking a button or a tab, and same can be maintained in the field.

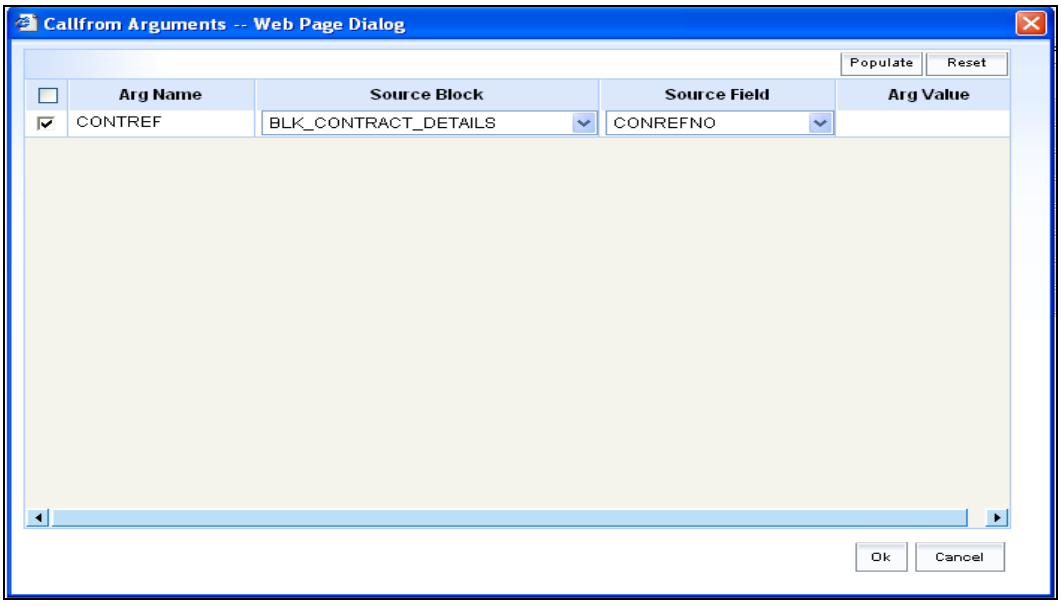
If the type is tab, at the run time on click of tab system will include the call form into the main screen as tab. The tab name under which the cal has to appear should be selected in this case.

4.16 Screen Arguments:

Whenever a Sub-Screen or Callform is launched from a main screen, there would be a necessity to send a value from calling function to the callform or sub screen, it is generally done by coding in function id.js, now Rad gives you an option to specify the screen arguments information, and thus it will reduce the code in function id js.

The button 

will launch a screen argument window as show below,



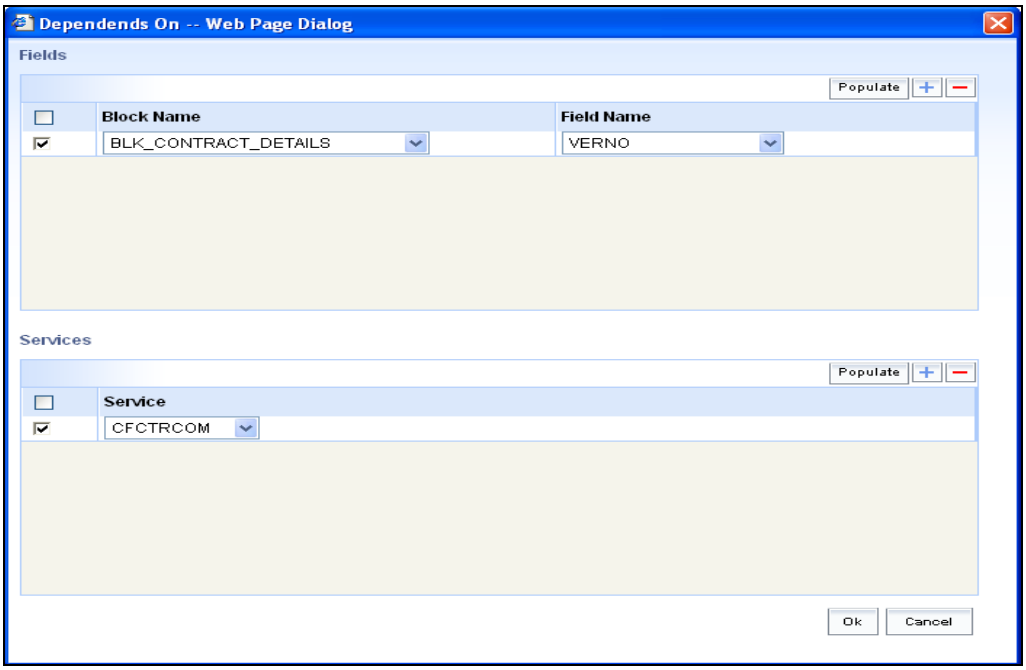
It will allow user to enter Screen Argument Name, and select Source block and field from which the value has to be taken. The user also has an option to directly give the value for the Screen Argument in the Argument value filed.

4.16.1 Dependent Fields:

After visiting some Sub-Systems, some Sub-System sensitive fields should not be allowed to change, it is generally done by coding in function id.js, now Rad gives you an option to specify the subsystem dependent information at callform level itself, and thus it will reduce the code in function id js.

The button

will launch a Dependent field's window as show below,

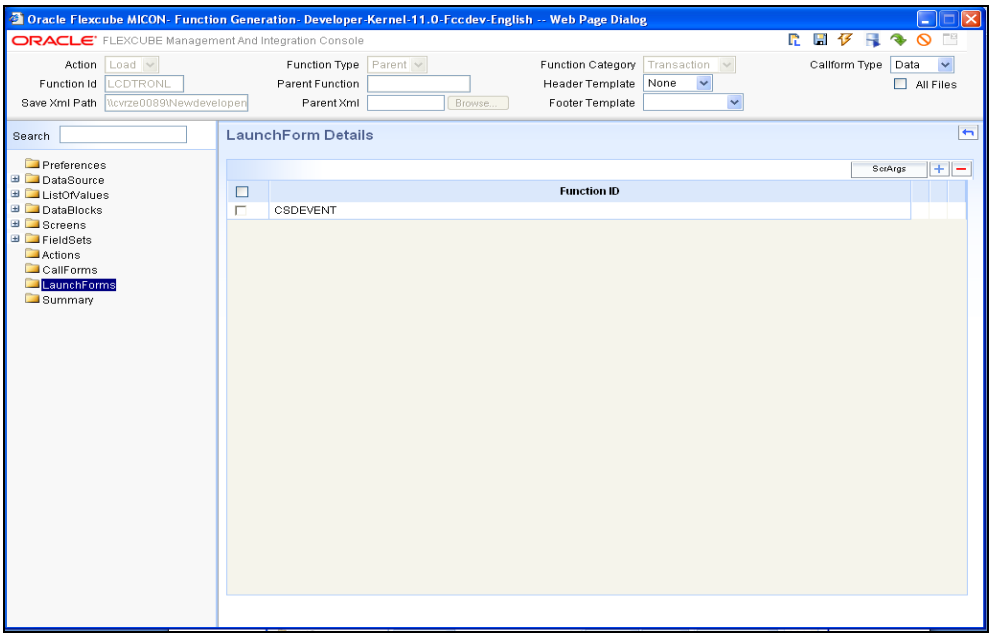


It will allow user to give Dependent fields by selecting block and field name. The user can also specify the Service name if the callform depends on any other services.

4.17 Launchforms

Launchforms are like callforms which can be launched from main screen, Launchforms third character need not be 'C'.

Launchforms details of a function can be maintained using Launchforms screen .below is the screen for the same.



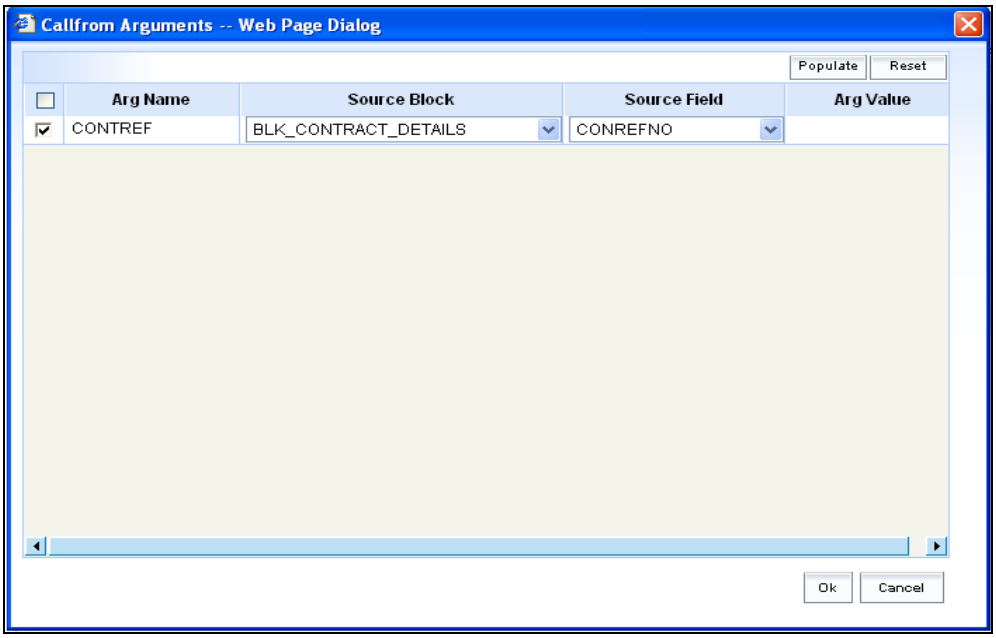
4.17.1 Function ID

It captures the call form function id .

4.17.2 Screen Arguments

Whenever a Callform or launchform is launched from a main screen, there would be a necessity to send a value from calling function to the Callform or launchform, it is generally done by coding in function id.js, now Rad gives you an option to specify the screen arguments information, and thus it will reduce the code in function id js.

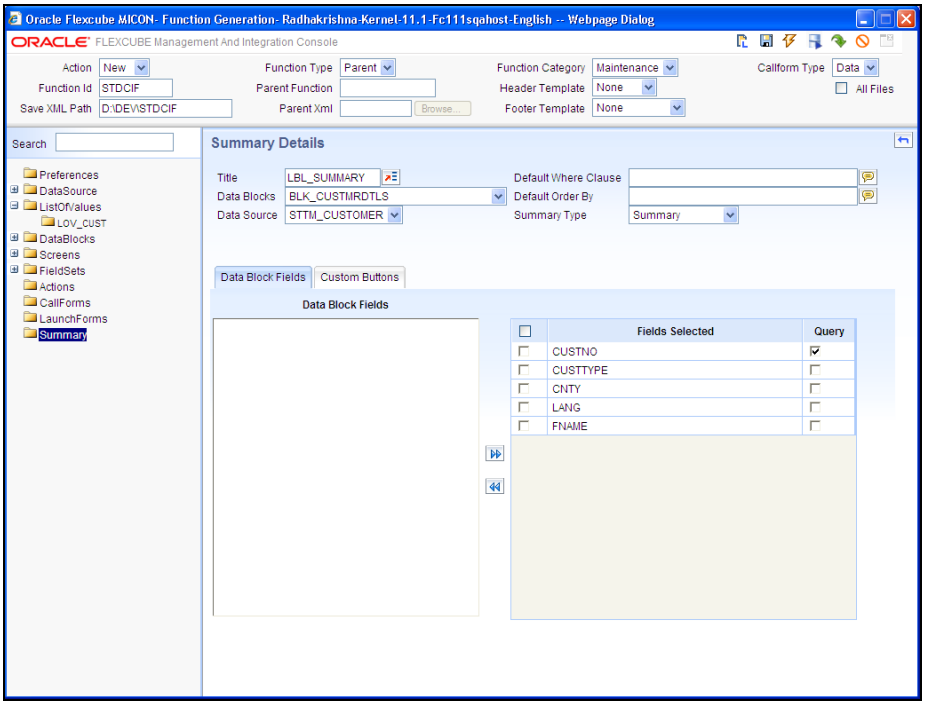
The button Screen Arguments will launch a screen argument window as show below,



It will allow user to enter Screen Argument Name, and select Source block and field from which the value has to be taken. The user also has an option to directly give the value for the Screen Argument in the Argument value filed.

4.18 Summary

Summary screen can be design in the Oracle FLEXCUBE Management and Integration Console using below screen.



4.18.1 Title

Summary screen title can be maintained here using a label code, and label code can be selected from the list of values attached to the field.

4.18.2 Data Block

This field will have all the data blocks of Multi Record “No” parameter of the Block Properties. On selecting of the data block all the data source will be populated to data source field which are attached to the selected data block.

4.18.3 Data Source

Captures the data source name of the summary, on which query should happen for summary.

4.18.4 Default where Clause

Captures the default where clause for summary query.

4.18.5 Default Order By

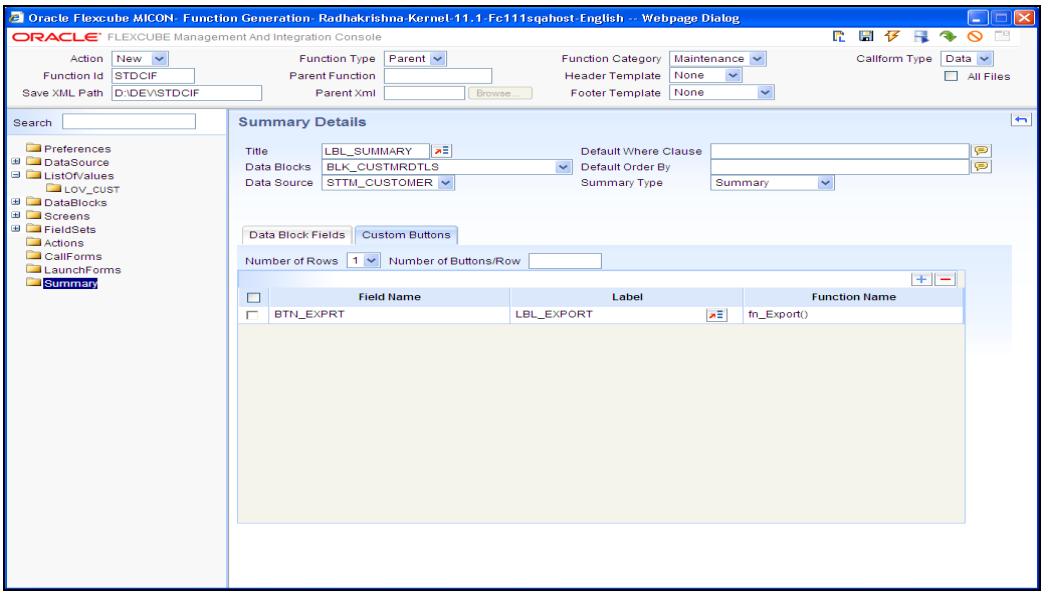
Captures the default order by for summary query.

4.18.6 Summary Type

Summary screen can be of any type Summary, Query, Bulk Authorization and upload and same can be maintained here.

4.19 Custom Buttons

Buttons can be added to summary screen using “Custom Buttons” Tab of the summary screen of Oracle FLEXCUBE Management and Integration Console, same is shown below



4.19.1 Number of Rows

Adding buttons to the summary screen can be of any number, and buttons cab arranged in more than one row same can be maintained using number of Rows field.

Max rows cab be added is 5 rows.

4.19.2 Number of Buttons/Row

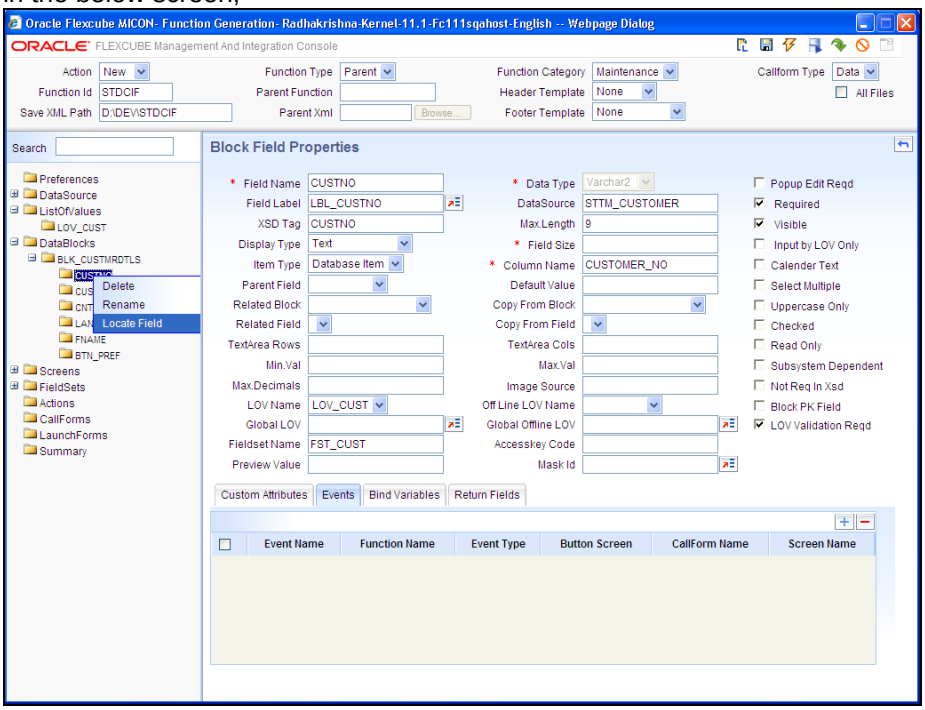
Number of Buttons per Row can be maintained.

5. Extended search option

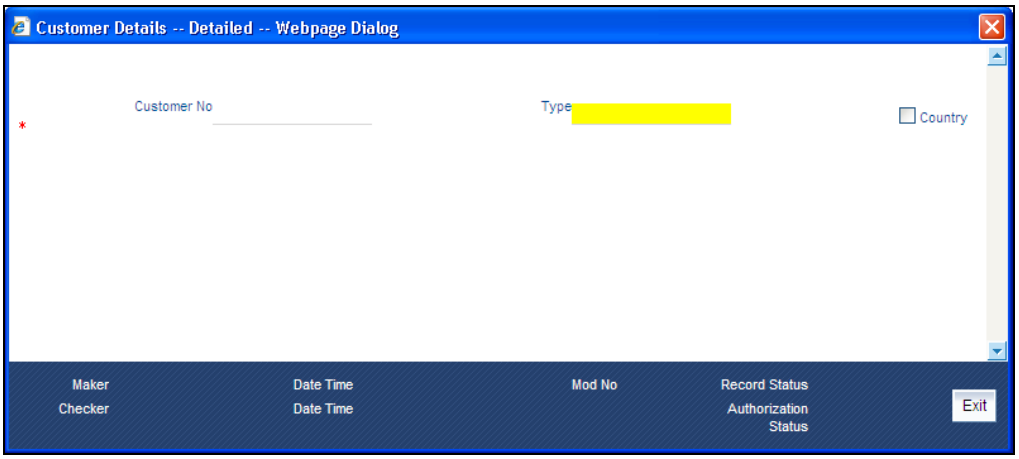
The Oracle FLEXCUBE Management and Integration Console give you extended searching option through which a block field can be searched in Function generation link.

5.1 Locate Field:

Upon right clicking on a block filed the user will get a Locate Field option in the menu as shown in the below screen,



On clicking the locate field option, FLEXCUBE MICON launches the screen which has the field in Preview mode and highlights the field and show it as shown below.



5.2 Going to Block Field from Preview

Upon double clicking on a filed in Screen in Preview mode ,FLEXCUBE MICON Closes the preview window and goes to the block field and displays its attributes.

6. Child and Parent functionality

Oracle FLEXCUBE Management and Integration Console has the capability to support Child Screens. The concept is very similar to inheritance in OOPs and a child screen can be derived from a parent screen.

Once a child screen is inherited a parent screen it gets all the sections from the parent.

Below are the possible operations on a child screen in Oracle FLEXCUBE Management and Integration Console.

- New data Sources can be added
- New data Blocks can be added
- Existing data Sources cannot be deleted
- Existing data Blocks cannot be deleted
- Screens and field sets can be completely modified

Oracle FLEXCUBE Management and Integration Console would track any modification done in child and stores separately. Whenever the parent screen gets modified, its Childs can be refreshed with the parent.

During Refresh Open Development would do the below

- Copies the block field properties from the parent child only if none of the field properties are modified in Child
- If any property of a Block field is modified, it retains the entire set of block fields from child itself.
- Copies the new data sources, fields, LOVS from the parent to child
- For screens and Field sets, Refresh does not alter the child screen and field sets .i.e any modifications done to the parent screen after the child is inherited would not get propagated to the child.

7. Open Development- generated Files

7.1.1 Front End Units

Oracle FLEXCUBE Management and Integration Console generates the below front end units.

- <Function_id>. Xml
- <Function_id>_SYS.JS
- <Function_id>_KERNLE.JS
- <Function_id>_CLUSTER.JS
- <Function_id>_CUSTOM.JS

7.1.2 Back End units

Oracle FLEXCUBE Management and Integration Console generates the below packages for maintenance Functions

- Main package
- Kernel package
- Cluster package
- Custom Package
- Static data Scripts
- XSDs

All the package specifications and package bodies will be generated only when a new function id is created for the first time in Open Development. Subsequent to that, only Main package gets generated. In case all the package is required developers can check the “All Packages” check box in the header. As many functions are existing function ids and are being converted, when the function id is converted for the first time, developers should check this check box to get Kernel, Cluster and Custom packages.

Based on the Release type and Action Code Oracle FLEXCUBE Management and Integration Console controls the file generation as below

7.1.3 Release Based

For Kernel release, Open Development would generate all files for Child releases (Cluster and Custom) also. Similarly while working for a cluster release tool generates Custom files as well.

7.1.4 Action Based

In case of New Tool generates all the files based on the release type as explained above. IN case of Reloading an existing screen, Open Development only generates the below files

- Main Package Specification
- Main Package Body
- Static data Scripts
- XSDS

Kernel, Cluster and Custom packages would get generated only while creating a new form based on the applicability of the Release .i.e. Kernel, Cluster and custom for Kernel release, Cluster and Custom for Cluster Release and only Custom Package for Custom Release. This is because these are the user hook packages and developers would have modified these packages. Re-Generating these packages would nullify the changes made by the developers if loaded in the database.

In case there is a need to generate these packages on load action, the “All Packages” should be checked. If this flag is checked Oracle FLEXCUBE Management and Integration Console would generate all the files applicable to the release type.

8. Standard Naming Conventions

There are some standard naming conventions to be followed while creating objects in Oracle FLEXCUBE Management and Integration Console.

8.1 Screen Name

Screen names should stats with “CVS_” i.e. “CVS_<Name of the screen>”

8.2 Block Name

Block Name should start with BLK_ and should be followed by the XSD Node. For example if the XSD Node is Product-Details Block name should be BLK_PRODUCT_DETAILS.

Basically its Upper case and “-“Replaced with “_”.

8.3 Field Set Names

Field set names should stats with “FST_” i.e. “FST_<Name of the Field set>”

8.4 LOV names

LOV names should stats with “LOV_” i.e. “LOV_<Name of the LOV>”

9. Conversion procedure

Below is some briefing on how convert the screens to New Open Development

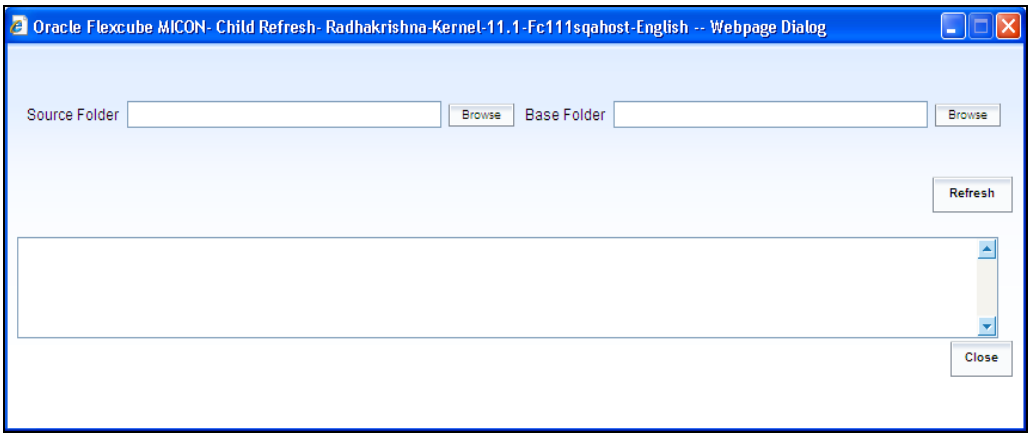
1. Launch Oracle FLEXCUBE Management and Integration Console and use Data Extraction from OLD Open Development xml.
2. This will populate the table CSTB_DATA_DICTIONARY with all the tables and columns used in the Open Development Xml
3. There is another column in the table FILED_NAME . This field should contain the XSD_TAG of the column name and this should be populated manually by the TL/PL who can give a reasonable short and intuitive name Below are some of the Examples
   - CODE CD
   - BRANCH BRN
   - DATE DT
   - CREDIR CR
   - DEBIR DR etc
4. It also populates another table RDTB_OLD_Open Development_LABELS which gives the Labels used in the OLD Open Development Xml. This data can be sued while populating the Label Descriptions
5. Now Load the OLD Open Development Xml in New Open Development. Oracle FLEXCUBE Management and Integration Console Automatically converts into the NEW Open Development Format and picks the FIELD_NAMES from data Dictionary and uses them as the BLOCK Filed Names. The same would be populated as the XSD Tag as well.
6. System would automatically default the label also which is nothing but LBL_<FIELD_NAME>. As mentioned in point 4, Descriptions can be picked up from RDTB_OLD_Open Development_LABELS.
7. After conversion, there some irregularities (Primarily because of Incorrect Block definitions in OLD Open Development). Please look for the below irregularities
   - In case a particular Data block contains more than one data source, both should either be Multi record or both should be single Record. And the Block also should also be either Multi or Single based on the Data Sources.
   - Check for incorrect Clubbing of Blocks. i.e. as the Block name is going to be the XSD Node name, we can only club functionally related data sources only. For example in product, CSTM_PRODUCT and the Module level product Master (say LCTM_PRODUCT_DEFINITION) can be defined as one Block in Oracle FLEXCUBE Management and Integration Console.
   - Check for Unnecessary UI fields.
8. Block Name Should Start with BLK_ and it should be same as XSD Node name for Example BLK_PRODUCT_DETAILS ----- Product-Details
9. Don't use UI fields as Bind Variables while defining LOVS
10. Avoid using Query data Sources. Instead Description Fields can be used for showing Descriptions

11. In Actions tab, in case web services are required, give the below
    - XSD Identifier - Short Description for the Function ID LCDPRMNT LCProduct
    - Service Name - Service Name
    - Operation ID - It is a critical parameter and which would indicate the Operation of the Function. For example LCDPRMNT is Product, LCDCONON is Contract Etc. Please note that in most cases module need not be given as there may not multiple Contracts /products within the same service. In case where multiple Products under the same service like (Deal, Portfolio etc) , please include a Prefix to the Identifier. For example for EDDDLPRD Identifier can be EDDDLPRD and Operation Id can be Deal Product. Please ensure that the Operation Id is in Sentence Case .
    - ON giving the above Parameters, System would automatically defaults the Operation Code.
12. Please ensure loading and check in of all INCS generated from Oracle FLEXCUBE Management and Integration Console.
13. Don’t use UI fields as Bind Variables while defining LOVS
14. Remove LOV Validations from the addon package. Please ensure these are present in Main Package before removing the validations from Addon package.

10. Refresh

The Oracle FLEXCUBE Management and Integration Console allows upgrading the base version by Refreshing the base version, the following types of Refresh are possible in Oracle FLEXCUBE Management and Integration Console,

- Source Refresh
- Child Refresh



10.1 Source Refresh

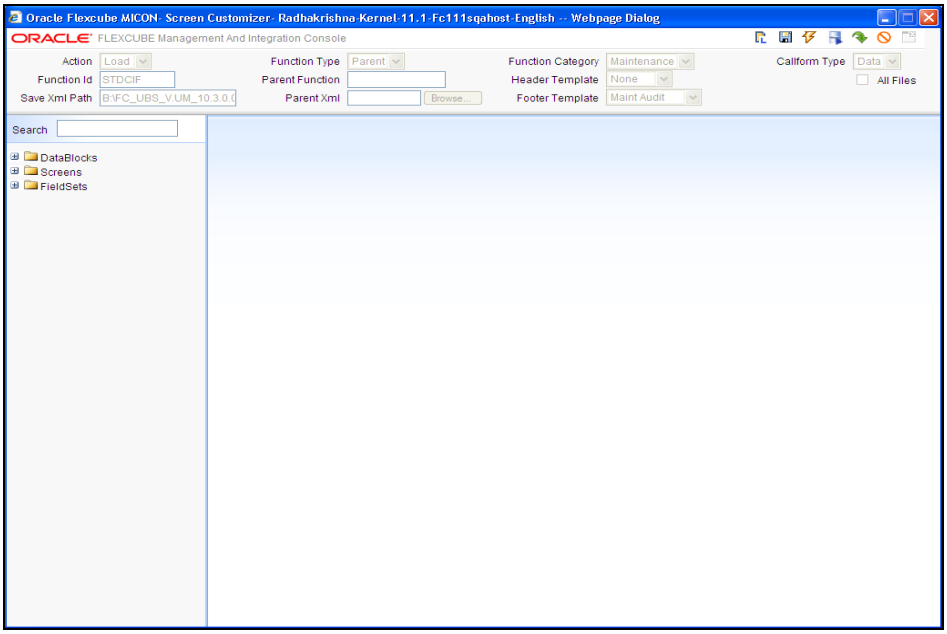
The Source refresh is done based on the release; from Cluster release it allows you to refresh the Kernel and from Custom release it allows to refresh both Kernel and Cluster. The Source refresh can be done by clicking on the Source refresh link in the Oracle FLEXCUBE Management and Integration Console, which will launch a new screen where you can specify the base and source path and click on the Refresh, which will indeed Refresh all the Open Development xmls in the source folder.

10.2 Child Refresh

The Child Refresh allows upgrading of all release respective changes; it refreshes all, Kernel, Cluster and Custom and upgrades all the releases of a Child function Id. The child refresh can be done by clicking on the child refresh link in the Oracle FLEXCUBE Management and Integration Console, which will launch the below screen where you can specify the base and source path and click on the Refresh, which will indeed Refresh all the Open Development xmls in the source folder.

11. Screen customizer

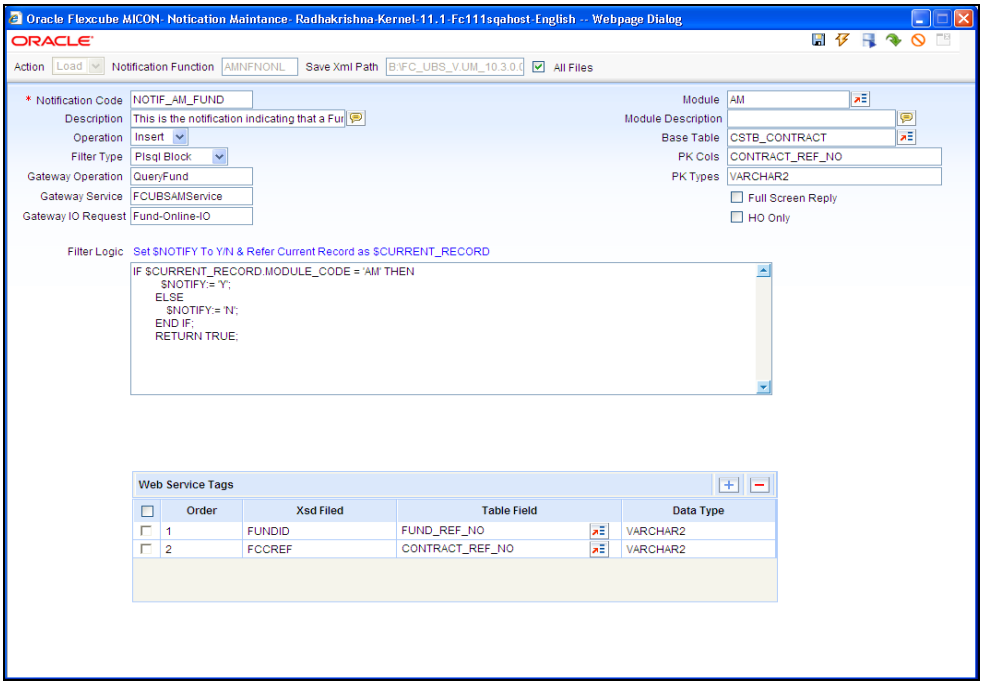
The Oracle FLEXCUBE Management and Integration Console allows certain users only to do UI changes, In that case the function generation screen will show only Datablocks, Screens and Fieldset nodes in the tree. And Screen would look as shown below,



The user will be allowed only to modify the layout of the screen.

12. Notification

The Oracle FLEXCUBE Management and Integration Console Notification can be used to create the notification Open Development xml and the Notification screen is shown below



12.1 Action

It specifies the Action, New or Load.

12.2 Notification Function

Notification function is unique function id with 8 characters and 3rd character would be ‘N’.

12.3 Notification Code

It stores the notification code.

12.4 Description

It stores the notification Description.

12.5 Operation

It stores the notification operation, it can be Insert or Delete or both.

12.6 Filter type

It stores the notification filter type, it can be plsql block or where clause.

12.7 Gateway Operation

It is used to specify the Gateway operation.

12.8 Gateway Service

It is used to specify the Gateway Service.

12.9 Gateway I/O Request

It is used to specify the Gateway I/O Request.

12.10 Notification Module

It is used to specify the Notification Module.

12.11 Base Table

It is used to specify the Base Table.

12.12 PK Cols

It is used to specify the Base Table PK Cols.

12.13 PK Types

It is used to specify the PK Cols's type.

12.14 Full Screen Reply

It is used to specify whether it Full screen reply or not.

12.15 HO Function

It is used to specify whether its Head Office function or not

12.16 Filter Logic

It is used to specify the Filter Logic.

12.17 Field Order

It is used to specify the field Order.

12.18 XSD Field

It is used to specify the XSD Field.

12.19 Table Field

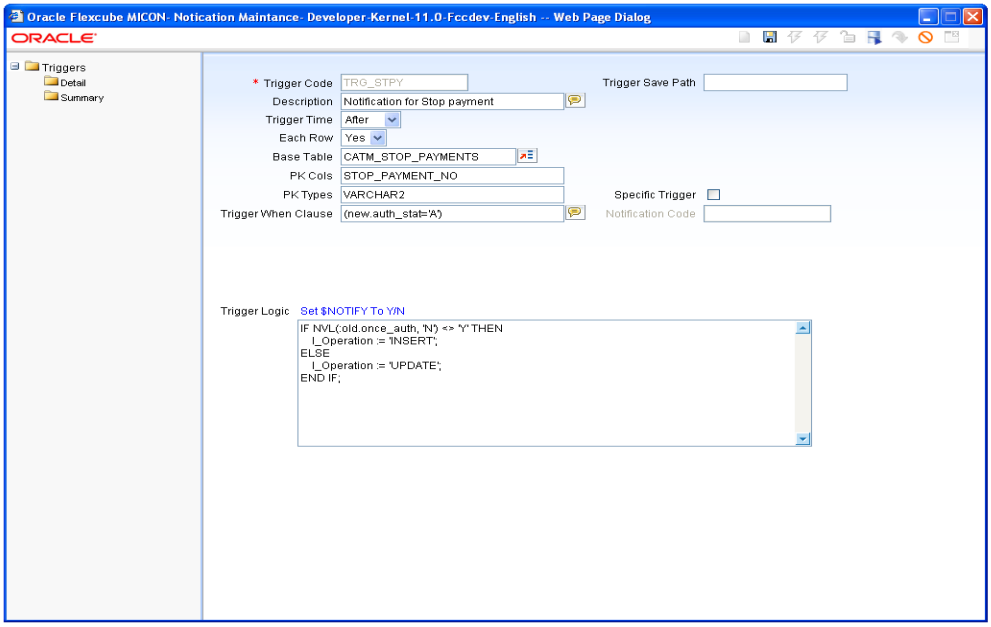
It is used to specify the Table field.

12.20 Data Type

It is used to specify the Data Type.

13. Trigger Maintenance

The Oracle FLEXCUBE Management and Integration Console Trigger Maintenance is used to Maintain Triggers for Notification, Trigger Maintenance Detail screen is shown below



13.1 Trigger Code

It Captures the trigger code.

13.2 Description

It captures the trigger Description.

13.3 Trigger Time

It captures when to fire the trigger, Before or after.

13.4 Each Row

It fires the trigger Each Row, if it is Yes.

13.5 Base Table

It is used to specify the Base Table.

13.6 PK Cols

It is used to specify the Base Table PK Cols.

13.7 PK Types

It is used to specify the PK Cols's type.

13.8 Trigger When clause

It Captures the Trigger when clause.

13.9 Specific Triggers

It is used to specify if it is a specific Trigger.

13.10 Notification Code

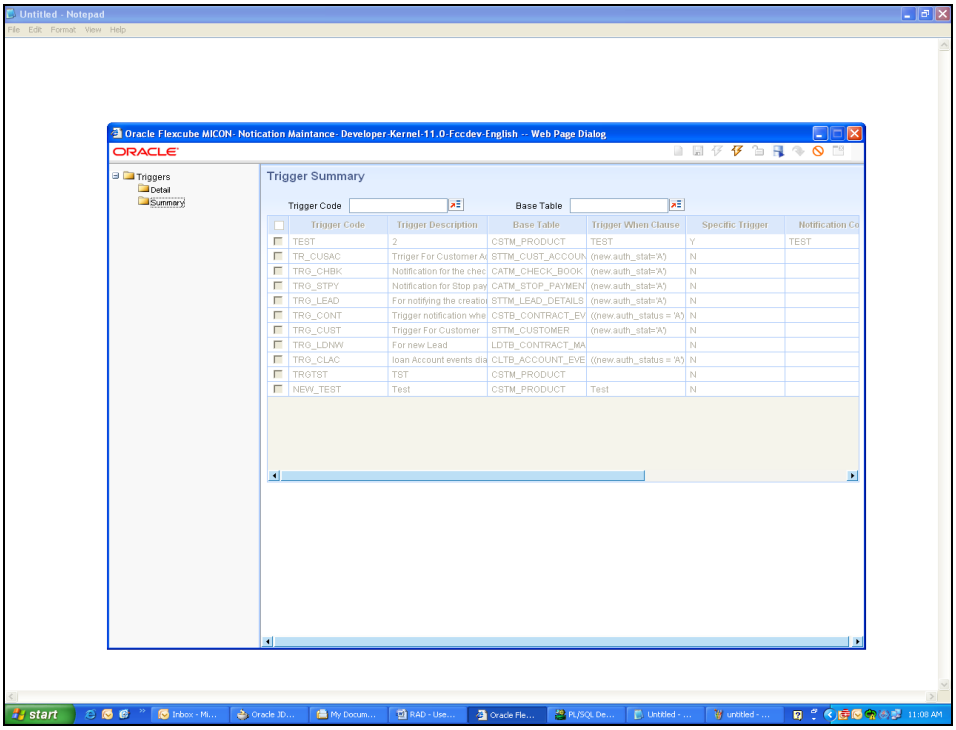
In case of Specific Triggers, it captures the Notification codes.

13.11 Trigger Logic

It is used to specify the Trigger Logic.

13.12 Summary

It is used to view the list of triggers available in GWTM_NOTIFICATION_TRIGGERS table. on Double-click of a row it launches the Detail screen.



14. Bulk Generation

The Oracle FLEXCUBE Management and Integration Console allows Bulk Generation of both front end and backend units for both New and Old Open Development Xmls, the Bulk Generation Screen is shown below

The user has to specify the Source folder where the Open DevelopmentXMLs are placed, and give the destination path where user wants the files to be generated and select the necessary files, if All files are required, option 'All' has to be selected and generate button has to be clicked. It will generate the files in the destination path and give log information in the log window which is present at the bottom of the screen.

14.1 Source Folder

Structure of the folders must be same as vercon area folder structure.

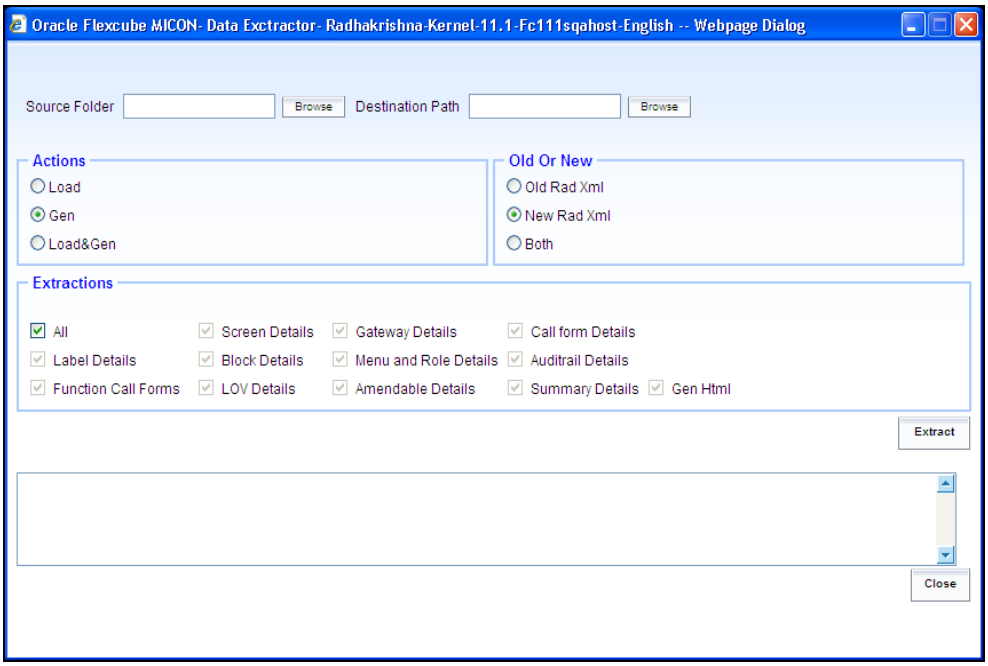
14.2 Destination Folder

For non-extensible Radxml's call forms should be copied at the Destination <Destination Path>/CALLFORM_PATH.

If folder not available Bulk generator tool will create CALLFORM_PATH and copy the all Open DevelopmentXMLs from source folder to newly created folder or overwrite existing files.

15. Data Extractor

The Oracle FLEXCUBE Management and Integration Console allows the data Extraction for both old and new Open DevelopmentXMLs, the Data Extractor link launches the below Screen.



The user has to specify the Source folder where the Open Development xmls are placed, and give the destination path where he wants the inc files to be generated and select the necessary options for which he want the Extraction, if All options are required, option 'All' has to be selected and Extract button has to be clicked. It will generate the INC files in the destination path and give the log information in the log window which is present at the bottom of the screen.

15.1 Actions

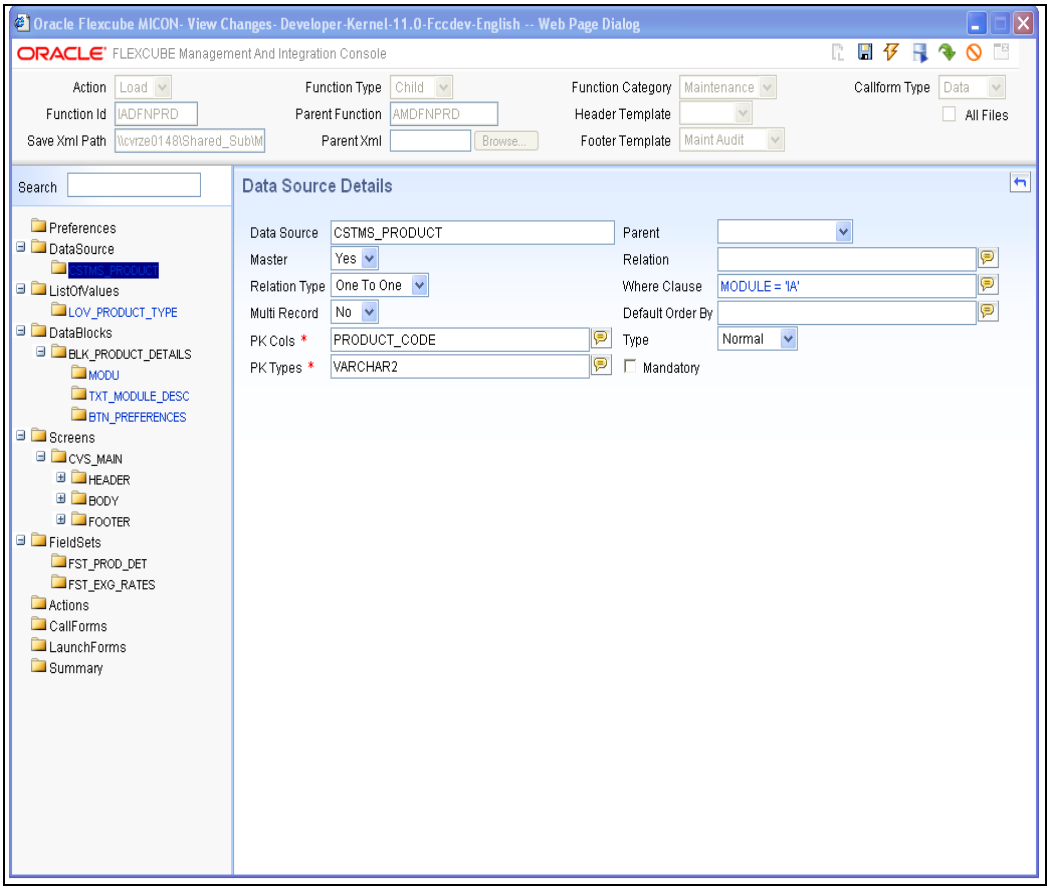
It is used to specify whether the inc has to be generated or loaded or both.

15.2 Old or New

It is used to specify whether the inc has to be generated only for New Rad xml or Old Rad xml or for both.

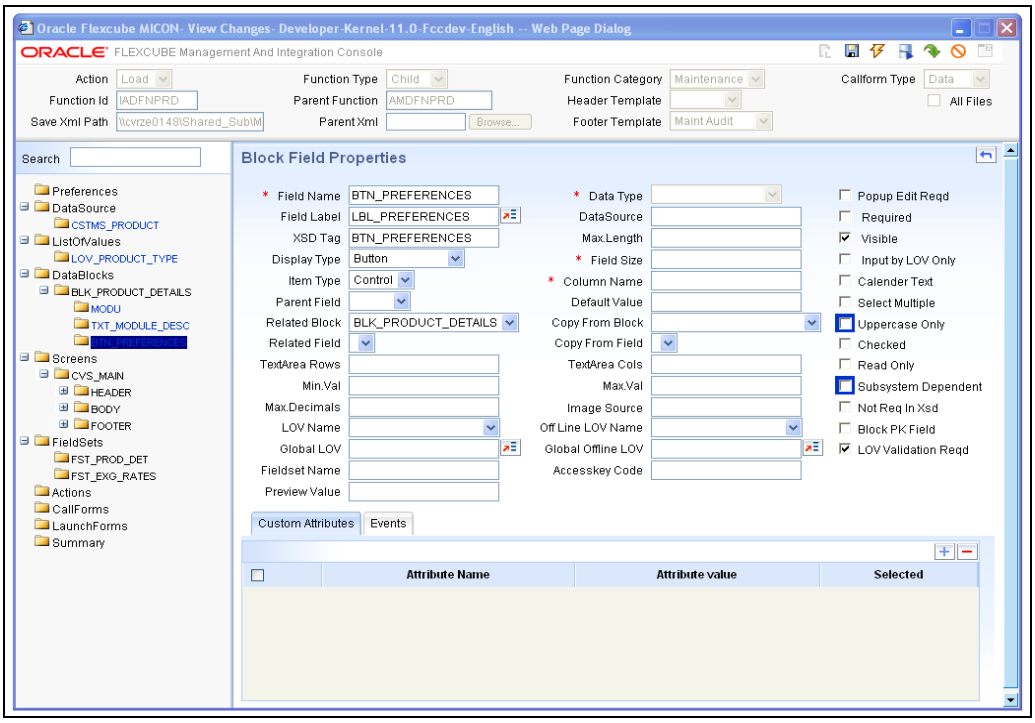
16. View Changes

The Oracle FLEXCUBE Management and Integration Console allows to view the changes done as part of Child or Cluster or Custom Release, upon loading the Cluster or Custom or Child Open Development xml, it shows the Modifications done as part of Cluster or Custom or Child. The View changes Screen is shown below,

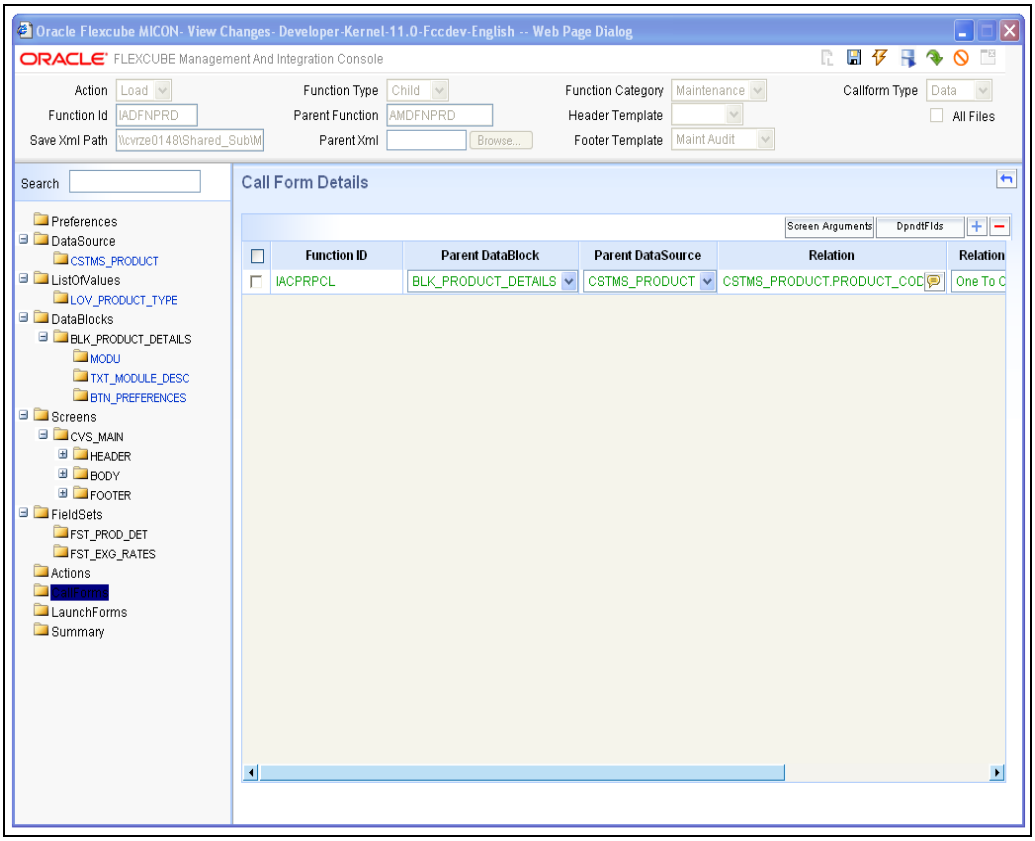


It shows the modified Data Sources, Data Source Fields, LOVs, Data Blocks, Block Fields,

Call forms etc in Blue Color as shown below,

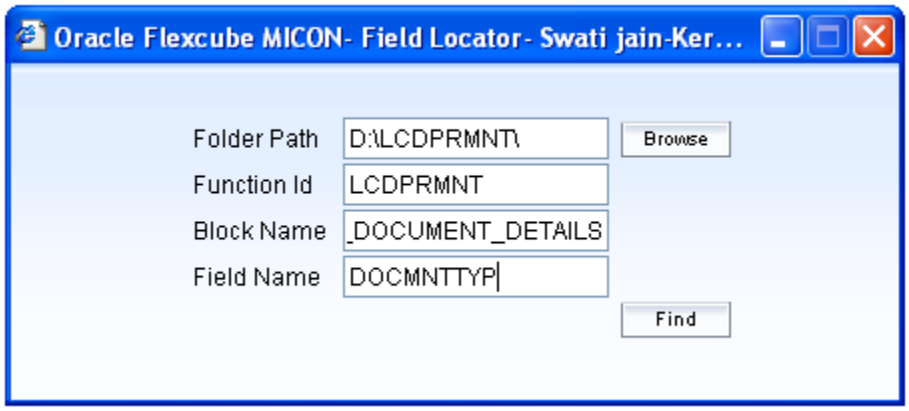


And it shows the newly added Data Sources, Data Source Fields, LOVs, Data Blocks, Block Fields, Call forms etc in Green Color as shown below,

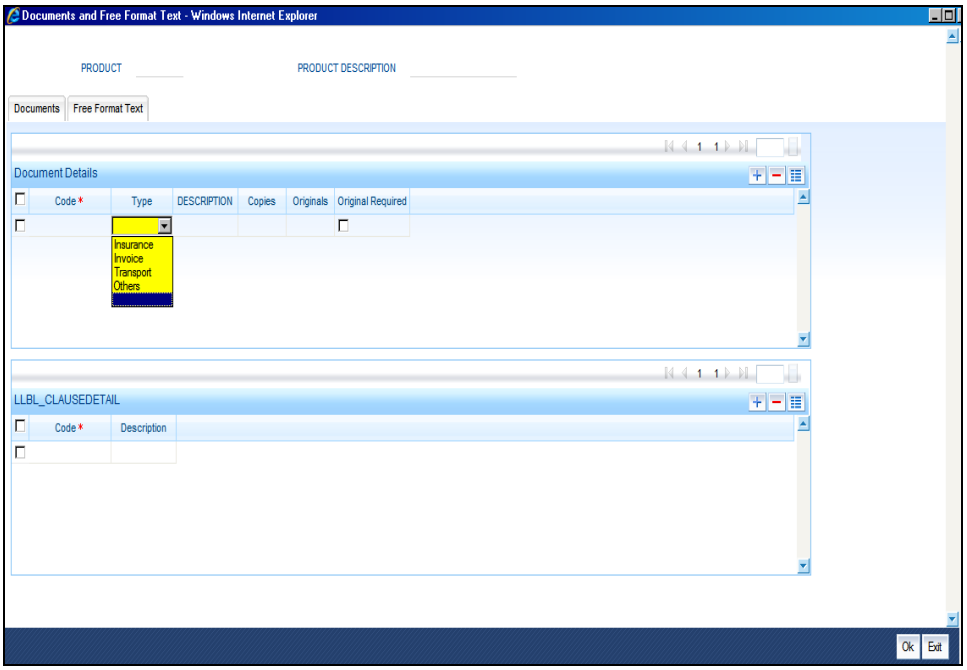


17. Field Locator

The Oracle FLEXCUBE Management and Integration Console Field Locator utility will help the user to identify a field in function id. In case of function ids with big or more screens it is always been a challenge for the user to find out the filed in the screen, with Field locator it is fairly simple, user can specify the folder path where the function id HTMLs are present, function id , block name and field name and then click on Find button the FLEXCUBE MICON opens the HTML which has the particular field and highlights the field.



The below screen shows how the field locator shows the field.



18. Design practices

- Do not use Query data Sources for Description Fields
- Do not use UI fields if a database item can be used.
- Do not use UI items as Bind variables. As we can use Bind variables from other Blocks, it's not required.
- Do not write Code in JS for sending values to sub screen or call forms.
- Do not write code in JS for disabling sub system dependent fields

Reference Guide
[May] [2018]
Version 14.0.0.0.0

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], [2018], 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.