Development of Maintenance Form Oracle FLEXCUBE Universal Banking Release 14.4.0.2.0 Part No. F36581-01 [November] [2020]



Contents

1.1 Audience 3 1.2 Related Documents 3 2.1 Introduction 4 2.1 How to use this Guide 4 3. Overview of Maintenance Screen 4 4. Screen Development 4 4.1 Header Information 4 4.2 Preferences 6 4.3 Data Sources 7 4.4 Data Sources 7 4.4 Data Blocks 12 4.5 Screens 14 4.6 Field Sets 14 4.6 Field Sets 12 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5 Generation and Deployment of files 30 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.2	1. Prefa	ıce	3
2. Introduction 4 2.1 How to use this Guide 4 3. Overview of Maintenance Screen. 4 4. Screen Development. 4 4.1 Header Information 4 4.2 Preferences. 6 4.3 Data Sources 7 4.4 Data Blocks 12 4.5 Screens 14 4.6 Field Sets 16 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5. Generation and Deployment of files 33 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 34 6.3 Other Units 34 <td>1.1</td> <td>Audience</td> <td>3</td>	1.1	Audience	3
2.1 How to use this Guide 4 3. Overview of Maintenance Screen 4 4. Screen Development. 4 4.1 Header Information 4 4.2 Preferences. 6 4.3 Data Sources 7 4.4 Data Sources 7 4.4 Data Sources 12 4.5 Screens 14 4.6 Field Sets 16 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5 Generation and Deployment of files 30 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 33 6.3 Other Units 34 <td>1.2</td> <td>Related Documents</td> <td>3</td>	1.2	Related Documents	3
3. Overview of Maintenance Screen 4 4. Screen Development 4 4.1 Header Information 4 4.2 Preferences 6 4.3 Data Sources 7 4.4 Data Blocks 12 4.5 Screens 14 4.6 Field Sets 16 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5. Generation and Deployment of files 30 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.2 Data Base Units 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.3 Other Units 34 6.3 Other Units 34 6.3 Other Units 34 6.3 Other Units 34 7.1 Extensibility in JavaScript Coding 34 7.2 Extensibility in JavaScript Coding 34 7.2 Extensibility in Backend Coding 35 7.2.1 Functions in Hook Packages 35 <t< td=""><td></td><td></td><td></td></t<>			
4. Screen Development. 4 4.1 Header Information 4 4.2 Preferences. 6 4.3 Data Sources 7 4.4 Data Blocks 12 4.5 Screens. 14 4.6 Field Sets. 16 4.7 LOV 20 4.8 Attaching Call forms. 23 4.9 Adding Summary. 27 4.10 Amendable fields Maintenance. 29 5. Generation and Deployment of files. 30 6.1 Front End Units. 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2 Data Base Units. 33 6.2.1 Static Scripts. 33 6.2.2 System Packages. 34 6.3 Other Units. 34 6.3 Other Units. 34 6.3 Other Units. 34 7.1 Extensibility in JavaScript Coding. 34 7.2 Extensibility in JavaScript Coding. 35 7.2.1 Functions in Hook Packages 35 7.2.2 Flow of control through Hook packages 35			
4.1 Header Information 4 4.2 Preferences 6 4.3 Data Sources 7 4.4 Data Blocks 12 4.5 Screens 14 4.6 Field Sets 16 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5. Generated Units 33 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2 Data Base Units 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 34 6.3 Other Units 34 6.3.1 Xsd 34 7.1 Extensibility in JavaScript Coding 34 7.2 Flow of control through Hook packages 35			
4.2 Preferences	4. Scree	-	
4.3 Data Sources 7 4.4 Data Blocks 12 4.5 Screens 14 4.6 Field Sets 16 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5 Generation and Deployment of files 30 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2 Data Base Units 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 33 6.2.4 Static Scripts 34 6.3 Other Units 34 6.3 Other Units 34 6.3.1 Xsd 34 7.1 Extensible Development 34 7.2 Extensibility in JavaScript Coding 35		Header Information	4
4.4 Data Blocks 12 4.5 Screens 14 4.6 Field Sets 16 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5. Generation and Deployment of files 33 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2 Data Base Units 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 34 6.3 Other Units 34 6.3.1 Xsd 34 7.1 Extensibility in JavaScript Coding 34 7.2 Flow of control through Hook packages 35 7.2.2 Flow of control through Hook packages 35			
4.5 Screens 14 4.6 Field Sets 16 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5. Generation and Deployment of files 30 6. Generated Units 33 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2 Data Base Units 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 34 6.3.1 Xsd 34 7.1 Extensibile Development 34 7.2 Extensibility in JavaScript Coding 35 7.2.1 Functions in Hook Packages 35 7.2.2 Flow of control through Hook packages 35			
4.6 Field Sets 16 4.7 LOV 20 4.8 Attaching Call forms 23 4.9 Adding Summary 27 4.10 Amendable fields Maintenance 29 5. Generation and Deployment of files 30 6. Generated Units 33 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2 Data Base Units 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 34 6.3 Other Units 34 6.3 Other Units 34 6.3 Cher Units 34 7.1 Extensibility in JavaScript Coding 35 7.2.2 Flow of control through Hook packages 35			
4.7LOV204.8Attaching Call forms234.9Adding Summary274.10Amendable fields Maintenance295.Generation and Deployment of files306.Generated Units336.1Front End Units336.1.1Language xml336.1.2SYS JavaScript File336.1.3Release Type Specific JavaScript File336.2Data Base Units336.2.1Static Scripts336.2.2System Packages336.3Other Units346.3Other Units347.1Extensible Development347.2Extensibility in JavaScript Coding347.2Flow of control through Hook packages357.2.2Flow of control through Hook packages35			
4.8Attaching Call forms234.9Adding Summary274.10Amendable fields Maintenance295.Generation and Deployment of files306.Generated Units336.1Front End Units336.1.1Language xml336.1.2SYS JavaScript File336.1.3Release Type Specific JavaScript File336.2Data Base Units336.2.1Static Scripts336.2.2System Packages336.2.3Hook Packages346.3Other Units346.3Other Units347.Extensible Development347.1Extensiblity in JavaScript Coding347.2Extensibility in Backend Coding357.2.2Flow of control through Hook packages35			
4.9Adding Summary274.10Amendable fields Maintenance295.Generation and Deployment of files306.Generated Units336.1Front End Units336.1.1Language xml336.1.2SYS JavaScript File336.1.3Release Type Specific JavaScript File336.2Data Base Units336.2.1Static Scripts336.2.2System Packages336.2.3Hook Packages346.3Other Units347.Extensible Development347.1Extensiblity in JavaScript Coding347.2Extensibility in Backend Coding357.2.1Flow of control through Hook packages35			
4.10Amendable fields Maintenance295.Generation and Deployment of files306.Generated Units336.1Front End Units336.1.1Language xml336.1.2SYS JavaScript File336.1.3Release Type Specific JavaScript File336.2Data Base Units336.2.1Static Scripts336.2.2System Packages336.2.3Hook Packages346.3Other Units346.3.1Xsd347.Extensible Development347.1Extensibility in JavaScript Coding357.2.1Functions in Hook Packages357.2.2Flow of control through Hook packages35			
5. Generation and Deployment of files 30 6. Generated Units 33 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2 Data Base Units 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 34 6.3 Other Units 34 6.3.1 Xsd 34 7. Extensible Development 34 7.1 Extensibility in JavaScript Coding 35 7.2.1 Functions in Hook Packages 35 7.2.2 Flow of control through Hook packages 35			
6. Generated Units 33 6.1 Front End Units 33 6.1 Front End Units 33 6.1.1 Language xml 33 6.1.2 SYS JavaScript File 33 6.1.3 Release Type Specific JavaScript File 33 6.2 Data Base Units 33 6.2.1 Static Scripts 33 6.2.2 System Packages 33 6.2.3 Hook Packages 34 6.3 Other Units 34 6.3.1 Xsd 34 7. Extensible Development 34 7.1 Extensibility in JavaScript Coding 34 7.2 Extensibility in Backend Coding 35 7.2.1 Functions in Hook Packages 35 7.2.2 Flow of control through Hook packages 35			
6.1Front End Units336.1.1Language xml336.1.2SYS JavaScript File336.1.3Release Type Specific JavaScript File336.2Data Base Units336.2.1Static Scripts336.2.2System Packages336.2.3Hook Packages346.3Other Units346.3Other Units347.Extensible Development347.1Extensibility in JavaScript Coding347.2Extensibility in Backend Coding357.2.1Functions in Hook Packages357.2.2Flow of control through Hook packages35			
6.1.1Language xml336.1.2SYS JavaScript File336.1.3Release Type Specific JavaScript File336.2Data Base Units336.2.1Static Scripts336.2.2System Packages336.2.3Hook Packages346.3Other Units346.3.1Xsd347.Extensible Development347.1Extensibility in JavaScript Coding347.2Extensibility in Backend Coding357.2.1Functions in Hook Packages357.2.2Flow of control through Hook packages35			
6.1.2SYS JavaScript File	0.12		
6.1.3Release Type Specific JavaScript File336.2Data Base Units336.2.1Static Scripts336.2.2System Packages336.2.3Hook Packages346.3Other Units346.3.1Xsd347.Extensible Development347.1Extensibility in JavaScript Coding347.2Extensibility in Backend Coding357.2.1Functions in Hook Packages35	6.1.1		
6.2Data Base Units.336.2.1Static Scripts336.2.2System Packages.336.2.3Hook Packages.346.3Other Units.346.3.1Xsd.347.Extensible Development.347.1Extensibile Development.347.2Extensibility in JavaScript Coding.347.3Functions in Hook Packages357.2.1Flow of control through Hook packages35	6.1.2	SYS JavaScript File	
6.2.1Static Scripts336.2.2System Packages336.2.3Hook Packages346.3Other Units346.3.1Xsd347.Extensible Development347.1Extensibility in JavaScript Coding347.2Extensibility in Backend Coding357.2.1Functions in Hook Packages357.2.2Flow of control through Hook packages35	6.1.3	Release Type Specific JavaScript File	
6.2.2System Packages336.2.3Hook Packages346.3Other Units346.3.1Xsd347.Extensible Development347.1Extensibility in JavaScript Coding347.2Extensibility in Backend Coding357.2.1Functions in Hook Packages357.2.2Flow of control through Hook packages35	6.2	Data Base Units	
6.2.2System Packages336.2.3Hook Packages346.3Other Units346.3.1Xsd347.Extensible Development347.1Extensibility in JavaScript Coding347.2Extensibility in Backend Coding357.2.1Functions in Hook Packages357.2.2Flow of control through Hook packages35	6.2.1	Static Scripts	
6.2.3Hook Packages	6.2.2	-	
6.3Other Units346.3.1Xsd347.Extensible Development347.1Extensibility in JavaScript Coding347.2Extensibility in Backend Coding357.2.1Functions in Hook Packages357.2.2Flow of control through Hook packages35	6.2.3		
6.3.1Xsd		C C	
7. Extensible Development			
7.1Extensibility in JavaScript Coding			
7.2Extensibility in Backend Coding	7. Exter		
7.2.1 Functions in Hook Packages			
7.2.2 Flow of control through Hook packages			
	7.2.1	Functions in Hook Packages	35
	7.2.2	Flow of control through Hook packages	35
	7.2.3		

1. Preface

This document describes Maintenance Screens in FLEXCUBE and the process of designing a simple Maintenance form using Oracle FLEXCUBE Development Workbench for Universal Banking

1.1 Audience

This document is intended for FLEXCUBE Application developers/users that use development Workbench to develop various FLEXCUBE components.

Proficiency	Resources
FLEXCUBE Functional Architecture	Training programs from Oracle
	Financial Software Services.
FLEXCUBE Technical Architecture	Training programs from Oracle
	Financial Software Services.
FLEXCUBE Screen Development	04-Development_WorkBench _Screen_Development-I.docx
Working knowledge of Web based applications	Self Acquired
Working knowledge of Oracle Database	Oracle Documentations
Working knowledge of PLSQL & SQL Language	Self Acquired
Working knowledge of XML files	Self Acquired

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

1.2 Related Documents

<u>04-Development_WorkBench_Screen_Development-I.docx</u> <u>05-Development_WorkBench_Screen_Development-II.docx</u>

2. Introduction

2.1 How to use this Guide

The information in this document includes:

- <u>Chapter 2 , "Introduction"</u>
- Chapter 3, "Overview of Call Form"
- Chapter 4 , "Screen Development"
- Chapter 5 , "Generated Units"
- <u>Chapter 5 , "Extensible Development"</u>

3. Overview of Maintenance Screen

Maintenance Function Id's are used for storing maintenance data which are required for processing of any contracts, batches or for any other maintenance which are dependent on this

Example: Customer maintenance screen

If any customer wants to use the service of a bank, details about the customer will have to be maintained in the system .This will be maintenance data which will be required for other maintenances (creating account for the customer) as well as for transaction processing (debiting of customer account)

Business logic for a maintenance function id would be provided by the Development Workbench generated files .Most of the cases, system provided logic would be sufficient .Extra validations can be coded in the hook packages by the developer.

4. Screen Development

Design and development of a Maintenance function id is similar to any other function Ids. This section briefs the steps in designing a Maintenance screen. STDCINF is sample function id used for demonstration in this document

For detailed explanation, refer the document: <u>4-Development_WorkBench</u>

<u>_Screen_Development-I.docx</u>

4.1 Header Information

Provide the header information as shown in the figure.

ORACLE FLEXCUBE Development Work	bench for Universal Banking	DEMOUSER
Browser +		Windows Options Sign Out
unction Generation		
		日 凶 国 伊 朝
Action None -	Function Type Parent *	Function Category Maintenance *
Function Id	Parent Function	Header Template None -
Save XML Path	Parent Xml	Footer Template None -
ListOfvalues DataBlocks Screens FieldSets Actions CallForms LaunchForms Summary		

- For new screen select action As New.
- Enter Function ID \rightarrow STDCIFD
- Function Type \rightarrow Parent
- Function Category \rightarrow Maintenance
- Parent Function Id \rightarrow None
- Parent Xml \rightarrow None
- Header Template \rightarrow None (Only for Process flow screens)
- Footer Template \rightarrow Maint Audit

	ch for Universal Banking	DEMOUSER
Browser .		Windows Options Sign Out
inction Generation		
		🖬 🗶 🕅 🎸 😫 <
Action New -	Function Type Parent -	Function Category Maintenance Save (CT8L + 5)
Function Id STDCIFD	Parent Function	Header Template None -
Save XML Path D1RADTOOL	Parent Xml	Foster Template None •
arch		
ListOffalues DataBlocks Screies Screies Adfons Califorms LaunchForms Summary		

User can save work at any point in time. Click the save icon on top right for the same .In order to work again with it select action as Load and load radxml from the hard disk path

Browser -	nent Workbench for Universa	I Banking	Windows	DEMOUSER Options Sign Out
Browser - Windows Option	s e 7 g 4			
Preferancias DalaSource ListOfvalues DalaBilocks Screens FieldSets Actions CaliForms LaunchForms	E	ror Description Error Code Of% of DownLoadFile from 10.184.132.100 Completed		

Fig 12.3: Saved File Information page

Note the following while providing header information for Maintenance screen

i) Naming Convention:

The third letter of the function id has to be D. Ideally the function id name should have 8 characters.

ii) Footer Template

Make sure that the master data source has the audit columns if footer template is provided as Maint log.

Refer <u>04-Development_WorkBench_Screen_Development-I.docx</u> for detailed explanation

4.2 Preferences

- Details entered in Preferences are used in generating INCS for SMTB_MENU, SMTB_FUNCTION_DESCRIPTION and SMTB_ROLE_DETAILS.
- **Control String** → Developer needs to select the actions which should be available for this screen in FLEXCUBE.

Function Generation Function Type Farent Function Type Farent Action New Function Type Farent Function Category Maintenance Function Id STDCIED Parent Function Header Template Save XML Path DIRADTOOL Parent Xmt Search Preferences Image: Head Office Function Image: Preferences Image: Head Office Function Image: Preferences Image: Head Office Function Image: DataSource Module Description		7 g
Action New Function Type Parent Function Type Parent Function Id STDCIFD Parent Function Save XML Path D.RADTOOL Parent Xml Parent Xml Preferences Preferences Preferences Parent Metad Office Function Module ST Parent Static Maintenance		
Function Id STDCIFD Parent Function Header Template None Save XML Path D:RADTOOL Parent Xml Footer Template None earch Preferences V Head Office Function Module ST PE DataSource Implement Addition Module Description Static Maintenance	•	
Save XML Path DVRADTOOL Parent Xml Footer Template None arch Preferences Preferences Preferences Preferences DataSource Preferences DataSource Preferences Prefe	•	
Preferences Preferences Preferences Preferences DataSource Up Logging Regulted Module ST PE		
Preferences IV Head Office Function Module ST I		
DataSource In Localno Required Module Decisional Module Concentration Static Maintenance		9
DataSource IDI Lopping Required Module Description Static Maintenance		
DataBlocks Zahovaues Josephilic State DataBlocks Zahovaues Josephilic State DataBlocks Zahovaues Josephilic State DataBlocks Zahovaues Josephilic State Zahovaues Zahovaues Josephilic State Zahovaues Zahovaues Zahovaues Josephilic State Zahovaues Zahovau		
Screens V Tank Modifications Process Code		
FieldSets VN Repository URL		
Actions Transaction Block Choose Block Choose Block Choose Block Choose Block		
LsunchForms Excel Export Required Transaction Field Choose Field •		
Summary Name		
C C C C C C C C C C C C C C C C C C C	ontrol String	+-
Function Id Module * Module Description		*
STDCIFD ST 23 Static Maintenance		_
		-

Note the following points while providing details in Preferences screen

i) Control String

REVERSE, ROLLOVER, CONFIRM, LIQUIDATE, HOLD operations are not applicable for maintenance screens.

ii) Defining Browser Menu Tree

Browser menu tree will be defined in the script generated for *smtb_function_description*.

The following labels has to be maintained for generation of proper script Main Menu: LBL_{function id}_MAIN_MENU Sub Menu 1: LBL_{function id}_SUB_MENU_1 Sub Menu 2: LBL_{function id}_SUB_MENU_2 Description: LBL_{function id}_DESC *Example: For STDCIFD, following labels has to be maintained* LBL_STDCIFD_MAIN_MENU, LBL_STDCIFD_SUB_MENU_1, LBL_STDCIFD_SUB_MENU_2, LBL_STDCIFD_DESC

Refer <u>Development_WorkBench_Screen_Development-I.docx</u> for detailed explanation on preferences

4.3 Data Sources

- Right Click on Data Sources; click on Add. Add table window gets opened.
- If user knows the exact table name, he can enter name directly; else go to List Of values to get the list of tables available. Select the required table from the list.

ORACLE FLEXCUBE Develo Browser - Function Generation	ment Workbench for Universal Banking	DEMOUSER Windows Options Sign Out
Action New • Function Id STDCIFD Save XML Path CHRADTOCL	Function Type Parent	G ≥ E 7 G ⇒ Function Category Mainlanance → Header Tampitale None → Foolar Tempitale None →
Search Preferences DataSource UstOvalues DataBlocks Screens FieldSata Actions CallForms LaunchForms	AddTable Table Rame Table Name STTM_CUSTOMER% Search Reset	X * ^
Summary	Table Name Table Name STTM_CUSTOMER_ALTERNATE_BRANCH STTM_CUSTOMER_CAT STTM_CUSTOMER_CAT STTM_CUSTOMER_NAM_MASTER STTM_CUSTOMER_PARAM STTM_CUSTOMER_PARAM STTM_CUSTOMER_PARAM STTM_CUSTOMER_SRC_DETALS STTM_CUSTOMER_SRNO STTM_CUSTOMER_SNNO STTM_CUSTOMER_UNUSED STTM_CUSTOMER_VW	

Fig 12.5: Adding Data Sources for the Function id

- Select Master as Yes if added data source is Master Data Source for the screen. Every function id should have one master data source..
- **Primary Key columns** (i.e. Pk Cols) and **Primary Types** (i.e. Pk Types) are mandatory. If it is already maintained in user schema in STTB_PK_COLS it will populated automatically otherwise user needs to enter values without fail. If user misses Pk cols and Pk Types package generation will fail. *Note: Master Data Source cannot have any parent.*

Browser -	ment Workbench for Universal	l Banking			Windo	ws		DEMC		
Function Generation									7.13	- >
							×		7 9	14
Action New -		Function Type Parent	() •)		Function Category Maintenance	•				
Function Id STDCIFD		Parent Function			Header Template None -					
Save XML Path DIRADTOOL		Parent Xml			Footer Template None					
Search	Data Source Details							- 4	- 4	9
 Preferences DataSource STTM_CUSTOMER ListONalues DataBlocks Screens FieldSets Actions CallForms LaunchForms Summary 	Master Ye Relation Type Of		0.0	Parent Relation Where Clause Default Order By Type	▼ Normal ▼ ☐ Mandatory		n n n			

Fig 12.6: Providing master Data Source Properties

• Right Click on Added Table (STTM_CUSTOMER) to add fields to the table. Popup window gets opened with available columns in data source. Select the required fields and click ok. Selected will get added to the Data Source Tree.

DRACLE' FLEXCUBE Develop	ment Workbench for Unive	ersal Banking						[DEM	ous	SER
Browser -						Wind	lows	Optic	ns	Sign C	Dut
function Generation											- ×
								×		Ø (4
Action New ·		Function Type Parent				Function Category Maintenanc	• •				
Function Id STDCIFD		Parent Function				Header Template None -					
Save XML Path D/RADTOOL		Parent Xml				Footer Template None					
Search	Data Source Detai	ls							-	-	9 .
Dreferences	Data Source	STTM_CUSTOMER			Parent	,	•				
DataSource STTM_CUSTOMER	Master	Yes -			Relation			9			
ListOfValues Add	Relation Type	One To One 💌			Where Clause			000			
DataBlocks Delete	Multi Record	No 🔻			Default Order By	and the second sec		2			
Screens		CUSTOMER_NO		2	Type	Normal •					
FieldSets	PK Types	VARCHAR2	0			Mandatory					
CallForms	Upload Table										
a LaunchForms											
Summary											
Summary											

Fig 12.7: Including Data Source Fields for the Data Source

	Function Type Parent Parent Function Parent Xms Sect Fields		×		Function Category Maintenance + Header Template None + Footer Template None	. .	3 7 9 4
Function Id STDGIFD Save Xell, Pelh CritRADTOOL Search Se	Parent Function Parent Xms		×		Header Template None •	Contraction of the	11 17 W 44
Function Id STDGIFD Save XML Path D/RADTOOL Search Se	Parent Function Parent Xms		×		Header Template None •	•	
Save XML Path D/(RADTOOL Search Se	Parent Xmr		×				
Search Se			×		Easter Tampista Mona		
(Second S	elect Fields		×		Podus remplate Notio	*	
							+ = 47
Deferances				Parent			
DalaSource STTM_CUSTOMER	V CUSTOMER_NO	VARCHAR2		Relation		20	
DistOfvalues	V CUSTOMER_TYPE	CHAR		Where Clause		22	
DalaBlocks	V CUSTOMER_NAME1	VARCHAR2	-	Delaull Order By Type	Normal 💌		
📴 FieldSals	ADDRESS_LINE1 VARCHAR2	(1)24	T Mandalory				
CallForms	ADDRESS_LINE3	VARCHAR2					
aunchForms	ADDRESS_LINE2	VARCHAR2					
Summary	ADDRESS_LINE4	VARCHAR2					
	COUNTRY	VARCHAR2					
	SHORT_NAME	VARCHAR2					
	2 NATIONALITY	VARCHAR2	E				
	V LANGUAGE	VARCHAR2					
		Ok Cancel	I				
		Angeorg and Angeorg and and and					

Fig 12.7: Selecting Data Source Fields for the Data Source

Data Source Field Properties:

Only max length can be modified by the developer in data source field properties. Rest will be defaulted from table definition

Browser -	ment Workbench for Unive	rsal Banking		Windows	 NOUSER Sign Out
unction Generation					70-
Action New Function Id STDCIFD Save XML Path D/RADTOOL earch	Action New Action New Function Id STDCIFD Save XML Path D:RADTOOL Data Source Field Details Preferences STM_CUSTOMER_NO CUSTOMER_NO CUSTOMER_NON CUSTOMER_NON CUSTOMER_NAME1 CUSTOMER_NAME CUSTOMER_NA	Parent Function Parent Xml		Function Category Maintenance - Header Template None - Footer Template None -	esh - G)
Preferences The Customer No Customer No	Column Name Block Name	CUSTOMER_NO	Data Type MaxLength Upload Table Column	VARCHAR2 9 Not Required in Upload Tables	

Data model of a single function id would include multiple tables .All the tables needs to added in the function id. Note the following while adding child data sources

Adding Child Data Source:

- Select Multi Record value as Yes if child data source is Multi record table.
- Child Data Source should always be associated with a parent.
- Relation is mandatory between parent and child. While giving relation, parent data source should come in left side of the relation.

DRACLE FLEXCUBE Develop	nent Workbench for Unive	rsal Banking			D	EMOL	JSEF
Browser .				Windows	s Option	s Sigr	_
unction Generation						E 🖗	
Action New ·		Function Type Parent		Function Category Maintenance			
Function Id STDCIFD		Parent Function		Header Template None •			
Save XML Path D/RADTOOL		Parent Xml		Footer Template None	•		
earch	Data Source Detail	s				- 4	- 9
 Preferences DataSource STTM_CUSTOMER STTM_CUST_GROUP GROUP_JD USTOMER_NO RELATIONSHIP ListOValues DataBlockis Screens FieldSets Actions CallForms LaunchForms Summary 	Master Relation Type Multi Record PK Cols •	STTM_CUST_GROUP No • One To Many • Yes • GROUP_ID-CUSTOMER_NO VARCHAR2-VARCHAR2	Parent Relation Where Clause Default Order By Type	STTM_CUSTOMER STTM_CUSTOMER_NO STTM_CUSTOMER_NO STTM_CUSTOMER_NO Mormal Mandatory	0018		

Fig 12.7: Providing properties for Child Data Source

Note: A data source cannot be parent to itself.

Note the following while adding data sources:

- i) If the data source is designed with relation type as 1: N with its parent, then it should have at least one more Pk col than its parent (assuming relationship is based on Pk cols).
- ii) Master data source needs to have the audit columns if footer template is Maint audit; but those should not be added to data source fields as system will handle it

Refer <u>Development_WorkBench_Screen_Development-I.docx</u> for detailed explanation on data sources.

4.4 Data Blocks

• Block Name should start with BLK_<short Name equivalent to data source but not exactly same as Data Source name>.

ld Block		×
Block Name	BLK_CUSTOMER	
	Ok Cancel	

- Select Parent block if added block is not Master Block.
- Select Multi Record (Yes/No) based on this value, available data sources will displayed in data source available text area.

ORACLE FLEXCUBE Development Workben	th for Universal Banking - Windows Internet Exp	plorer	And Manual State	1000		
ORACLE FLEXCUBE Develop	nent Workbench for Universal Banking				DE	MOUSER
Browser .				Windows	Options	Sign Out
Function Generation						-
					🗄 🗶 🗏	77 🧐 🗟
Action New 💌	Function T	Type Parent 🔻		Function Category Maintenance 🔻		
Function Id STDCIFD	Parent Fund	tion		Header Template None 🔻		
Save XML Path D:IRADTOOL	Parent)	Xml		Footer Template None	•	
Search	Block Properties				+	- 🗷 🌍
 Preferences DataSource STTM_CUSTOMER STTM_CUST_GROUP ListOtValues DataBlocks BLX_CUSTOMER Screens FieldSets Actions CallForms LaunchForms Summary 	Block Name Block Title Parent Relation Type Block PK Fields	×	XSD Node XSD Node Annotation Master Block Multi Record Block Type	Customer Yes V Normal V ce Added		
	Fig 12.9: Provid	ling properties fo	or Data Block			

• Select the required data source and click move button to attach Data Source to the block

ORACLE FLEXCUBE Development Workber	nch for Universal Banking - V	Vindows Internet Explorer	Name and Address	 and the local		1. 1.				x
ORACLE FLEXCUBE Develop	ment Workbench for Univ	ersal Banking						DEM	DUSE	ER
Browser .						Windows	Opti	ons S	Sign Ou	ıt
Function Generation										- ×
						6	X		۶ 🤘	4
Action New -		Function Type Parent	•		Function Category	Maintenance 👻				
Function Id STDCIFD		Parent Function			Header Template	None 👻				
Save XML Path D:\RADTOOL		Parent Xml			Footer Template	None -				
Search	Block Properties							4 -	x 4) ^
Preferences DataSource B STM_CUSTOMER B STM_CUST_GROUP ListOfValues DataBlocks BLK_CUSTOMER Screens FieldSets CaliForms CaliForms LaunchForms Summary	Block Name Block Title Parent Relation Type Block PK Fields	BLK_CUSTOMER	e Available		Customer Yes V No V Normal V					

Fig 12.10: Attaching Data Sources to Data Block

Adding multi record data source to data block:

User on selecting Multi record Yes in data block properties all the data sources with multi record Yes will be populated. *Multi Data Source once used to one block won't available for reuse where as single record data source can be used in multiple blocks*

Select Block Fields:

- Right click on added block. Select Fields window will get opened. Developer needs to check the right side check box to add the required fields.
- **Field Name**: It should not be the same as column name .Special characters are also not allowed in the field name (including underscore and space)
- Label Code: It will be automatically populated based on field name.

ACLE FLEXCUBE Develo	pment Workbench for Universal B	anking				DE	MOUSE
owser 🗸					Windows	Options	Sign Out
ction Generation							
							77 🌒
Action New 💌		Function Type Parent	¥		Function Category Mainlenance		
Function Id STDCIFD		Parent Function			Header Templale None 👻		
Save XML Pain DARADTOOL		ParentXml			Fooler Template None	•	
rch	Select Fields & Add UI Fields				×	+	- 2
	DataSource fields UI Fields	1					
DataSource	Databource lields Of Fields	2			omer		
STTM_CUSTOMER	Datasource STT	M_CUSTOMER -			*		
STTM_CUST_GROUP IstOfvalues	Column Name	Field Name	Label Code	*	•		
DelaBlocks	CUSTOMER_NO	CUSTNO	LBL_CUSTNO		nal 👻		
BLK_CUSTOMER	CUSTOMER_TYPE	CUSTTYPE	LBL_CUSTTYPE				
Screens FieldSels	CUSTOMER_NAME1	CNAME	LBL_CNAME		led		
Actions	ADDRESS_LINE1	ADDR1	LBL_ADDR1				
CaliForms	COUNTRY	CNTY	LBL_CNTY				
LaunchForms Summary		NLTY	LBL_NLTY				
		LANG	LBL_LANG				
	V						
	55 F						
				+			
				01			
				Ok C	ancel		

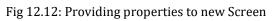
Fig 12.11: Adding Block Fields to Data Block

Refer <u>Development_WorkBench_Screen_Development-I.docx</u> for detailed explanation on data blocks and block field properties

4.5 Screens

- Right click on Screens node to add a new screen
- Screen Name should start with CVS_<Name>...
- By default screen are divided into 3 parts.
- One Main Screen is Mandatory.
- Tabs can be defined on any of the screen portions as required
- User can add sections to tabs.
- Each section can be divided into partitions.

ORACLE FLEXCUBE Development Workber	oench for Universal Banking - Windows Internet Explorer					х
ORACLE FLEXCUBE Develop	opment Workbench for Universal Banking		DEI	иои	ISE	R
Browser -		Windows	Options	Sign		
Function Generation				_	-	. ×
			× =	V	4	4
Action New -	Function Type Parent - Function Category Ma	intenance 👻				
Function Id STDCIFD	Parent Function Header Template No	ne 🔻				
Save XML Path D:\RADTOOL	Parent Xml Footer Template No	ne 🔻				
Search	Screen Details		-	Aï 🖸	9	^
Preferences ListOfValues ListOfValues ListOfValues Constitution CVS_MAIN ListOfValues ListOfValues CVS_MAIN ListOfValues	Screen Name CVS_MAIN Screen Title LEL_CUST Screen Size Small Exit Button Type Default Cancel CVS_MAIN Exit Button Type Default Cancel CVS_MAIN CVS CVS_			•		
FOOTER FieldSets Actions CallForms LaunchForms Summary	Argument Name Source Block Source Field Argument Value Target Block	Target Field	Active			



ORACLE FLEXCUBE Develop	nent Workbench for Universal Banking		Windows	DEN Options	IOUSE Sian Out	
Function Generation						
Action New Function Id STDCIFD Save XML Path DXRADTOOL	Function Type Parent Parent Parent Parent Xml	Function Category Mainle Header Templale None Fooler Template None	enance 💌		P 100	
Search	Tab Details		Dependent	Fields 🥠	- 🛛 🏟	, T
Preferences DataSource ListOfvalues DataBlocks CVS_HAIN HEADER	Screen Name CVS_MAIN F Visible Tab Name TAB_MAIN Tab Label Tab Type Add Section	×				
TAB_HEADER TAB_HEADER TAB_MANN TAB_FOOTER TAB_FOOTER FieldSels Actions	Section Name SEC_CUST Ok Cancel					
Califorms LaunchForms Summary						
	Fig 12.12. Creating new section in TAD MAIN in th					

Fig 12.13: Creating new section in TAB_MAIN in the body of screen CVS_MAIN

.

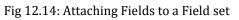
Ø ORACLE FLEXCUBE Development Workben	ch for Universal Banking - Windows Internet Explorer	
ORACLE' FLEXCUBE Develop	ment Workbench for Universal Banking	DEMOUSER
Browser +		Windows Options Sign Out
Function Generation		_ ×
		🖫 🗷 🗏 🖗 🤤
Action New 👻	Function Type Parent	Function Category Maintenance -
Function Id STDCIFD	Parent Function	Header Template None 💌
Save XML Path D:\RADTOOL	Parent Xml	Footer Template None -
Search	Section Details	- K 🥱 🕯
Preferences DataSource ListOValues DataBlocks DataBlocks Screens	Section Name SEC_OUST IV Visible Section Label LBL_SECT II Collapse	
CVS_MAIN HEADER	Partition Details	+-
TAB_HEADER	Partition SI No Partition Name	Width Sub-partitions
BODY TAB_MAIN	1 PART1 PART2	50 -
SEC_CUST	2 PART2	50 🔻
FOOTER TAB_FOOTER FieldSets CallForms CallForms Summary		.*
	Fig 12.14: Defining partitions for the Section	on

4.6 Field Sets

A group of fields can be grouped together in a Field set which can be placed together in the screen

- Field Set Name should start with FST_<>.
- Select the Block adding to field set.
- All fields available to the block will be displayed in to the data block fields text area. Move fields from data block fields to Field set fields.
- The order of fields in *field set fields* will reflect in the screen as well

	ent Workbench for Universal Banking			DEMOUSER
Browser -			Win	dows Options Sign Out
unction Generation				-
				🖫 🗵 🗏 🚱 🛛
Action New 👻	Function Type Parent		Function Category Maintenan	ce 🔻
Function Id STDCIFD	Parent Function		Header Template None	-
Save XML Path D:\RADTOOL	Parent Xml		Footer Template None	•
earch	Fieldset Properties			- 🛚 🌍
 Preferences DataSource ListOVAlues DataBlocks Screens FieldSets FST_CUST1 CST_CUST2 Actions CallForms LaunchForms Summary 	Fieldset Name Fieldset Label Data Block Mutil Record View Type Fieldset Height CUSTNO CUSTNO CUSTNYPE CNAME ADDR1 CNTY LANG	Screen Portion Tab Name Section Name Partition Name Number Of Rows	S_MAIN	 Horizontal Fieldset ReadOnly Navigation Button ✓ Visible



DRACLE' FLEXCUBE Developm Browser	nent Workbench for Universal Banking	DEMOUSE Windows Options Sign Ou
inction Generation		· · · · · · · · · · · · · · · · · · ·
Action New Function Id STDCIFD Save XML Path D:RADTOOL arch	Function Type Parent Parent Function Parent Xmt Fieldset Properties	Function Category Maintenance Header Template None Footer Template None
 Preferences DataSource ListOValues CataBlocks Screens FieldSets FST_CUST1 FST_CUST2 Actions CallForms LaunchForms Summary 	Fieldset Name Fieldset Label Data Block Mutil Record View Type Fieldset Height Data Block Fields CNTY NLTY LANG	Screen Name CVS_MAIN Horizontal Fieldset ReadOnly Nawigation Button Section Name Partition Name Visible Visible Visible CUSTNO CUSTNO CUSTYPE CADR1 ADDR1 Image: Custor of the section of the sect

• Select the screen portion (Header/Body/Footer) where this field set has to be placed. Select remaining details like tab, section and partition.

DRACLE FLEXCUBE Developm				
Developin	ent Workbench for Universal Banking			DEMOUSER
Browser 🗸			Windows	Options Sign Out
unction Generation				_ ×
			l	🗄 🗵 🗏 7 🧃 🔿
Action New -	Function Type Parent	Function	Category Maintenance -	
Function Id STDCIFD	Parent Function		Template None -	
Save XML Path D:\RADTOOL	Parent Xml	Footer	Template None	•
earch	Fieldset Properties	~		- 🛛 🗐 🔺
 Preferences Treferences LataSource LataSource LataSocks Screens FildSets FST_CUST1 FST_CUST2 Actions CallForms LaunchForms Summary 	Fieldset Name FST_CUST1 Fieldset Label FST_CUST1 Data Block BLK_CUSTOMER Mutit Record No View Type Single Fieldset Height Data Block Fields CNTY NLTY LANG	Screen Name Screen Portion Tab Name Section Name Partition Name Number Of Rows FieldSet Fields CUSTNO CNAME CUSTNO CUSTNO CONAME ADDR1	•	 Horizontal Fieldset ReadOnly Navigation Button ✓ Visible

Fig 12.15: Providing details where Field Set has to be placed

Once fields are added to field set, developer can check the preview of the designed screen. Right click on Screen Name and click on Preview.

🔶 Main		N 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997
🗗 New 🖾 Enter Query		
Customer No		
Name		
Туре		
Address		
Maker	Date Time:	
Checker		
	Date Time:	Exit
Mod No	Record Status	
Midd No	Authorization Status	

Fig 12.16: Preview of the designed Screen

Adding Multi entry block to field set.

- On selecting a multiple block, Multi Record Property will be defaulted to Yes..
- In case of Multi record, View type can be either Single or Multiple (By Default).

🔶 Main			×
🗗 New 🤄 Enter Query			
Customer No Name Type Address			
I≪ 1 of 1 ▶ ▶	Go to Page		+ - =
Group Id	Customer No	Relation	*
<		3 11 1.	*
Maker		Date Time:	
Checker		Date Time:	
Mod No	Re	cord Status	Exit
		tion Status	

Below image shows a multiple view multi record field set



• For multi record single view navigation button should be checked.

		king - Windows Internet Explorer		-			
RACLE' FLEXCUBE Develop	ment Workbench fo	r Universal Banking					DEMOUSE
Browser -						Windo	ws Options Sign Out
nction Generation							
							📱 🗶 🗏 7 🧃
Action Load		Function Type Parent			Functi	on Category Maintenance	*
Function Id STDCIFD		Parent Function				er Template None 🔻	
Save XML Path STDCIFD_RAI	BROWSE	Parent Xml				er Template Maint Audit	•
Save XML Faul STDCIPD_RAL	BROWGE	Farent Am			FUUL	er remplate maint Audit	•
arch	Fieldset Prop	perties					- R 🧐
🚞 Preferences	Fieldset	Name FST_CUST2	Sc	reen N	ame CVS_MAIN	•	🔲 Horizontal Fieldset
a DataSource	Fieldset	Label	Scr	een Po	ortion Body	-	ReadOnly
STTM_CUSTOMER CUSTOMER_NO	Data	Block BLK_GROUP		Tab N	Jame TAB_MAIN	-	Navigation Button
CUSTOMER_TYPE	Multi R	ecord Yes -	Se	ction N	Jame SEC_GROUP	~	Visible
CUSTOMER_NAME1	View	Type Single		ition N		•	1.
DDRESS_LINE1	Fieldset		Numb				
COUNTRY							
DATIONALITY							
CANGUAGE STTM_CUST_GROUP		Data Block Fields			Field Set Fields	Subpartition Name	
GROUP_ID				_			
CUSTOMER_NO					ROUP_ID	· · · · · · · · · · · · · · · · · · ·	
Carl RELATIONSHIP				C	UST_NO	•	
ListOfValues			DD	🗆 R	ELATION	•	
DataBlocks			W				
BLK_COSTOMER			44				
Screens							
🖃 🧰 CVS_MAIN							
BODY							
TAB_MAIN SEC_CUST							
SEC_GROUP							
i FieldSets							
DIFST_CUST1							
FST_CUST2							

6 or

Fig 12.18: Properties for Single View Multi Record Field set

🔶 Main		×
🗗 New 🔄 Enter Query		
Customer No Name Type Address		(1 of 1)
Group Id Customer No Relation		
Maker Checker	Date Time: Date Time:	Exit
Mod No	Record Status Authorization Status	

Below figure shows the preview of a single view multi record field set

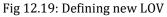
Fig 12.18: Preview for Single View Multi Record Field set

4.7 LOV

List Of values can be defined for the function id using LOV node

- To add LOV right click on List of Values Node. LOV Name should start with LOV_<name>.
 - *Example: LOV_COUNTRY.*
- Enter valid query and click on populate button

Function Generation						_ >
				🗄 🗶	E 77 (9 🔿
Action Load Function Id STDCIFD Save XML Path STDCIFD_RAI	Function Type Pare Parent Function BROWSE Parent Xml		Header Tem	gory Maintenance 👻 plate None 👻 plate Maint Audit 👻		
Search	List Of Values Details				- 🔊	9
Preferences DataSource STM_CUSTOMER STM_CUST_GROUP ListONalues ListONalues ListONalues ListONalues	LOV Name * LOV_OCUNTRY LOV Query select country_code,descri	ption from sttm_country where auth_stat = "	A' and record_stat = 'O'	9	Populate	
🗄 🧰 DataBlocks	Query Columns Data Type	Visible Reduction Field	Reduction Field Type	Reduction/Column Label	^	í I
					-	
Actions CaliForms SumarkForms Summary						



LOV Query	×
select country_code,description from sttm_country where auth_stat = 'A' and record_stat = 'O'	
Ok Cancel]

Fig 12.20: Providing LOV query

Function Generation												_ ×
									×	= 1	3	⇔
Action Load + Function Id STDCIFD Save XML Path STDCIFD_RAI		ent Function Parent Xml				Head	ion Category Maintenance ler Template None ter Template Maint Audit	▼				
Search	List Of Values Details									-	Aï 🗳) ^
Preferences DataSource DataSource Distrin_cOUSTOMER Distrin_cOUST_GROUP ListOValues LoV_CCUNTRY	LOV Name * LOV_OCI LOV Query Select cou		from sttm_co	untry where auth_stat :	= 'A' and record	_stat = 'O']		(Рори	ulate	
DataBlocks	Query Columns	Data Type	Visible	Reduction Field	Reduction	n Field Type	Reduction/Colu	mn La	bel		^	
Greens FieldSets	COUNTRY_CODE	VARCHAR2 -	Yes 🔻	Yes 👻	TEXT	- /	LBL_CNTRY		× E			
CallForms CallForms LaunchForms Summary	DESCRIPTION	VARCHAR2 -	Yes 🔻	Yes -	TEXT	• (LBL_COUNTRYCD		*=		Ŧ	

Fig 12.21: Providing LOV details

- Redn/Col Labels are mandatory. If user won't provide will get error on click of LOV button after deployment in FLEXCUBE
- After defining LOV go to block and corresponding field where the LOV has to be attached.

Block Field Properties to attach LOV to the field

- **Display Type:** Select display type as Lov.
- Lov Name: Select the required Lov name from the list of all defined LOV's.
- Click on return fields tab. The result fields maintained in the LOV query will be populated on click of *Default from Lov Definition* button

- Select the desired field (and its block)to which the result of the LOV query should be defaulted
- If return field is not required to be defaulted to any field in the screen, return field value can be left blank

Function Generation		-
		🖬 🗵 🗏 🐬 🇐 🤇
Action Load 👻	Function Type Parent	Function Category Maintenance
Function Id STDCIFD	Parent Function	Header Template None -
Save XML Path STDCIFD_RAL	BROWSE Parent Xml	Footer Template Maint Audit 👻
Search	Block Field Properties	- A 📮 🖗
Preferences DataSource DataSource LISTM_CUSTOMER LISTM_CUSTGROUP LISTOValues LISTOValues LISTOVALUES LISTOVALUES LISTOVALUES LISTOVE CUSTTYPE COMME ADDR1 CONTY LANG LANG LANG LANG LANG LANG LANG LANG	Field Name CNTY Field Label LBL_CNTY Field Label DataSource STTM_CUSTOMER Column Name COUNTRY Data Type Varchar2 V Display Type Lov V Item Type Database Item V Parent Field Related Field V LOV Name V Fieldset Name	XSD Tag CNTY Required XSD Annotation Visible Field Size Read Only Maximum Length Calender Text Minimum Value Popup Edit Required Maximum Decimals Uppercase Only Maximum Decimals Visible TextArea Rows Required TextArea Columns Input by LOV Only Default Value Not Required in Xsd Preview Value Report Parameter
Screens FieldSets Actions CallForms	Custom Attributes Events Bind Variables Return Fields R Return Fields Mapping Ouery Column	elated Field
aunchForms	COUNTRY_CODE BLK_CUSTOME	
🚞 Summary	DESCRIPTION BLK_CUSTOME	
	Fig 12.22: Attaching	

Use of Bind Variable

If the list of values should be based on any other field value from the screen, bind variables can be used.

Example:

Define lov as shown in below query; where clause should contain condition with '?'.

SELECT cust_ac_no, branch_code, ccy from sttms_cust_account where cust_no = ? and record_stat = 'O' and once_auth = 'Y' and ac_stat_de_post = 'Y'

In the block field, after selecting return fields, click on bind variables tab. Click on **Default from Lov Definition** button. New rows will be created depending on the number of bind variable provided in the LOV query. Select the bind filed in the screen (and its block) for the LOV. Data type of the field has also to be selected.

Action Last w	Function Type Parent	143	Function Category U	famance (er)			
Function Id STDCINF	Parent Function		Header Template None				
Save XIL Path D'RADTOOLS	Parent Xml		Footer Template Main	t Audit 💉			
larch	Block Field Properties					- 2	4
Preferences DataSource DataSource DataSource LOV_COUNTRY CLOV_ACCOUNT OutaBlocks CUSTNO CUSTNO CUSTNO CUSTNO COUNTRY NAME ADDRLN1 COUNTRY NAME CUSTNO CUSTNO CUSTNO CUSTNO CUSTNO CUSTNO CUSTNO Soreens Adons		Ratum Fields	Data Tipe Vardhar2 DataSource STTMS_CUST_C Max.Length Field Ste Column Name CUSTOMER_NO Defaut Value Preview Value Accessive/Code Test-real Cols Max.Val Mask.Id Off Line LOV Name Image Source		Popup Edit Regd Required Required Visible Input by LOV Only Calender Text Select Multiple Uppercase Only Reg in Xad Report Parameter Read Only		
Califforms LaunchForms Summary	Bind Variables Mapping Biock Name BUX_CUSTONER	×	Ce Bind Variable CUSTNO	Guil from LoV definitio Distatype STRIteG			

Fig 12.23: Defining bind variable for the LOV

4.8 Attaching Call forms

Maintenance Call forms can be attached to a maintenance screen. Refer the document <u>14-</u> <u>Development of Call Form.docx</u> for developing call forms

Attaching Call forms

- Add button to block to launch call form on button click.
 - Right click on Block
 - Select Add fields. Select fields and Add UI field's window will be launched
 - Select UI Fields tab. Click add row button. Enter button name and click ok.
 - Select display type as button and enter field label.

Select	Fields & Add	Ul Fields		×
Datas	Source fields	UI Fields		
			+-	
		Field Name Da	ata Type	^
V	BTM_MIS		•	
				-
			Ok	Cancel
			UK	cancel

Fig 12.24: Defining Button field

• Add Call form details to Call form node

						E 7 🧐
Action Load -		Function Type Parent	F	Function Category Main	tenance 🔻	
Function Id STDCIFD		Parent Function		Header Template None		
Save XML Path STDCIFD_RAI	BROWSE	Parent Xml		Footer Template Main	t Audit 👻	
arch	Call Form Details					9
📴 Preferences a 📴 DataSource						
B D STTM_CUSTOMER				Screen Arguments	Dependent	Fields+-
TTM_CUST_GROUP ListOfValues	Function ID	Parent Data Block Parent DataSo	Irce Relation	Relation Type	Callform Screen	Display 1 ^
LISTONATES	MICCUSTM	BLK_CUSTOMER - STTM_CUSTO	IER - TTM_CUSTOMER.COSTOMER_NO	= 🔁 🛛 One To One 🔻	-	Button
CNTY ANG LANG ETM_MIS ETM_MIS Screens FieldSets Adions CallForms CallForms Summary						
	٠		"			

Fig 12.25: Defining details of the Call form to be attached in call form node

- Add event to button.
 - On selecting event type as call form or launch form or sub screen button will be displayed on bottom of the screen.
 - If user needs to place button position in desired place on the screen, event type should be Normal .User has to write code in release specific JavaScript file to launch the screen

									-
							:	× = 1	و 😼 🌾
Action Load 👻		Function Type	Parent	•		Function Category	Maintenance 🔻		
Function Id STDCIFD		Parent Function				Header Template	None 👻		
Save XML Path STDCIFD_RAI	BROWSE	Parent Xml				Footer Template	Maint Audit 🛛 👻		
earch	Block Field Prope	ties						- AL	Q 9
Preferences DataSource STM_COUSTOMER STTM_CUST_GROUP ListOfValues LOV_COUNRTY	Field Name Field Label DataSource Column Name Data Type	LBL_MIS	*1	XSD Ann	eld Size *		, ⊓ □	Required Visible Read Only Calender Te Popup Edit I	
	Display Type Item Type Parent Field Related Block Related Field	Text Control Control	•		a Rows			Uppercase LOV Validati Required Input by LOV Not Require Report Para	on / Only d In Xsd
INLTY LANG BTM_MIS BLK_GROUP Screens CVS_MAIN	LOV Name Off Line LOV Name Fieldset Name Custom Attributes	vents Related Field	• •		Mask Id	H		+ -	
	Event Nam	e Fund	ction Name	Event Type	Button Screen	CallForm Nam	ne Screen	Name	•
 BODY TAB_MAIN SEC_CUST SEC_CUST SEC_OROUP FieldSets Addons CaliForms LaunchForms Summary 	vonunload	•		Caliform •	CVS_MAIN -	MICCUSTM -	CVS_CUS	TO	r.

• Check the preview.

🔶 Main				×
🖹 New 🦻 <u>Enter Query</u>				
Customer No Name Type Address				
I≪ 1 of 1 ▶ ▶	Go to Page		+ - =	
Group Id	Customer No	Relation	^	
			Ψ	
•			4	
MIS Change Log				
Maker	I	Date Time:		
Checker		Date Time:	E	xit
Mod No		ord Status		
	Authoriza	tion Status		

Fig 12.27: Preview of the screen with the Call Form button

4.9 Adding Summary

1) Add entry in Preferences node for Summary screen

				1 🗏 7 🧃
Action Load - Function Id STDCIFD Save XML Path STDCIFD_RAI	Function Ty Parent Funct BROWSE Parent X Preferences	ion	Function Category Maintenance Header Template None Footer Template Maint Audit	
Preferences DataSource STTM_CUSTOMER STTM_CUSTGROUP ListOV/alues BLK_CUSTOMER LLK_CUSTMOR CUSTNYPE CNAME ADDR1 CNTY	 ✓ Head Office Fu ✓ Logging Requi ✓ Auto Authorizat ✓ Tank Modificati ✓ Field Log Requi Multi Branch Act ✓ Excel Export Ref 	red Module Description ion Branch Program Ir ons Process Cod uired SVN Repository URI ccess Transaction Block	Static Maintenance	
🚞 NLTY 🛅 LANG			Control	String + -
BTM_MIS	Function Id	Module *	Module Description	^
BLK_GROUP Screens	STDCIFD		Static Maintenance	
CVS_MAIN CVS_MAIN FieldSets Actions Califorms LaunchForms Summary	STSCIFD	ST A	Static Maintenance	~

- 2) Click on Summary Node.
 - Enter Summary title .Select label code from lov.
 - Select Data Block master block and summary blocks will be displayed. Select required block from drop down list.
 - Select Data Source for summary.
 - Select Summary Type.
 - Select Summary Screen size.
 - Enter if any where clause is required.
 - Enter Default order by if required.
 - Enter Multi Branch where clause if required.
 - Attach the fields required in the summary result grid
 - If the field is required as part of filtering, query has to be checked for the particular field
 - Provide the position of fields in Result grid and Summary Query set .

									×	I 1	< 🧃	
Action Load 🔻			Function Type Parent	Ŧ		Function	Category Maintenance	-				
Function Id STDCIFD			Parent Function			Header	Template None 🔻					
Save XML Path STDCIFD_RA	BROWSE		Parent Xml			Footer	Template Maint Audit	•				
arch	Summary	/ Details									0	9
Preferences		Title		# E	Default Where Clause							
DataSource	D	ata Blocks	BLK_CUSTOMER	•	Default Order By			1				
STTM_CUSTOMER		ata Source	STTM_CUSTOMER	•	Multi Branch Where							
B D STTM_CUST_GROUP STTM_CUST_GROUP STTM_CUST_GROUP		mary Type	Summary	•	Clause							
DataBlocks	Summary S		Medium	•	Main Summary Screen							
BLK_CUSTOMER	000000000000000000000000000000000000000	0.0011 0.20				C WebServ Required						
CUSTNO	-		312	5		Required						
CUSTTYPE	Data Block	Fields C	ustom Buttons Fields Ordering	1								
DAME												
CNTY												
DITY			Data Block Fields		Fields Selected	Query	LOV Name					
BTM_MIS					CUSTNO			•]			
BLK_GROUP					CNAME			Ŧ				
🫅 Screens 🗉 🚞 CVS_MAIN				1.00	CUSTTYPE			Ŧ	1			
FieldSets				Þ	ADDR1			Ŧ	1			
actions				4	CNTY			Ŧ	ĺ			
🚞 CallForms 词 LaunchForms					NLTY			Ŧ	ĺ			
Summary					LANG			Ŧ				
									1			
		I							1			

Summary Preview

Right click on summary node and click on preview.

🔷 🛃 Exe	ecute Query 💓 Advance	ed Search 🏼 🏟 Res	et 🖓 Clear All							×
	Authorization Status Customer No	•	<u>×=</u>		Rec	ord Status	•			
Reco	rds per page 15 🔻 🚺	🛯 1 of 1 🕨 🔰		e						*
	Authorization Status	Record Status	Customer No	Name	Туре	Address	Country	Nationality	Language	
										=
										-
•									•	•
									Exit	

Fig 12.29: Summary Screen Preview

4.10 Amendable fields Maintenance

Amendable Fields

If user needs to modify data of a particular field on unlock, in Workbench developer has to maintain fields as amendable.

- Click ACTIONS node.
- Click on Amendables button next to the action for which the field has to be made amendable
- Select the fields in each block which user can modify for the selected action.

ndable DetailsQUERY		
Data Blocks	DataBlock Fields	;
JSTOMER ROUP	New Allowed Delete Allowed All I	Records 🔲 Mandatory
	Field Name	Amendable
	CUSTNO	
	CUSTTYPE	
	CNAME	
	ADDR1	
	CNTY	
	NLTY	
	LANG	
	BTM_MIS	
		Ok Cance

Fig 12.30: Maintaining amendable fields

5. Generation and Deployment of files

Generate Files

• Click on generate button select the required files to generate and click on Generate button.

ation		×	4	lela Data	Others	
Error Description Request successfully Processed		Error Code RD-SAVE-007	Menu Details Datasource Details LOV Details Menu Details Menu Details Screen Details	Label Cretails Lock PK Columns Function Call Forms Gateway Details Notification Details	Forms Screen Html ils VUpload Table Trigger etails VUpload Tables Definition	
		0% of DownLoadFile from 10.184	.132.100 Completed	nction Parameters	Archive Table Definition	
		File Download		X Hyb Let date		
		Do you want to open or sa	ve this file?			
		Name: RAD.ZIP		ые Туре	Status	
		Type: WinRAR ZIP archive			Generated 👻	
		From: 10.184.1	32.100		Generaled *	
		Open Save Cancel			Generated 👻	
					Generaled *	
STDCIFDCVS_MAINTAB_FOOTER html slpks_stdoid_main.spc		While files from the Inte	ernet can be useful, some files can potenti	ally	Generated 👻	
		While these from the internet can be useful, some thes can botentially harm your computer. If you do not trust the source, do not open or save this file. <u>What's the risk?</u>		x'	Generaled *	
	stpks_sldclfd_kernel.spc				Generated 👻	
	slpks_stdcifd_main.sql		SQL	-	Generaled *	
	stpks_sldclfd_kernel.sql		SOL		Generated 👻	
	CSTB_FIELD_LABELSSTDCIFD.INC		INC		Generaled +	
	CSTB_OTHER_LABELSSTDCIFD INC		INC		Generated 👻	
	CSTB_FID_CALLFORMSSTOCIFD.INC		INC		Generaled *	

Fig 12.30: Generation of Files

Deploy files

• Click on deploy button select the required files to deployed to server and click on deploy. On successful deployment status will be displayed as Deployed.

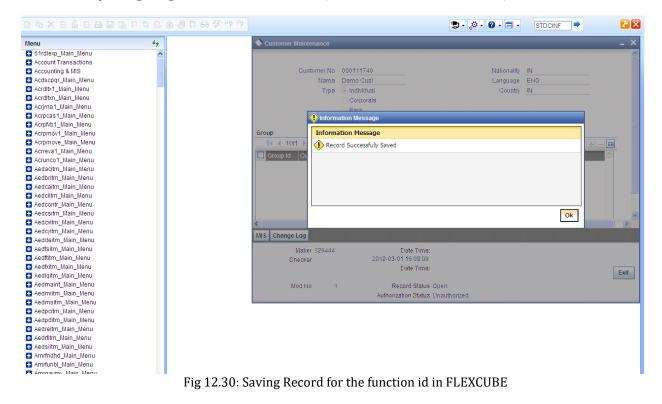
Front-End Files	s System Packages	Hook Packages		Meta Data	Others	-
RadXML Z Screen Xml Z System JS	Main Package Spec Main Package Body Notification Triggers Upload Package Spec Upload Package Body	Kernel Package Spec Kernel Package Body Uluster Package Body Uluster Package Body Uluster Package Body Custom Package Body	Menu Details Datasource Details D Details D LOV Details Block Details Screen Details Amendable Details Call form Details Summary Details	Label Details Block PK Columns Function Call Forms Cateway Details Nothication Details Function Parameters Purge Details	Xsds Xsd With Annotations Screen Html Upload Table Trigger Upload Table Definition Archive Table Definition	
	CSTB_FIELD_LABELSSTDCIFD.INC		IN	С	Deployed -	-
	CSTB_OTHER_LABELSSTDCIFD.INC		IN	с	Deployed 👻	
	CSTB_SUMMARY_INFOSTDCIFD.INC		IN	С	Deployed 👻	
	STTB_AUDIT_PK_COLSSTDCIFD.INC		IN	С	Deployed 👻	
	CSTB_FID_DATA_BLOCKSSTDCIFD.INC		IN	с	Deployed 👻	
	CSTB_FID_DATA_SOURCESSTDCIFD.INC		IN	с	Deployed 👻	
	CSTB_FID_SCR_TABSSTDCIFD.INC		IN	с	Deployed 👻	
0	CSTB_FID_SCREENSSTDCIFD.INC		IN	С	Deployed 👻	
1	SMTB_MENUSTDCIFD.INC		IN	С	Deployed 👻	
2	SMTB_ROLE_DETAILSTDCIFD.INC		IN	С	Deployed 👻	
3	SMTB_FUNCTION_DESCRIPTIONSTDCIFD.IN	IC	IN	с	Deployed 👻	
4	SMTB_FCC_FCJ_MAPPINGSTDCIFD.INC		IN	С	Deployed 👻	
5	STDCIFD RAD.xml		R	ADXML	Generated -	

Fig 12.30: Deployment of Files

Testing

•

- Launch the screen from FLEXCUBE
- Try sample operations on the screen (NEW, MODIFY, QUERY etc)



6. Generated Units

The following units will be generated for a Maintenance screen.

Refer document <u>Development_WorkBench_Screen_Development-II.docx</u> for detailed explanation on the same

6.1 Front End Units

6.1.1 Language xml

This file is an XML markup of presentation details, for the designed Call Form specific to a language.

6.1.2 SYS JavaScript File

This JavaScript file mainly contains a list of declared variables required for the functioning of the screen

6.1.3 Release Type Specific JavaScript File

This file won't be generated by the Tool. It has to be manually written by the developer if he has to write any code specific in that release

6.2 Data Base Units

6.2.1 Static Scripts

The following static scripts generated are required for the proper functioning of a Call Form screen. Refer document on generated units for detailed explanation

i) Menu Details

Scripts for SMTB_MENU and SMTB_FCC_FCJ_MAPPING, SMTB_ROLE_DETAIL, SMTB_FCC_GCJ_MAPPING are required for the functioning of Maintenance screen

- ii) Lov Details
- iii) Amendable Details
- iv) Label details
- v) Screen Details
- vi) Block details
- vii) Data Source Details
- viii) Call form details
- ix) Summary Details

6.2.2 System Packages

The Main Package contains the basic validations and backend logic for the Maintenance function id. The Main package contains the mandatory checks required. It will also contain function calls to the other packages generated by Workbench.

The main package has the below stages for a maintenance form:

- Converting Ts to PL/SQL Composite Type
- Checking for mandatory fields
- Defaulting and validating the data
- Writing into Database
- Querying the Data from database

• Converting the Modified Composite Type again to TS

Each of these stages has a 'Pre' and 'Post' hooks in the Kernel, Cluster and Custom Packages. And these Hooks are called from the Main Package itself

Main Package has the system-generated code and should not be modified by the developer Kernel, Cluster and Custom Packages are the packages where the respective team can add business logic in appropriate functions using the Pre and Post hooks available

6.2.3 Hook Packages

Release specific packages will be generated based on the release type (KERNEL.CLUSTER or CUSTOM). Developer can add his code in the release specific hook package.

The Main Package has designated calls to these Hook Packages for executing any functional checks and Business validations added by the user. The structure for all the Hook Packages are the same, like:

Fn_Post_Build_Type_Structure Fn_Pre_Check_Mandatory Fn_Post_Check_Mandatory Fn_Pre_Default_and_Validate Fn_Post_Default_and_Validate Fn_Pre_Upload_Db Fn_Post_Upload_Db Fn_Pre_Query Fn_Post_Query

These Functions are called from the Main package using the Pre and Post Hooks available in the Main Package. The 3 Hook Packages namely Kernel, Cluster and Custom Packages have similar structure and are for the respective teams to work on.

6.3 Other Units

6.3.1 Xsd

Xsd 's will be generated if gateway operations are required for the particular function id. Maintenance for the same has to be done in *Actions* node

7. Extensible Development

Developer can add his code in hook packages and release specific JavaScript file.

7.1 Extensibility in JavaScript Coding

For release specific JavaScript coding, code has to be written in release specific JavaScript

file.

It follows the naming convention as : (Function Id)_(Release Type).js *Example: Code in STDCIFD_CLUSTER.js is exclusive to cluster release*

This JavaScript file allows developer to add functional code and is specific to release.

The functions in this file are generally triggered by screen events. A developer working in cluster release would add functions based on two categories:

- Functions triggered by screen loading events *Example: fnPreLoad_CLUSTER(), fnPostLoad_CLUSTER()*
- Functions triggered by screen action events *Example: fnPreNew_ CLUSTER (), fnPostNew_ CLUSTER ()*

7.2 Extensibility in Backend Coding

Release specific code has to be written in the Hook Packages generated.

7.2.1 Functions in Hook Packages

Different functions available in the Hook Package of a Maintenance Form are:

- 1) Skip Handler : Pr_Skip_Handler This can be used to skip the logic written in another release. *Example: logic written in KERNEL release can be skipped in CLUSTER release*
- 2) Fn_post_bulid_type_structure If any change has to be made in the field values obtained from the form befor start of processing, code can be written here
- 3) Fn_pre_check_mandatory
- 4) Fn_post_check_mandatory

Any extra mandatory checks on the field values from the screen can be written here.

- 5) Fn_pre_query
- 6) Fn_post_query

Any specific logic while querying can be written in these functions. It is called from fn_query of the main package

- 7) Fn_pre_upload_db
- 8) Fn_post_upload_dbAny logic while uploading data to tables can be written here .

9) Fn_pre_default_and_validate

10) Fn_post_default_and_validate

Any release specific logic for defaulting and validation can be written here . It is called from the fn_default_and_validate in the main package

7.2.2 Flow of control through Hook packages

The flow of control through the Hook Packages for a particular stage is as explained in the figure below

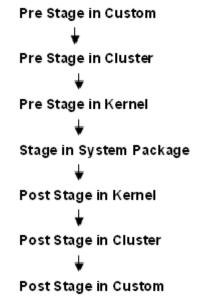
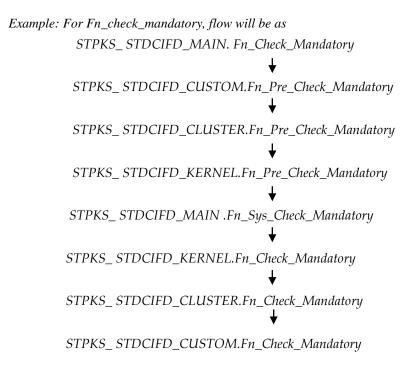


Fig 12.31: Flow of control through Hook Packages



7.2.3 By passing Base Release Functionality

There are auto generated functions like FN_SKIP_<RELEAE_TYPE> which would determine whether or not a particular hooks needs to be called.

Developer also has an option to bypass the base release hook if need be. For example if the validations written in *STPKS_STDCINF_KERNEL.FN_PRE_CHECK_MANDATORY* are not required or not suitable for the Cluster release, system provides an option to bypass the code written by Kernel team. Similarly a Custom release can also bypass the code written by Kernel and Custom Releases. This can be achieved by calling procedures

PR_SET_SKIP_<RELEASE_TYPE> and *PR_SET_ACTIVATE_<RELEASETYPE>*. These procedures will be made available in the main package and the development teams of Customization teams can use these procedures to skip and re-activate the hooks of parent release.

The Developer should avoid adding validations or Checks in the Pre Stage of any function, like Fn_Pre_Check_Mandatory, etc and should aim to add all the validations in the Fn_Post_Default_and_Validate.

For Example let us see the flow for the Mandatory Stage for STDCIFD:

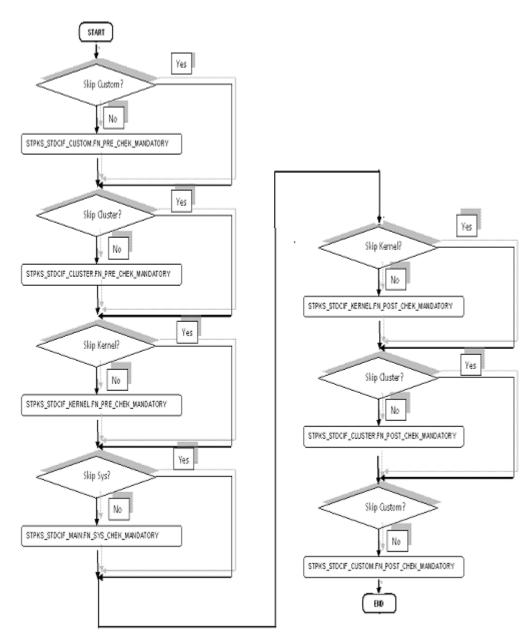


Fig 12.31: Flow of control explaining skip logic in pacakges



Development of Maintenance Form [November] [2020] Version 14.4.0.2.0

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, 2020, 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.