Oracle® Banking Enterprise Limits and Collateral Management
Child and Screen Childs - Concept and Design

ORACLE®

Contents

1

Preface

- **Purpose**
- **Audience**
- **Documentation Accessibility**
- **Critical Patches**
- **Diversity and Inclusion**
- **Basic Actions**
- **Related Documents**
- **Conventions**
- **Screenshot Disclaimer**
- **Acronyms and Abbreviations**
  The list of the acronyms and abbreviations used in this guide are as follows:
- **Symbols and Icons**
  The lists of symbols, buttons and shortcut key that are used in the application to perform various tasks are covered in this topic.
- **Prerequisite**

1.1 Purpose

This guide is designed to help acquaint you with the Oracle Banking Enterprise Limits and Collateral Management (ELCM) application. This guide provides answers to specific features and procedures that the user need to be aware of the module to function successfully.

User can further obtain information specific to a particular field by placing the cursor on the relevant field and pressing <F1> on the keyboard.

1.2 Audience

This guide is intended for the following User/User Roles:

Table 1-1   Audience

Role	Function
Back office data entry clerk	Input functions for funds
Back office managers/officers	Authorization functions
Product Managers	Product definition and authorization
End of day operators	Processing during end of day / beginning of day

1.3 Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

1.4 Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at Critical Patches, Security Alerts and Bulletins. All critical patches should be applied in a timely manner to ensure effective security, as strongly recommended by Oracle Software Security Assurance.

1.5 Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1.6 Basic Actions

Table 1-2   Basic Actions

Action	Description
Approve	Used to approve the initiated report. This button is displayed, once the user click **Authorize**.
Audit	Used to view the maker details, checker details, and report status.
Authorize	Used to authorize the report created. A maker of the screen is not allowed to authorize the report. Only a checker can authorize a report, created by a maker.
Close	Used to close a record. This action is available only when a record is created.
Confirm	Used to confirm the performed action.
Cancel	Used to cancel the performed action.
Compare	Used to view the comparison through the field values of old record and the current record. This button is displayed in the widget, once the user click **Authorize**.
Collapse All	Used to hide the details in the sections. This button is displayed, once the user click **Compare**.
Expand All	Used to expand and view all the details in the sections. This button is displayed, once the user click **Compare**.
New	Used to add a new record. When the user click **New**, the system displays a new record enabling to specify the required data.

ORACLE®

Table 1-2   (Cont.) Basic Actions

Action	Description
OK	Used to confirm the details in the screen.
Save	Used to save the details entered or selected in the screen.
View	Used to view the report details in a particular modification stage. This button is displayed in the widget, once the user click **Authorize**.
View Difference only	Used to view a comparison through the field element values of old record and the current record, which has undergone changes. This button is displayed, once the user click **Compare**.
Unlock	Used to update the details of an existing record. System displays an existing record in editable mode.

1.7 Related Documents

For more information refer to the Oracle Banking manuals on:

- Development of Launch Forms and Others Screens

- Enterprise Collaterals User Guide

- Enterprise Limits and Collaterals Common User Guide

1.8 Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1.9 Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

1.10 Acronyms and Abbreviations

The list of the acronyms and abbreviations used in this guide are as follows:

Table 1-3   Acronyms and Abbreviations

Acronyms	Abbreviations
CIF	Customer Information File

Table 1-3   (Cont.) Acronyms and Abbreviations

Acronyms	Abbreviations
CASA	Current Account and Savings Account
DDA	System that holds the CASA account and balances
ELCM	Enterprise Limits and Collateral Management
ECA	External Credit Approval
FCUBS	Oracle FLEXCUBE Universal Banking Solution
GW	Gateway
HTTP	Hyper Text Transfer Protocol
ID	Identification Number
Mark EOTI	Mark End of Transaction Input
Mark TI	Mark Transaction Input
OFSAA	Oracle Financial Services Analytical Applications
ORMD	Oracle Revenue and Billing Management
PK	Primary Key
RDBMS	Relational Data Base Management System
SMS	Security Services
UI	User Interface
VD	Value Date
XML	Extensible Mark-up Language
XSD	XML Schema Definition
XSLT	Extensible Stylesheet Language Transformations

1.11 Symbols and Icons

The lists of symbols, buttons and shortcut key that are used in the application to perform various tasks are covered in this topic.

Table 1-4   Symbols and Icons

Icons	Function
🔍	Perform search
⌐ ⌐	Minimize
▶	Navigate to the next record
◀	Navigate to the previous record
(toggle)	Toggle OFF
(toggle)	Toggle ON
✕	Delete
+	Click this icon to add a new row.

Table 1-4   (Cont.) Symbols and Icons

Icons	Function
—	Click this icon to delete an existing row.
	List view
	Maximize
	Navigate to the first record
	Navigate to the last record
	Advance search
	Search record
	Save the record
	Reset the record
	Clear the record

Table 1-5   Symbols and Icons - Audit Details

Icons	Function
	A user
	Branch details
	Date and Time

1.12 Prerequisite

Specify the **User ID** and **Password**, and login to **Home** screen.

2

Child Screen

This topic provides an overview of the Child screen in the Open Development Tool.

Screens are required in Oracle FLEXCUBE Universal Banking where the base functionality is the same and the differences are either in the layout or in some additional processing that is there are some sets of screens where the majority of the functionality is the same with some variance with existing screens.

Development Workbench supports developing such screens as the child screens of the base function with a facility to upgrade the child screens whenever the base or parent screen changes.

For example, **Term Deposit Account Booking**, account for term deposit booking will have all the features of a normal customer account with some additional features. Thus it can be designed as the child screen of normal customer account maintenance.

Workbench provides an option to design a child screen as a derivative of a parent screen. Workbench automatically inherits the parent screen and tracks the modifications made in the child screen. Workbench does not allow the developer to delete elements inherited from the parent screen.

For child screens, in addition to release-specific hooks for each logical stage, the system also calls the corresponding hook from the parent function's programs. This allows the reuse of the common business logic among all the child functions. Oracle FLEXCUBE Universal Banking does not support multi-level inheritance (that is the child of the child is not supported).

Screen Development

Refer to the topic #unique_25 for the screen development.

Generation of Files

The process of generation of files is similar to that of normal function ID. All the units generated for a normal function ID will be generated for the child screen as well. In the script for **SMTB_MENU**, column **PARENT_FUNCTION** would contain the name of the parent function ID.

Extensible Development

The developer can add code in hook packages and release specific JavaScript files. The extensible development process is similar to that followed in a maintenance or transaction screen. Refer to respective documents for a detailed explanation. The structure of the system packages will be the same as for a normal maintenance or transaction function ID.

For child screens, in addition to release-specific hooks for each logical stage, the system also calls the corresponding hook from the parent function's programs. This allows the reuse of the common business logic among all the child functions.

For example, assume **STDCUSAC** function ID is a normal maintenance screen and **STDCUSTD** function ID is the child of this screen. The below figure shows the snippet of code from **fn_default_and_validate** of the child main package.

Figure 2-1   stpks_stdcustd_main.sql

```
IF NOT stpks_stdcustd_Main.Fn_Skip_Master THEN
    Pr_Convert_Child_To_Master(p_stdcustd,l_STDCUSAC);
    Pr_Convert_Child_To_Master(p_Prev_stdcustd,l_Prev_STDCUSAC);
    Pr_Convert_Child_To_Master(p_Wrk_stdcustd,l_Wrk_STDCUSAC);
    Dbg('Calling stpks_stdcusac_Kernel.Fn_Pre_Default_And_Validate ');
    IF NOT stpks_stdcusac_Kernel.Fn_Pre_Default_And_Validate (p_Source,
        p_Source_Operation,
        p_Function_Id,
        p_Action_Code,
        p_function_Id  ,
        l_STDCUSAC,
        l_Prev_STDCUSAC,
        l_Wrk_STDCUSAC,
        p_Err_Code,
        p_Err_Params) THEN
        Dbg('Failed in stpks_stdcusac_Kernel.Fn_Pre_Default_And_Validate ');
        RETURN FALSE;
    END IF;
    Pr_Convert_Master_To_Child(l_Prev_STDCUSAC,p_Prev_stdcustd);
    Pr_Convert_Master_To_Child(l_Wrk_STDCUSAC,p_Wrk_stdcustd);
END IF;
```

The developer has written defaulting and validation for the parent screen being called before doing validations specific to the child screen. Hence business logic for the parent is reused. The developer has the option to skip doing parent validation for the child screens using skip handlers if required.

Note the functions **Pr_convert_child_to_master** and **Pr_convert_master_to_child**. These procedures have to be used every time a call is made to the parent function id packages. This procedure prepares object types for the parent and child respectively.

- Develop New Child screen
  This topic explains systematic instructions to develop a new Child screen.

2.1 Develop New Child screen
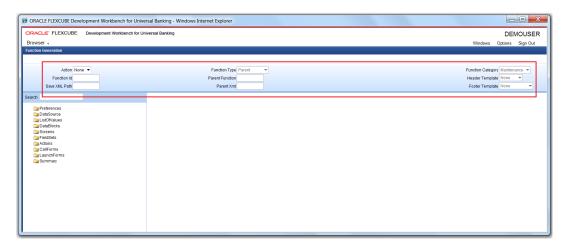
This topic explains systematic instructions to develop a new Child screen.

Specify the **User ID** and **Password** in the Development Workbench Universal Banking Login Page and log in to the **Homescreen**.

1. On **Expand Menu**, click **Function Generation** node.

   The **Function Generation** screen displays.

Figure 2-2   Function Generation

2. On the **Function Generation** screen, specify the fields in the **Header** section.
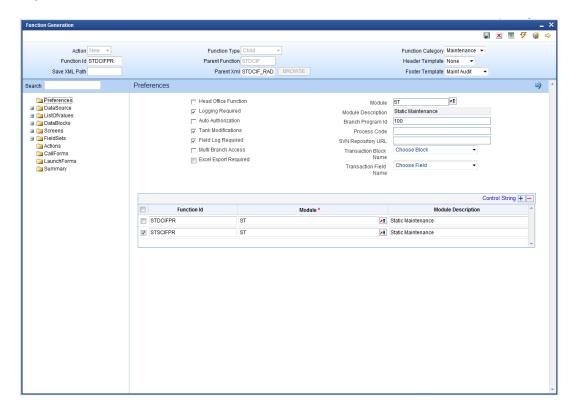
Table 2-1   Function Generation - Field Description

Field	Description
Action	Select **New** from the drop-down list.
Function Type	Select **Child** from the drop-down list.
Parent XML	On entering details for fields **Action** and **Function Type**, the **Parent XML** field is enabled. Click **BROWSE** and select the required parent function from the hard disk. On successful loading, the parent function will get updated with the parent function ID value. Parent function can be either a parent function ID or a child function ID.
Parent Function	The **Parent Function** field defaulted to the name of the function id that has loaded as the parent.
Function Category	The **Function Category** field defaulted with the category of the parent. It cannot be modified.
Function ID	Enter Child function ID in the **Function ID** field. Apply standard naming conventions.
Header Template	Select the header template from the drop-down list: • **None** • **Process**
Save XML Path	Specify the XML path.
Footer Template	Select the footer template from the drop-down list. • **None** • **Maint Audit** • **Maint Process** • **Process**

3. On the **Function Generation** screen, click **Save**.

4. On the **Function Generation** screen, click **Generate**.

    The **Preferences** screen displays.

Figure 2-3   Preferences

In the **Preferences** screen, the function ID name must be of the child screen, and not of the parent screen. On the tab out of the **Preferences** node, the function name gets changed automatically if not changed manually. Hence developer has to visit the **Preferences** node at least once before generating files for the child function ID.

The Child screen design is similar to normal screen design. Refer to the Oracle FLEXCUBE Enterprise Limits and Collateral Management ODT Screen Development document for the detailed information.

Note the following while designing a Child screen:

- The developer will not be allowed to delete or rename elements created in the parent function.

- New data sources or adding columns to existing data sources are allowed.

- The addition of new LOVs and modification of existing LOVs are allowed.

- Modifying properties of existing block fields(of the parent) is allowed.

- New Screens, tabs, sections, etc can be added by the developer.

- Deletion and Renaming of screens, tabs, sections, etc are not allowed. Instead, the developer has the option to hide them.

- Addition or removal of fields from fieldset is allowed. Properties of fields can also be modified. Note that the block to which the fieldset is attached cannot be changed at the child level.

- New fieldsets can be defined.

- The developer can add a new Launch Form and Call Form in the child screen. The existing launch Form/Call form can be made inactive, if not required at the child level.

- Amendable fields can be modified.
- The summary screen can also be completely redesigned at the child level.

> **Note:**
>
> If enhancement on the parent screen happens parallel to the child screen design developer can use the child refresh option to upgrade the latest changes in the parent screen to the child screen.

3

Screen Child

This topic provides an overview of the Screen Child.

Screen child can be derived from a parent screen or a child screen. Screen child uses the packages for maintenance functions from the parent itself, only screen-level modifications can be done in a Screen Child.

The Screen Child is not exactly the child of child screens (second-level inheritance). It can be the child of a parent or a child screen. The difference between a child and a screen child is that only screen-level changes can be done in a screen Child. Business logic will remain the same as its parent. In the child screen business logic can also be enhanced for the child function ID.

Screen Development
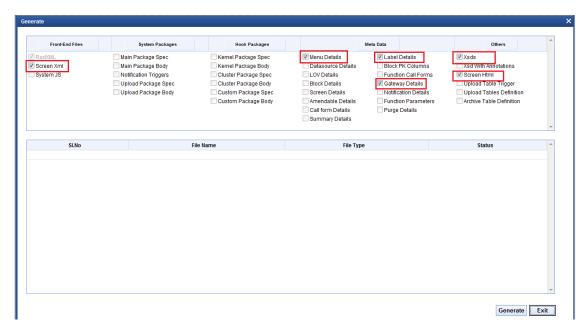
Refer to the topic #unique_28 for the screen development.

Generated Files

Only Screen XML (language-specific XML), Menu Details, Label Details, XML Schema Definitions, and Screen HTML will be generated from the Workbench for Screen Child Screen.

Figure 3-1   Generate

Extensible Development

The screen child uses the packages and JavaScript files of the parent itself.

* Develop New Screen Child
  This topic explains systematic instructions to develop a new Screen Child screen.

3.1 Develop New Screen Child
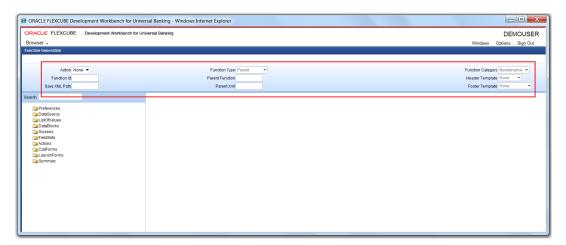
This topic explains systematic instructions to develop a new Screen Child screen.

Specify the **User ID** and **Password** in the Development Workbench Universal Banking Login Page and log in to the **Homescreen**.

1. On **Expand Menu**, click **Function Generation** node.

   The **Function Generation** screen displays.

   Figure 3-2   Function Generation

2. On the **Function Generation** screen, specify the fields in the **Header** section.

   Table 3-1   Function Generation - Field Description

Field	Description
Action	Select **New** from the drop-down list.
Function Type	Select **Screen Child** from the drop-down list.
Parent XML	On entering details for fields **Action** and **Function Type**, the **Parent XML** field is enabled. Click **BROWSE** and select the required parent function from the hard disk. On successful loading, the parent function will get updated with the parent function ID value. Parent function can be either a parent function ID or a child function ID.
Parent Function	The **Parent Function** field defaulted to the name of the function ID that has loaded as the parent function.
Function Category	The **Function Category** field defaulted with the category of the parent. It cannot be modified.
Function ID	Enter Screen Child function ID in the **Function ID** field. Apply standard naming conventions.

Table 3-1   (Cont.) Function Generation - Field Description

Field	Description
Header Template	Select the header template from the drop-down list: • **None** • **Process**
Save XML Path	Specify the XML path.
Footer Template	Select the footer template from the drop-down list. • **None** • **Maint Audit** • **Maint Process** • **Process**

Figure 3-3   Function Type - Screen Child

3. On the **Function Generation** screen, click **Save**.

4. On the **Function Generation** screen, click **Generate**.

   Only 4 nodes of the parent can be modified in a Screen Child as follows:

   • **Data Blocks**

   • **Screens**

   • **Field Sets**

   • **Actions**

Figure 3-4   Screen Child- Generated Screen

Below are the possible operations that can be done on a Screen Child in Oracle FLEXCUBE Universal Banking Development Workbench:

- Existing elements cannot be deleted.

- New blocks or block fields cannot be added.

- Only certain properties of the fields in the block can be modified.

- Properties of screens and fieldsets can be modified.

- New screens and fieldsets can be added.

- Tabs, sections, and partitions can be added to the existing screens.

- Web Service and operation can be configured for the screen child, but amendable fields cannot be modified.

Field-level properties that can be modified are as follow:

- **Field Label**

- **Field Size**

- **Maximum Decimals**

- **Default Value**

- **Preview Value**

- **Visible**

- **Read Only**

- **Calendar Text**

- **Popup Edit Required**

- **Uppercase Only**

- **Input by LOV only**

- **Not Required in XSD**

Screen level modification can be done according to the requirement. Most of the properties at the screen level can be modified. Fieldset level modification also can be done as per requirement. The block to which the field set is attached cannot be changed at Screen Child for existing fieldsets. However, all properties can be modified for new fieldsets.

If parent screen enhancements are happening parallel to the screen, the child design developer can use the Screen Child refresh option to upgrade the latest parent changes to the screen child.
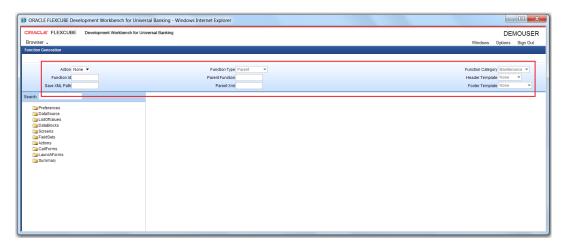
4

Develop New Child screen

This topic explains systematic instructions to develop a new Child screen.

Specify the **User ID** and **Password** in the Development Workbench Universal Banking Login Page and log in to the **Homescreen**.

1. On **Expand Menu**, click **Function Generation** node.

   The **Function Generation** screen displays.

   Figure 4-1   Function Generation

2. On the **Function Generation** screen, specify the fields in the **Header** section.

   Table 4-1   Function Generation - Field Description

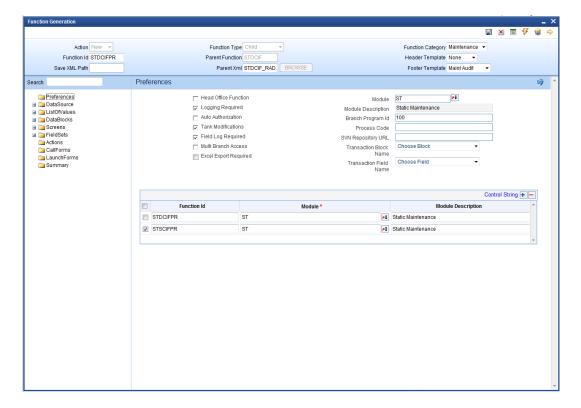 | Field | Description |
   |---|---|
   | **Action** | Select **New** from the drop-down list. |
   | **Function Type** | Select **Child** from the drop-down list. |
   | **Parent XML** | On entering details for fields **Action** and **Function Type**, the **Parent XML** field is enabled. Click **BROWSE** and select the required parent function from the hard disk. On successful loading, the parent function will get updated with the parent function ID value. Parent function can be either a parent function ID or a child function ID. |
   | **Parent Function** | The **Parent Function** field defaulted to the name of the function id that has loaded as the parent. |
   | **Function Category** | The **Function Category** field defaulted with the category of the parent. It cannot be modified. |
   | **Function ID** | Enter Child function ID in the **Function ID** field. Apply standard naming conventions. |

Table 4-1   (Cont.) Function Generation - Field Description

Field	Description
Header Template	Select the header template from the drop-down list: • **None** • **Process**
Save XML Path	Specify the XML path.
Footer Template	Select the footer template from the drop-down list. • **None** • **Maint Audit** • **Maint Process** • **Process**

3. On the **Function Generation** screen, click **Save**.

4. On the **Function Generation** screen, click **Generate**.

   The **Preferences** screen displays.

Figure 4-2   Preferences

In the **Preferences** screen, the function ID name must be of the child screen, and not of the parent screen. On the tab out of the **Preferences** node, the function name gets changed automatically if not changed manually. Hence developer has to visit the **Preferences** node at least once before generating files for the child function ID.

The Child screen design is similar to normal screen design. Refer to the Oracle FLEXCUBE Enterprise Limits and Collateral Management ODT Screen Development document for the detailed information.

Note the following while designing a Child screen:

- The developer will not be allowed to delete or rename elements created in the parent function.

- New data sources or adding columns to existing data sources are allowed.

- The addition of new LOVs and modification of existing LOVs are allowed.

- Modifying properties of existing block fields(of the parent) is allowed.

- New Screens, tabs, sections, etc can be added by the developer.

- Deletion and Renaming of screens, tabs, sections, etc are not allowed. Instead, the developer has the option to hide them.

- Addition or removal of fields from fieldset is allowed. Properties of fields can also be modified. Note that the block to which the fieldset is attached cannot be changed at the child level.

- New fieldsets can be defined.

- The developer can add a new Launch Form and Call Form in the child screen. The existing launch Form/Call form can be made inactive, if not required at the child level.

- Amendable fields can be modified.

- The summary screen can also be completely redesigned at the child level.

> **Note:**
>
> If enhancement on the parent screen happens parallel to the child screen design developer can use the child refresh option to upgrade the latest changes in the parent screen to the child screen.
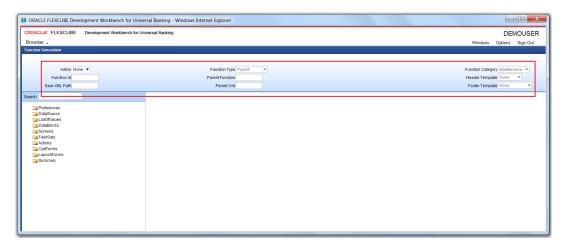
5

Develop New Screen Child

This topic explains systematic instructions to develop a new Screen Child screen.

Specify the **User ID** and **Password** in the Development Workbench Universal Banking Login Page and log in to the **Homescreen**.

1. On **Expand Menu**, click **Function Generation** node.

   The **Function Generation** screen displays.

   Figure 5-1   Function Generation

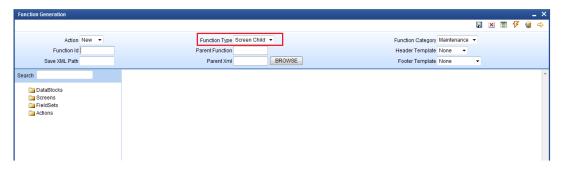2. On the **Function Generation** screen, specify the fields in the **Header** section.

   Table 5-1   Function Generation - Field Description

Field	Description
Action	Select **New** from the drop-down list.
Function Type	Select **Screen Child** from the drop-down list.
Parent XML	On entering details for fields **Action** and **Function Type**, the **Parent XML** field is enabled. Click **BROWSE** and select the required parent function from the hard disk. On successful loading, the parent function will get updated with the parent function ID value. Parent function can be either a parent function ID or a child function ID.
Parent Function	The **Parent Function** field defaulted to the name of the function ID that has loaded as the parent function.
Function Category	The **Function Category** field defaulted with the category of the parent. It cannot be modified.
Function ID	Enter Screen Child function ID in the **Function ID** field. Apply standard naming conventions.

Table 5-1   (Cont.) Function Generation - Field Description

Field	Description
Header Template	Select the header template from the drop-down list: • **None** • **Process**
Save XML Path	Specify the XML path.
Footer Template	Select the footer template from the drop-down list. • **None** • **Maint Audit** • **Maint Process** • **Process**

Figure 5-2   Function Type - Screen Child

3. On the **Function Generation** screen, click **Save**.

4. On the **Function Generation** screen, click **Generate**.

   Only 4 nodes of the parent can be modified in a Screen Child as follows:

   • **Data Blocks**

   • **Screens**

   • **Field Sets**

   • **Actions**

Figure 5-3   Screen Child- Generated Screen

Below are the possible operations that can be done on a Screen Child in Oracle FLEXCUBE Universal Banking Development Workbench:

- Existing elements cannot be deleted.

- New blocks or block fields cannot be added.

- Only certain properties of the fields in the block can be modified.

- Properties of screens and fieldsets can be modified.

- New screens and fieldsets can be added.

- Tabs, sections, and partitions can be added to the existing screens.

- Web Service and operation can be configured for the screen child, but amendable fields cannot be modified.

Field-level properties that can be modified are as follow:

- **Field Label**

- **Field Size**

- **Maximum Decimals**

- **Default Value**

- **Preview Value**

- **Visible**

- **Read Only**

- **Calendar Text**

- **Popup Edit Required**

- **Uppercase Only**

- **Input by LOV only**

- **Not Required in XSD**

Screen level modification can be done according to the requirement. Most of the properties at the screen level can be modified. Fieldset level modification also can be done as per requirement. The block to which the field set is attached cannot be changed at Screen Child for existing fieldsets. However, all properties can be modified for new fieldsets.

If parent screen enhancements are happening parallel to the screen, the child design developer can use the Screen Child refresh option to upgrade the latest parent changes to the screen child.