Oracle FLEXCUBE Universal Banking ® 12.87.05.0.0 Development of Online Forms

June 2018



Contents

1. Prefa	ace	3
1.1	Audience	3
1.2	Related Documents	3
2. Intro	oduction	4
2.1	How to use this Guide	4
3. Over	rview of Online Form	4
4. Scree	en Development	5
4.1	Header Information	5
4.2	Preferences	5
4.3	Data Sources	
4.4	Data Blocks	8
4.5	Screens	
4.6	Field Sets	
4.7	Actions	
4.8	Launch Forms	
4.9	Call Forms	
4.9.1	Sub System Pickup/Processing	15
4.10	Summary	16
4.11	Preview	18
5. Gene	erated Units	19
5.1	Front End Units	19
5.1.1	Language xml	19
5.1.2	2 SYS JavaScript File	20
5.1.3	Release Type Specific JavaScript File	20
5.2	Data Base Units	20
5.2.1		
5.2.2	-	
5.2.3		
5.3	Other Units	
5.3 5.3.1		
6. Exte	nsible Development	
6.1	Extensibility in JavaScript Coding	
6.2	Extensibility in Backend Coding	22

1. Preface

This document describes the features of Online Forms in FLEXCUBE and the process of designing a Online form screen 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. 04-Development_WorkBench FLEXCUBE Screen Development _Screen_Development-I.docx Self Acquired Working knowledge of Web based applications Oracle Documentations Working knowledge of Oracle Database Working knowledge of PLSQL & SQL Self Acquired Language 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>
- <u>Chapter 3 , "Overview of Online Form</u>"
- <u>Chapter 4</u>, "Screen Development"
- Chapter 5 , "Generated Units"
- <u>Chapter 5</u>, "Extensible Development"

3. Overview of Online Form

Online Forms are function Id's (screens) which is used for creating Contracts for respective modules. Same contracts can be processed further for Payments, Availments, Amendments, Reassignments and Authorizations also using Online forms.

All the transaction processing in FLEXCUBE is carried out through Online screens Online form screens should be launched independently.

Example: Letter Of Credit (LC) contract

An LC contract is an instruction wherein a customer requests the bank to issue, advice or confirm a letter of credit, for a trade transaction. An LC substitutes a bank's name and credit for that of the parties involved. The bank thus undertakes to pay the seller/beneficiary even if the remitter fails to pay.

Thus for each module we should develop different function Id's for creating contracts and others online forms for other operations like Payments, Availments, Amendments, Reassignments and Authorizations.

On launching the Online form screen, user has to input the respective values to create the contract. Form may have the different user-defined actions like Product-Default, Enrich, and Subsystem-Pickup while creating contract. Once all the user-defined actions performed finally user has to save the contract.

4. Screen Development

Design and development of a Online Form function id is similar to any other function Ids. This section briefs the steps in designing a Online Form screen.

For detailed explanation, refer the document: *04-Development_WorkBench _Screen_Development-I.docx*

4.1 Header Information

Provide the header information as shown in the figure.

Function	Generation					-	×
				×	V	9	4
	Action New Function Id LCDTRONL Save XML Path	Function Type Parent Parent Function Parent Xml	Function Category Transaction				
Search							4
D D D S S F I A C C L	eferences staSource stOValues ataBlocks reens eldSets tions sliForms unchForms ummary		Form has des Information				

Fig 4.1 Online Form header Information

Note the following while providing header information.

i) Name of the Online form :

Online Form name has to have the third character as 'D'. Ideally, the length of the name should be 8 characters.

Example: LCDTRONL, BCDTRONL etc are valid online form names

- ii) Online Form Category: Function Category has to be **Transaction**
- iii) Footer Template:

For Transaction screens, footer template has to be selected as **NONE**. System does not provide any default template for transaction screens; hence developer has to design the footer portion of the screen manually. Developer has to make sure that footer designed has generic fields like transaction status (TXNSTAT), authorization status(AUTHSTAT) etc

For Online Process Flow Screens footer template should be selected as **PROCESS**. Function Type :

Parent and Child functionality is supported for Online forms.

4.2 Preferences

iv)

Provide the menu details in the Preferences screen

Action Load -		Function Type	Parant -	1	Function Category Transaction			
Function Id LCDTRONL		Parent Function]	Header Template None -			
Save XML Path LCDTRONL_F	BROWSE	Parent Xmi			Footer Template None			
		Farent Ani			Pooler remplate None			
earch	Preferences							Ľ,
in Preferences		Head Office Funct	ion	Module	LC 🗾	٦		
DataSource		🗖 Logging Required	1	Module Description	Letters Of Credit			
⊞ 🚞 ListOfValues ⊛ 🚞 DataBlocks		Auto Authorization		Branch Program Id				
B Creens		Tank Modification:	S	Process Code]		
FieldSets		Field Log Require	d	SVN Repository URL				
in Actions CallForms		Multi Branch Acce	ss	Transaction Block Name	BLK_CONTRACT_DETAILS -			
LaunchForms		📄 Excel Export Requ	ired	Transaction Field	BRANCH			
🚞 Summary				Name				
					Co	ntrol Stri	ng 🕂 -	-
	E F	Inction Id		Module *	Module Description			*
	LCDTRONL		LC	×I	Letters Of Credit			
	LCSTRONL		LC		Letters Of Credit			

Fig 4.2 Online Form Preferences

Note the following while providing Preferences for Online Forms.

i) Module name :

Module name is a mandatory field and has to be provided. It is recommended that the first two letters of the function id is kept as same as the module name. Naming of the generated package will be derived from the module code maintained

- *ii)* Script for the following tables will be generated by Workbench (menu details) which are essential for launching of an Online screen.
 - 1. SMTB_MENU
 - 2. SMTB_FCC_FCJ_MAPPING
 - 3. SMTB_FUNCTION_DESCRIPTION
 - 4. SMTB_ROLE_DETAILS

Type string of the Onlne screens will be generated as 'O' in *smtb_menu* table.

iii) Transaction specific action codes has to checked in the control string whichever applicable

Example: LIQUIDATE, ROLLOVER, REVERSAL etc

4.3 Data Sources

Identify the tables/views for the Online form. Define data sources and add data source fields as required.

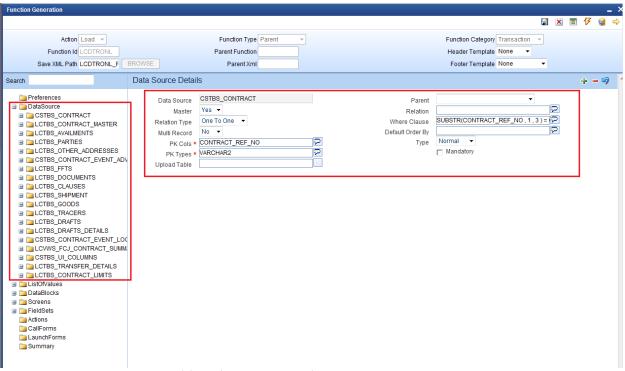


Fig 4.3 Adding data sources and maintaining properties

Note the following while creating data sources

- i) Master Data Source has to be a single entry data source.
- ii) Logical Relationships has to be maintained for all data sources except the parent
- iii) Provide PK Cols and PK types for all data sources.
 If data source is a multi record block, then make sure it has at least one more pk than its parent which helps to uniquely identify each record of multi record block
- iv) *Minimize the use of views in the data sources*. For transaction screens, system generated upload logic (fn_sys_upload_db) is not called within the system package. It is up to the developer to decide whether the system generated code can be used or not. *If views are used in data sources, then this function should not be used by the developer*.
- v) Usually for Online forms, a separate view can be used for summary purpose. This view will have all the fields required to be displayed in the summary. *Example: LCVWS_FCJ_CONTRACT_SUMMARY*

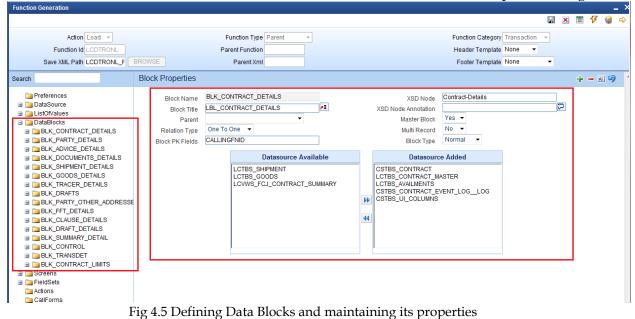
Function Generation		_ ×
		🖫 🗵 🗐 7 🤘 🔿
Action Load 👻	Function Type Parent	Function Category Transaction 👻
Function Id LCDTRONL	Parent Function	Header Template None 👻
Save XML Path LCDTRONL_F	BROWSE Parent Xml	Footer Template None 💌
Search	Data Source Field Details	Refresh 🗕 🗐 🤺
Preferences DataSource CSTBS_CONTRACT BOOK_DATE BRANCH CONTRACT_REF_NO CONTRACT_REF_NO CONTRACT_STATUS EXTERNAL_REF_NO LATEST_EVENT_SEQ_NO LATEST_VERSION_NO LATEST_VERSION_NO LATEST_VERSION_NO PRODUCT_TYPE SOURCE STOP_DATE USER_DEFINED_STATUS USER_REF_NO CURR_EVENT_CODE LATES_CONTRACT_MASTER USELCTBS_CONTRACT_MASTER LATES_CONTRACT_MASTES LATES_CONTRACT_EVENT_A COMPACT_CONE LATES_CONTR	Column Name AUTH_STATUS Block Name Field Name	Data Type CHAR MaxLength 1 Upload Table Column Not Required in Upload Tables

Fig 4.4 Adding data sources fields and its properties

Max length of the data source field can be modified as per requirement

4.4 Data Blocks

Determine the block structure for the function id .Define Data Blocks as per the design



Note the following while creating data blocks

- i) Master Data block has to be a single entry data source.
- ii) Provide Xsd node name if the block is normal and is required in gateway request

- iii) Block order and block field order can be changed by re arranging blocks and block fields in the browser tree (drag and drop).**Note that all units will have to be regenerated if block or block field order is changed** (including xsd's)
- iv) Related currency fields should be placed above the amount field in the tree

Function Generation	1	_ >
		🔚 🛛 🗏 🗗 🚱 🔿
Action Load 👻	Function Type Parent Function	tion Category Transaction 👻
Function Id LCDTRONL	Parent Function Hea	der Template None 👻
Save XML Path LCDTRONL_F	BROWSE Parent Xml For	oter Template None 👻
Search	Block Field Properties	- 🛚 🗖 🦻
Deferences	Field Name * PRTYTYP XSD Tag PRTYTYP	Required
🗃 🚞 DataSource	Field Label LBL_PRTYTYP	Visible
ListOfValues	DataSource LCTBS_PARTIES Field Size * 3	Read Only
DataBlocks BLK_CONTRACT_DETAILS	Column Name * PARTY_TYPE Maximum Length 3	Calender Text
BLK_PARTY_DETAILS	Data Type * Varchar2 v Minimum Value	
CONTREFN05	Display Type Lov Maximum Value	Popup Edit Required
PRTYTYP	Having Designed	Uppercase Only
PARTYDESC		LOV Validation Required
PARTYCIFID	Parent Field	Input by LOV Only
CUSTNAME	Related Block	Not Required In Xsd
CUSTADDLIN1	Related Field	
CUSTADDLIN3	LOV Name LOV_PART_TYPE Mask Id	Report Parameter
CUSTADDLIN4	Off Line LOV Name	
COUNTRYCD	Fieldset Name FST_CONTRACT_PARTIES	
CUSTREFNO	Custom Attributes Events Bind Variables Return Fields Related Field	
CUSTREFDATE	Return Fields Mapping	Default From Lov Definition
ISSBANK	Query Column Block Name	Return Field Name
🚞 ESN	PARTY_TYPE BLK_PARTY_DETAILS -	PRTYTYP -
TEMPLATE_ID		
BLK_ADVICE_DETAILS BLK_DOCUMENTS_DETAILS	ITEM_VAL_DESC BLK_PARTY_DETAILS	PARTYDESC -
BLK_DOCOMENTS_DETAILS		
BLK GOODS DETAILS		
BLK_TRACER_DETAILS		
BLK_DRAFTS		
BLK_PARTY_OTHER_ADDRE		
BLK_FFT_DETAILS		
BLK_CLAUSE_DETAILS BLK_DRAFT_DETAILS		
		-

Add block fields to the data block as required.

Fig 4.6 Attaching Block Fields and maintaining its properties

Note the following while attaching block fields to data blocks

- i) In case the field is not required in XSD, check not Required XSD
- ii) Ensure that Related Block and Field are given for Amount Fields

iii) Minimize the use of query data sources by using DESC fields wherever possible. Note: Query data sources is rarely required for a Online Form screen; as launch form can be used for query only screens

iv) Master block should contain reserved field names like TXNSTAT, AUTHSTAT and SUBSYSSTAT(this is not shown) as shown in the figure .These are reserved field names which are essential for an online form. These will be used by FLEXCUBE Infra while processing. Normally TXNSTAT and AUTHSTAT are added as part of the footer of the screen

COLUMN NAME	BLOCK FIELD NAME
CONTRACT_STATUS	TXNSTAT
AUTH_STATUS	AUTHSTAT
SUBSYSTEM_STAT	SUBSYSSTAT

inction	Generation												
motion	Seneration										×	= 7	
										un			9
	Action Load 👻				Function Type	Parent .	-	Functi	on Category Tr	ansaction 👻			
	Function Id LCDTROM	IL			Parent Function			Head	er Template No	one 👻			
	Save XML Path LCDTRON	IL E	BROWSE		Parent Xml			Foot	er Template No	one 🔻			
arch	REVOLVE		Block H	Field Proper	ties							- 🗷 📮	2 🗉
		-		Field Name *	CONSTAT		100 T	CONSTAT			Requ	ired	
							XSD Tag	CONSTAN					
				Field Label	LBL_CONSTAT	<u>~=</u>	XSD Annotation		~	,	Visibl		
	FREQ			DataSource	CSTBS_CONTRACT		Field Size *	1		Г	Read	Only	
	INEXTREINDT		С	olumn Name *	CONTRACT_STATUS		Maximum Length	1		г	- Caler	nder Text	
	ALLOWREPAY			Data Type *	Char -		Minimum Value					p Edit Req	awir
	CLOSTYP				Text -		Maximum Value						
	TRANSBLE			Display Type			Maximum Decimals			ſ	- Uppe	rcase Only	y
	MAYCONFIRM			Item Type	Database Item 👻					F		/alidation	
	REMARK			Parent Field		-	TextArea Rows				Requ		
	RELLCREF		F	Related Block		•	TextArea Columns			ſ	Input	by LOV On	nly
	BTN_DEFAULT			Related Field	•		Default Value		×=	Г	Not R	equired In	a X:
	CONREFNOLOG						Preview Value			r	Repo	rt Paramet	eter
	MAKER			LOV Name		· ·	MaskId		# E	1			
	MAKDTTIME		Off Lin	e LOV Name		•							
	CHKR		Fi	eldset Name	FST_PRODDET								
	CHKDTTIME		Queter	Attributes Ev	ents Related Field								
- I	TXNSTAT		Custon	Aundules	ents Related Field								
- 1	CONSTAT											+ -	
[authstat				ttribute Name		Attribute Value	A				~	
	BTN_NEXT			А	undute name		Attribute value	Active		Position			
	BTN_PREVIOUS	E											
	availesn												
	BTN_CHARGES												
	BTN_SETT												
	BTN_TAX		5										
	BTN_COLLATERAL												
	BTN_EVENTS												
	BTN_LINKAGES												
	BTN_UDF												
	🚞 BTN_MIS												
	🚞 OF												
	VERSIONLBL											*	
	BACKTOBACK												
	ACKNREVCD												
	ACKDT												

4.5 Screens

Design the screen layout based on the requirement

							× = ·	V 🧐
Action Load 👻		Function Type Parer	nt 👻		Function Category	Transaction 👻		
Function Id LCDTRONL		Parent Function			Header Template	None 🔻		
Save XML Path LCDTRONL_F	BROWSE	Parent Xml			Footer Template	None 👻		
arch	Screen Details						- X	1 🗖 🖬
Preferences DataSource ListOtValues DataBlocks Screens CVS_MAIN HEADER	Screen Name Screen Title Screen Size Exit Button Type	CVS_MAIN LBL_CONTRACT_DETAILS Large Default Cancel	v v	I Main Sc I Visible	reen			
							+	
	Argument N	lame Source Block	Source Field	Argument Value	Target Block	Target Field	Active	~
SEC_TOL	CONTREF		_		BLK_CONTRACT_DETAILS -	CONREFNO -	Yes 👻	
SEC_CUST	ESN				BLK_CONTRACT_DETAILS -	LATEVNSEQNO -	Yes 🔻	
 								_
								Ŧ

_ ×

Fig 4.7 Designing Screens and providing Screen Properties

Note the following while creating screens

• One Screen should be identified as the main screen.

ction Generation			-
Action Load -	Function Type Parent		Function Category Transaction
Function Id LCDTRONL	Parent Function		Header Template None -
Save XML Path LCDTRONL_F	VSE Parent Xml		Footer Template None -
rch Ta	o Details		Dependent Fields 🕂 🛥 🖬 🌍
Preferences DataSource ListOfValues DataBlocks Greens	Screen Name CVS_MAIN Tab Name TAB_MAIN Tab Label LBL_MAIN Tab Type Data	🔽 Visible	
GUS, MAN HEADER TAB MAN TAB MAN SEC_MAIN SEC_TOL SEC_CUST			
TAB_PREFERENCES TAB_PARTIES TAB_PARTIES TAB_PARTIES_LIMIT TAB_PARTIES_LIMIT TAB_TARTIES_LIMIT TAB_TRACERS TAB_TRACERS			
FieldSets Actions CallForms LaunchForms Summary			

Add Tabs, sections and partitions as per the screen design

Fig 4.8 Creating Tabs and maintaining Properties

Note the following when creating tabs and sections for the screen

- i) If the screen does not have multiple tabs, then only the TAB_MAIN needs to be used. TAB_HEADER should not contain any sections in this scenario
- Normally Online forms are large screens with multiple tabs. In this case, all the tabs needs to be used .TAB_HEADER should contain the header information.
 TAB_MAIN should be the first tab in the body .Other tabs has to be added in the body portion as required
- *iii)* Footers are often designed by the developer for Online forms. Provide sections in TAB_FOOTER as required. *Note that in large screens ,footer supports 4 partitions while other portions support 3 partitions*

					E 77 🧐
Action Load -		Function Type Parent	*	Function Category Transaction 👻	
Function Id LCDTRONL		Parent Function		Header Template None 👻	
Save XML Path LCDTRONL_F	BROWSE	Parent Xml		Footer Template None 🔹	
arch	Section Details				- 🕅 🗳
Preferences DataSource ListONalues DataBlocks Sceens	Section Name Section Label	SEC_TOL	Visible		
🖃 🚞 CVS_MAIN	Partition Details	5			+ -
	Partition SI N	0	Partition Name	Width Sub-partition	IS ^
🖃 🚞 TAB_MAIN	1	PART_TOL1		66 🔻 2 👻	
SEC_MAIN	2	PART_TOL2		33 🔻 👻	
SEC_TOL		_			
SEC_CUST					~
SEC_STAT					Ŧ
SEC_STAT					Ŧ
SEC_STAT TAB_PREFERENCES TAB_PARTIES					*
SEC_STAT					*
TAB_PREFERENCES					×
□ SEC_STAT □ TAB_PREFERENCES □ TAB_PARTIES □ TAB_PARTIES_LIMIT □ TAB_SHIPMENT					Ŧ
BEC_STAT DEC_ST					Ŧ
SEC_STAT					Ŧ
SEC_STAT					Ŧ
SEC_STAT SEC_STAT TAB_PREFERENCES TAB_PARTIES TAB_SHIPMENT TAB_SHIPMENT TAB_SHIPMENT TAB_TRACERS TAB_TRACERS FOOTER FOOTER FOOTER GOVER					v
SEC_STAT					T
SEC_STAT B TAB_PREFERENCES TAB_PARTIES TAB_PARTIES TAB_SHIPMENT TAB_SHIPMENT TAB_TRACERS TAB_TRACERS TAB_ADVOCES TAB_ADVOCES CVS_DRAFT CVS_DRAFT CVS_DRAFT CVS_TRANSDET FieldSets					
SEC_STAT					
SEC_STAT SEC_STAT SAD_PARTIES TAB_PARTIES TAB_PARTIES_LIMIT TAB_SHIPMENT TAB_TAB_COCUMENTS TAB_TARCERS TAB_ADVICES CVS_DRAFT CVS_DRAFT CVS_DRAFT CVS_TRANSDET FieldSets Actions CaliForms					
SEC_STAT					

Multiple Screens can be designed if required.

4.6 Field Sets

Create Field sets and attach the fields to the field sets as required

		-	🖫 🗶 🗏 7 🧐 🖨
Action Load -	Function Type Parent	Function Category Transaction]
Function Id LCDTRONL	Parent Function	Header Template None 🔻	
Save XML Path LCDTRONL_F BROW	Parent Xml	Footer Template None	•
Search Fiel	dset Properties		- 🛛 🌍
Screens FieldSets FST_PROD FST_PROD FST_REF FST_REF FST_CREPT FST_CREPT FST_CREPT FST_CREPT FST_CREPT FST_CREPT FST_PREF FST_PREF FST_PREF FST_PREF FST_DRAVEE_DETAILS FST_DRAWEE_DETAILS FST_LORAVEE_DETAILS FST_ST_SHOWENT FST_ST_SHOWENT FST_FORT FST_SNEWNENT FST_FORTE, 1	Fieldset Name FST_PROD Fieldset Label PL Data Block BLK_CONTRACT_DETAILS ~ Mutit Record No ~ View Type Single ~ Fieldset Height Data Block Fields BTN_DRAFTS BTN_COMMISSION PRTYTYP CROLIN LINECID INCCAMDNO EVENTCD BTN_CHARGES BTN_COLLATERAL BTN_COLLATERAL BTN_COLLATERAL BTN_CULATERAL BTN_FUNTS BTN_LINKAGES	Screen Name CVS_MAIN Screen Portion Header Tab Name TAB_HEADER Section Name SEC_HEADER2 Partition Name PAR11 Number Of Rows Field Set Fields Subpartition Name PRDCD BTN_P SCALLINGFNID CALLINGFNID CALLINGFNID COMPARENT COMPARENT	 ✓ Horizontal Fieldset ReadOnly Navigation Button ✓ Visible
FST_CONTRACT_PARTIES FST_CONTRACT_ADVICES FST_CONTRACT_DOCUMENT FST_CONTRACT_TRACERS FST_CONTRACT_OTHER_ADI FST_CONTRACT_OTHER_ADI FST_CONTRACT_CLAUSES FST_CONTRACT_CLAUSES FST_CONTRAT_DRAFTS_DET FST_CONTRAL_DRAFTS_DET FST_CONTRAL FST_CONTRAL FST_CAURENTEE FST_GAURENTEE FST_GAURENTEE FST_GAURENTEE			

_ ×

Fig 4.9 Field Set Properties

Note the following when attaching field to a field set

i) If a field is not required in the screen, but kept as hidden and value defaulted; then **The field has to be made invisible and attached to a field set**. If it is not attached to any fields set, the screen html won't contain the field and may result in script error while accessing the field.

4.7 Actions

Mention the web service and amendable information in Actions Screen

Action Load 👻 Function Type Parent 👻				F	unction Category Transacti	on 👻	
Function Id LCDTRONL		Parent Function				•	
Save XML Path LCDTRONL	F BROWSE	Parent Xml			Footer Template None	•	
rch	Form Actions						G
Preferences	XSD Type Iden	tifier Contract		Service Name FCU	BSLCService	×1	
DataSource ListOfValues DataBlocks Screens	Operatio	on Id Contract					•
i FieldSets	Web Service	Action Code	Opera	ation Code	Action Stage Type	Amendables	
CallForms		QUERY	QueryContract			Amendables	
iaunch-orms ia Summary		NEW	CreateContract			Amendables	
		MODIFY	ModifyContract			Amendables	
		AUTHORIZE	AuthorizeContract			Amendables	
		DELETE	DeleteContract			Amendables	
		CLOSE	CloseContract			Amendables	
		REOPEN	ReopenContract		V	Amendables	
		REVERSE	ReverseContract			Amendables	
		ROLLOVER	RolloverContract		V	Amendables	
		CONFIRM				Amendables	
		LIQUIDATE				Amendables	
		SUMMARYQUERY					

Fig 4.10 Actions Screen

Note the following while maintaining web services and amendable information

- i) Online forms will generate Type XSD and Message XSD.
 - Operation specific message xsd's will be generated.

Example: for the example given in the figure, name of the xsd generated will be LC-Contract-Types.xsd (Type XSD for LC Contract) LC-CreateContract-Req-Full-MSG.xsd (Create Message XSD for LC Contract) LC-CreateContract-Req-IO-MSG.xsd (Create Message XSD for LC Contract) LC-CreateContract-Res-Full-MSG.xsd (Create Message XSD for LC Contract) LC-CreateContract-Res-Full-MSG.xsd (Create Message XSD for LC Contract)

ii) Operation Id and Operation Code need be maintained for the above mentioned reason

iii) Amendable information has to be maintained similar to any other function ids.

4.8 Launch Forms

Launch Forms can be attached to Online form screen.

Function Id LCDTRONL Parent Function Header Template None Save XML Path LCDTRONL, F BROWSE Parent Xml Footer Template None earch Launch Form Details □ DataSource □ DataSource □ DataBlocks □ DataBlocks □ DataBlocks □ Screen S Function ID Active	unts the
arch Launch Form Details Preferences DataSource ListON/alues DataBlocks Function ID	vonte
Preferences DataSource ListOfValues DataBlocks ListOfValues ListOfVal	vonte d
DataSource ListOrValues DataBlocks Function ID Active	unte L
ListOfValues Screen Argum DataBlocks Function ID Active	onte 🕂 🗖
	iento T
□ FleidSets CSDEVENT Yes ▼	
Actions MSDALMSG Yes V	
LaunchForms Ves Yes	

Screen Arguments should be maintained for the launch form to query the proper contract record from the main online functions.

Function Generation								
								🗏 7 🥑 🧇
Action Load +		Functio	n Type Parent 👻		Fund	ton Calegory Tran	saction 👻	
Function Id LCDTRON		Parent Fi	unction		Head	ter Templale Non	0 👻	
Save XML Palh LCDTRONI	L_F BROWSE	Pari	ant Xmil		Foo	ler Template	•	
Search	Launc	ch Form Details						6)
Preferences								
DalaSource								
 ListOfValues DataBlocks 							Screen Arguma	nts 🛧 🗕
 Screens 				Function ID			Active	^
 FieldSets Actions 		CSDEVENT					Yes 💌	
CaliForms		MSDALMSG					Yes *	
LaunchForms		MSDMSPRV					Yes 👻	
🚞 Summary		Call Form Arguments				×		
					Populate Reset			
		Argument Name	Source Block	Source Field	Argument Value			
		CONTREF	BLK_CONTRACT_DET	AILS - CONREFNO -				
		ACTION_CODE		•	EXECUTEQUERY			
					Ok Cancel			
								Ŧ

Process to attach launch forms is similar to any other function Id's.

4.9 Call Forms

Call forms can be attached to Online form. Each call form should be mapped to Parent Data Block, Parent Data Source and proper relations should be maintained with parent data source of main online form.

Action Load		Function Type Parent		Function Category Transaction	7
Function Id LCDTRONL		Parent Function		Header Template None 👻	
Save XML Path LCDTRO		Parent Xml		Footer Template	•
arch	Call Form Details				
Preferences DataSource					
i 🔄 ListOfValues				Screen Arguments Dep	pendent Fields + -
∎ 🛅 DataBlocks ∎ 🛅 Screens	Function ID	Parent Data Block	Parent DataSource	Relation	Relation Type
FieldSets	CFCTRCOM	BLK_CONTRACT_DETAILS	▼ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
Actions	CFCTRCHG	BLK_CONTRACT_DETAILS	✓ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
CallForms	ISCTRSTL	BLK_CONTRACT_DETAILS	✓ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
🚞 Summary	LCCTRCLT	BLK_CONTRACT_DETAILS	▼ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	TACTRTAX	BLK_CONTRACT_DETAILS	▼ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	CSCTRLNK	BLK_CONTRACT_DETAILS	✓ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	CSCTRUDF	BLK_CONTRACT_DETAILS	✓ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To Many 👻
	MICTRMIS	BLK_CONTRACT_DETAILS	✓ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	LCCBCLNK	BLK_CONTRACT_DETAILS	✓ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	CSCTRSPT	BLK_CONTRACT_DETAILS	▼ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	BCCTRPRF	BLK_CONTRACT_DETAILS	▼ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	CSCOFACT	BLK_CONTRACT_DETAILS	✓ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	BCCBRDET	BLK_CONTRACT_DETAILS	✓ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻
	CSCDOCTR	BLK_CONTRACT_DETAILS	▼ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 👻
	LCCILUTL	BLK_CONTRACT_DETAILS	▼ LCTBS_CONTRACT_MASTER ▼	LCTBS_CONTRACT_MASTER.CONT	One To One 🔻

Sreen Arguments should be given to each callform. So that the call form will display the respective data of calling main function.

Dependant Fields are required to re default the call form values when the user changes input data in the main form.

Each of the subsytem pickup logic will have to be coded by the developer in release specific packages. Processing logic (sub system pickup) for the attached call forms has to be called from the main form package.

4.9.1 Sub System Pickup/Processing

Subsystem pickup refers to the process of picking up the values in sub systems. Normally values in sub systems will be defaulted based on the data given in the main screen of the online form .

i) Defaulting of sub system

After providing values in the main screen ,user may click on any sub system to view or change the value.

On clicking the sub system for the first time ,sub system values will be defaulted based on the values provided in the main screen . Action code passed will be **SUBSYSPKP** .

The code for defaulting will have to written by the developer in corresponding hook packages in function *Fn_Post_Subsys_Pickup*

In this case SUBSYSSTAT for all subsystems will go as 'D' and processing done based on this flag for each sub system (call form). Note that SUBSYSPKP action will default values for all subsystems and not only the sub system being launched Example:

MICTRMIS:D;ISCTRSTL:D;TACTRTAX:D;CSCTRUDF:D;CFCTROCH:D;CSCTRADV:D; FTCCGCLM:D;

If user saves the contract without visiting any call forms, then all the subs systems will be defaulted before saving

ii) Uploading of sub system

If after launching the subsystem with defaulted values; User changes the value in subsystem; the new user input values has to be uploaded to the system. Hence while saving , *the subsystems which has been modified by user will be uploaded while others will be defaulted*.

In this case SUBSYSSTAT for the subsystem which has been modified will go as 'U' .Developer has to write code for processing based on the flag

Example: if user changes MIS details (MICTRMIS) from what was defaulted; then SUBSYSSTAT will go as

MICTRMIS:U;ISCTRSTL:D;TACTRTAX:D;CSCTRUDF:D;CFCTROCH:D;CSCTR ADV:D;FTCCGCLM:D;

iii) Re defaulting of sub system

After launching and changing subsystem values; if user changes any values in main screen which are dependent field for the subsystem : subsystem values will have to be defaulted again based on the new main screen values . Hence the sub system will be re defaulted. In this case value entered by the user in susb system will be lost .

In this case SUBSYSSTAT for the subsystem whose dependent fields has been modified will go as 'R'. .Developer has to write code for processing based on the flag

Example: In a Funds Transfer Contract Input Screen, assume that charge subsystem(CFCTROCH) is dependent on the values entered for debit and credit account. After launching the sub system and changing the charges manually; if user changes the account again the charges will have to re defaulted. The manully entered charges will not be considered. SUBSYSSTAT will go as

MIČTRMIS:U;ISCTRSTL:D;TACTRTAX:D;CSCTRUDF:D;CFCTROCH:R;CSCTR ADV:D;FTCCGCLM:D;

Values for other subsystems will depend on each of their dependencies .

4.10 Summary

Summary screens can be designed for Online Form if required

								×	E 17	۶ 🧯
Action Load -		Function Type Parent 👻			Function	Category Transact	on 👻			
Function Id LCDTRONL		Parent Function			Header	Template None	•			
Save XML Path LCDTRONL_F	BROWSE	Parent Xml			Footer	Template	•			
earch	Summary Details									۵
 Preferences DataSource ListOValues DataBlocks Screens FieldSets Actions CallForms 	Title Data Blocks Data Source Summary Type Summary Screen Size	LBL_SUMMARY BLK_SUMMARY_DETAIL • LCVWS_FCJ_CONTRACT_SUN • Summary • Medium •	*I	Default Where Clause Default Order By Multi Branch Where Clause Main Summary Screen	MODULE_C MODULE_C	ODE = "LC" AND EXI	STS (SEL			
aunchForms	Data Block Fields C	ustom Buttons Fields Ordering								
🚞 Summary					-					
Summary		Data Block Fields		Fields Selected	Query	LOV Name		•		
Summary Summary	CLOSDT SETLMTH	Data Block Fields		UTHSTAT	v	LOV Name	•			
Summary	SETLMTH EFFDT	Data Block Fields	_	UTHSTAT ONSTAT	<u>।</u> ज	LOV Name	• •	* E		
Summary	SETLMTH EFFDT CIFID MAXCONT	AMT		UTHSTAT ONSTAT ONREFNO	प प प प	LOV Name	•	•		
Summary	SETLMTH EFFDT CIFID MAXCONT MAXLIABAI CURRAVA	AMT AT		UTHSTAT ONSTAT ONREFNO RDCD	े ज्य य	LOV Name	* * *			
Summary	SETLMTH EFFDT CIFID MAXCONT MAXLIABAI	AMT AT	A	UTHSTAT ONSTAT ONREFNO RDCD ONTCCY	प प प प प प	LOV Name	* * * *			
Summary	SETLMTH EFFDT CIFID MAXCONT MAXLIABAI CURRAVA	AMT AT		UTHSTAT ONSTAT ONREFNO RDCD ONTCCY ONTAMT	प प प प प प प प	LOV Name	* * *			
Summary S	SETLMTH EFFDT CIFID MAXCONT MAXLIABAI CURRAVA	AMT AT		UTHSTAT ONSTAT ONREFNO RDCD ONTCCY	प प प प प प	LOV Name	* * * *			

4.11 Preview

The figure shows the preview of the Online form Input screen developed

🖹 New 🦻 Enter Quen	/					
Product Code *	P	Contract Reference		Operation Code	*	
Product Description		User Reference		Source Code	FLEXCUBE	
		Source Reference		Version Number ∢	Of	1
Product Type	.					1
Main Preferences Part	ies Parties Limits Shipment	Documents Tracers Advice	ces			
LC Details						
Currency *		Customer *		Issue Date		
Contract Amount *		Customer Name		Effective Date		
Positive Tolerance		Party Type *		Tenor		
Negative Tolerance		Dated		Expiry Date		
Max Amount		Customer Reference		Expiry Place		
Liability Tolerance		License Expiry Date			Auto Closure	
Liability Amount		Amount		Closure Date		
Tolerance Text	· ·	Liability		Stop Date		
				Pre-Advice Date		
						E
				Reference To Pre- advice		
Credit				- Guarantee Details		
Туре		Credit Available With		Type of guarante		
Mode	· · · · · · · · · · · · · · · · · · ·	Details		Guarante	e 👻	
		Details				
Revolving Detail						
Revolves in			Automatic Reinstateme	ent Remarks		
Units			Cumulative	internality	Default	
		Next Reinstatement	Cumulauve		Loan for Collateral	
Frequency		Date			Partial Closure	
-Reimbursement Undert	aking					
Undertaking Expiry		Availed Undertaking				
Date		Amount				_
Undertaking Amount						
Status						
	Charges Settlement				talls BC Linkages	
Split Settlement Loa	n Preference Brokerage	All Messages Docume	nts Message Preview	Import License		
Maker	Da	ate Time	Status			
Checker		ata Tima	Authorization Status			Exit
	Da	ate Time				

The figure shows the preview of the Online form Summary screen developed

		d Search 🏼 🎝 Res					
	Authorization Status	-		Contract St	atus	•	
	Contract Reference		20	Product C	Code	<u> </u>	
	Currency	<u>×</u> Ξ		Contract Am	ount	x =	
	Branch		x	Operation C	Code	•	
eco	rds per page 15 👻 📊	▲ 1 of 1	Go to Page				
	Authorization Status	Contract Status	Contract Reference	Product Code	Currency	Contract Amount	User Refe
_		III					Þ
							Exit

Fig 4.10 Online Form Summary Screen Preview

Generate the units for Online form and deploy them in the FLEXCUBE server for unit testing.

5. Generated Units

The following units will be generated for a Online Form screen. Refer document on generated units on detailed explanation on the same

5.1 Front End Units

5.1.1 Language xml

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

Example – **LCDTRONL.xml** (uixml for LC Contract Screen)

5.1.2 SYS JavaScript File

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

Example – LCDTRONL_SYS.js (JS for LC Contract Screen)

5.1.3 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

Example – LCDTRONL_KERNEL.js (JS for KERNEL Release) *Example* – LCDTRONL_CLUSTER.js (JS for CLUSTER Release) *Example* – LCDTRONL_CUSTOM.js (JS for CUSTOM Release)

5.2 Data Base Units

5.2.1 Static Scripts

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

5.2.2 System Packages

Main package would be generated by the Tool and should not be modified by the developer.

Example – Lcpks_Lcdtronl_Main.spc, Lcpks_Lcdtronl_Main.sql (Main Package for LC Contract)

Main package contains functions for :

- Converting Ts to PL/SQL Composite Type
- Calling fn_main.
- Resolve Ref Numbers (fn_resolve_ref_numbers)
- Mandatory checks (fn_check_mandatory).
- Product Default (fn_product_default)
- Subsystem Pickup(fn_subsys_pickup)
- Enriching (fn_enrich)
- Default and validation(fn_default_and_validate)
- Uploading into DB tables(fn_upload_db)
- Processing the contract input values(fn_process)
- Querying(fn_query)
- Converting the Modified Composite Type again to TS

Except the functions for type conversions, others functions calls the respective hook functions in hook packages of the Online forms. Thus no processing logic within the main package is used

But the package contains many other system generated functions for operations like

- Mandatory checks(fn_sys_check_mandatory)
- Default and validation(fn_sys_default_and_validate)
- Uploading to DB(fn_sys_upload_db)
- Query operation (fn_sys_query) etc

These functions are not called anywhere in the package. These functions if required can be called by the developer from the release specific package. Otherwise developer can write his own logic for the same in the Hook Packages

5.2.3 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.

<i>Example</i> – Lcpks_Lcdtronl_Kernel.spc,	Lcpks_Lcdtronl_Kernel.sql (Kernel Package)
Lcpks_Lcdtronl_Cluster.spc,	Lcpks_Lcdtronl_Cluster.sql (Cluster Package)
Lcpks_Lcdtronl_Custom.spc,	Lcpks_Lcdtronl_Custom.sql (Custom Package)

5.3 Other Units

5.3.1 Xsd

Only Type XSD and message XSD will be generated for a Online Form function Id. This type xsd will be used in the type xsd of any function which uses the particular Online form.

Example - LC-Contract-Types.xsd (Type XSD for LC Contract)

LC-CreateContract-Req-Full-MSG.xsd (Create Message XSD for LC Contract) LC-CreateContract-Req-IO-MSG.xsd (Create Message XSD for LC Contract) LC-CreateContract-Res-Full-MSG.xsd (Create Message XSD for LC Contract) LC-CreateContract-Res-PK-MSG.xsd (Create Message XSD for LC Contract)

6. Extensible Development

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

6.1 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 LCDTRONL_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 ()*

6.2 Extensibility in Backend Coding

For online forms, generated code does not provide any business logic . Insert statements won't be present as part of generated code in online packages. Developer has to write the business logic in release specific packages (or make call to server functions from release specific packages).

Hooks will be provided in the following stages

- Resolving reference numbers
- Checking mandatory fields
- Defaulting and validating
- Uploading to db
- Process
- Subsystem pickup
- Enrich
- Product Default
- Query

Note that the system generated code for uploading; defaulting etc

(*fn_sys_default_and_validate,fn_sys_upload_db etc*) won't be called by the main package in online flow. If it is required, developer has to call it explicitly from release specific packages.

Note that in online flow, upload to base tables happens first and processing is done on the inserted data after uploading. After processing , the response type will be build

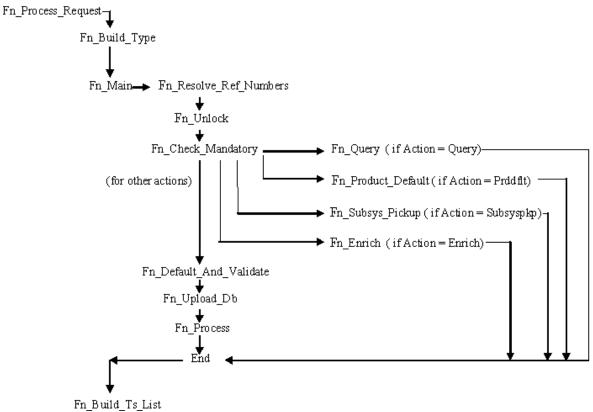


Fig 4.10 Flow of control in an Online main package

Release specific code has to be written in the Hook Packages generated. Different functions available in the Hook Package of a Online 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 Main

This is called form the fn_main in main package.

3) Fn_pre_resolve_ref_numbers

4) Fn_post_resolve_ref_numbers

This function validates the reference number. It is called from fn_ resolve_ref_numbers of the main package

5) Fn_pre_unlock

6) Fn_post_unlock

This function holds the contract level validations and modification logic for existing contract. It is called from fn_unlock of main package.

- 7) Fn_pre_check_mandatory
- 8) Fn_post_check_mandatory

Any mandatory checks can be validated here. It is called from fn_chchk_mandatory of main package.

9) Fn_pre_query

10) Fn_post_query

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

11) Fn_pre_product_default

12) Fn_post_product_default

This function has the logic to default the values for the contract based on the product maintenance. It is called from fn_product_default of main package.

13) Fn_pre_subsys_pickup

14) Fn_post_subsys_pickup

This function does the subsystem pickup for the subsystem's (call form's) as per product maintenance for the contract. It is called from fn_subsys_pickup of main package.

15) Fn_pre_enrich

16) Fn_post_enrich

After product default, user can default others values. That logic can be put here. It is called from fn_enrich of main package.

17) Fn_pre_default_and_validate

18) 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.

19) Fn_pre_upload_db

20) Fn_post_upload_db

Any logic while uploading data to tables can be written here. It is called from fn_upload_db of main package.

21) Fn_pre_process

22) Fn_post_process

These hook functions are specific to transaction online form screens. This function should have the call to all the server functions which process the input data for the contract as per the functionality. These are called from fn_process of the main package.



Development of Online Forms June 2018

Oracle Corporation World Headquarters 500 Oracle Parkway Redwood Shores, CA 94065 U.S.A.

Worldwide Inquiries: Phone: +1.650.506.7000 Fax: +1.650.506.7200 www.oracle.com/ financial_services/

Copyright © 2017-2018 Oracle Financial Services Software Limited. All rights reserved.

No part of this work may be reproduced, stored in a retrieval system, adopted or transmitted in any form or by any means, electronic, mechanical, photographic, graphic, optic recording or otherwise, translated in any language or computer language, without the prior written permission of Oracle Financial Services Software Limited.

Due care has been taken to make this document *Development of Online Forms* and accompanying software package as accurate as possible. However, Oracle Financial Services Software Limited makes no representation or warranties with respect to the contents hereof and shall not be responsible for any loss or damage caused to the user by the direct or indirect use of this *Development Of Online Forms* and the accompanying Software System. Furthermore, Oracle Financial Services Software Limited reserves the right to alter, modify or otherwise change in any manner the content hereof, without obligation of Oracle Financial Services Software Limited to notify any person of such revision or changes.

All company and product names are trademarks of the respective companies with which they are associated.