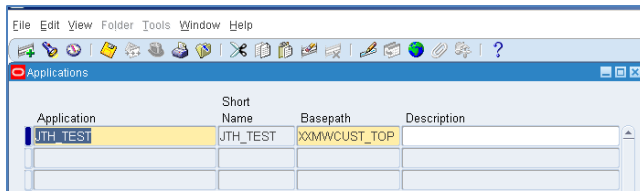


Impact of using custom applications associated with Responsibilities

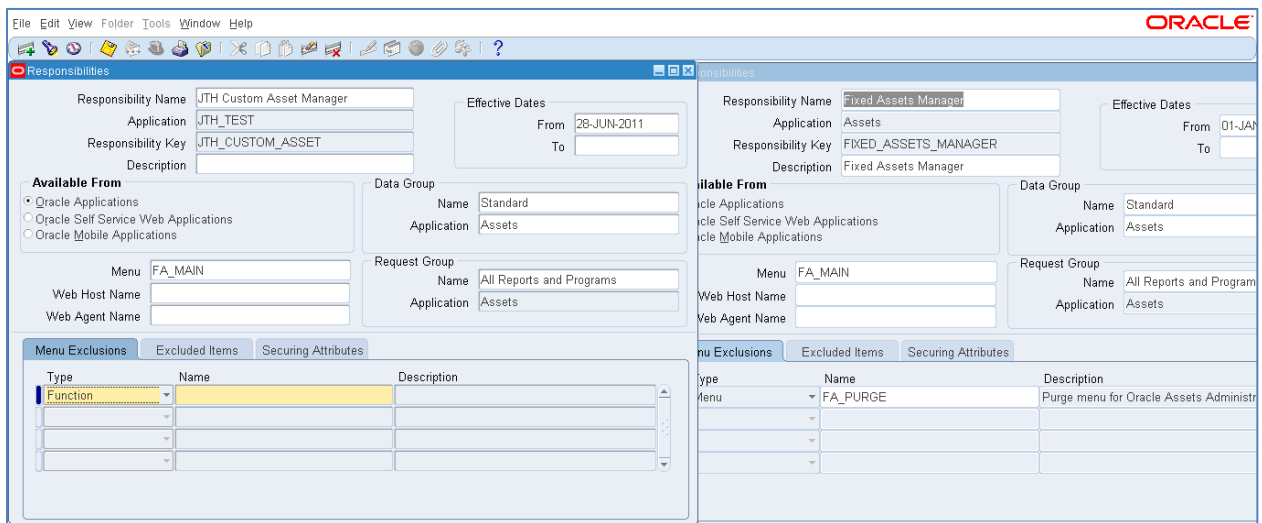
Article:

When designing security, companies some organizations set up a custom “Application” to group customizations and their ‘custom’ responsibilities. Here is an example of a custom application called “JTH_TEST”



Application	Short Name	Basepath	Description
JTH_TEST	JTH_TEST	XXMWCUST_TOP	

Here is an example of a custom responsibility associated with this custom application:



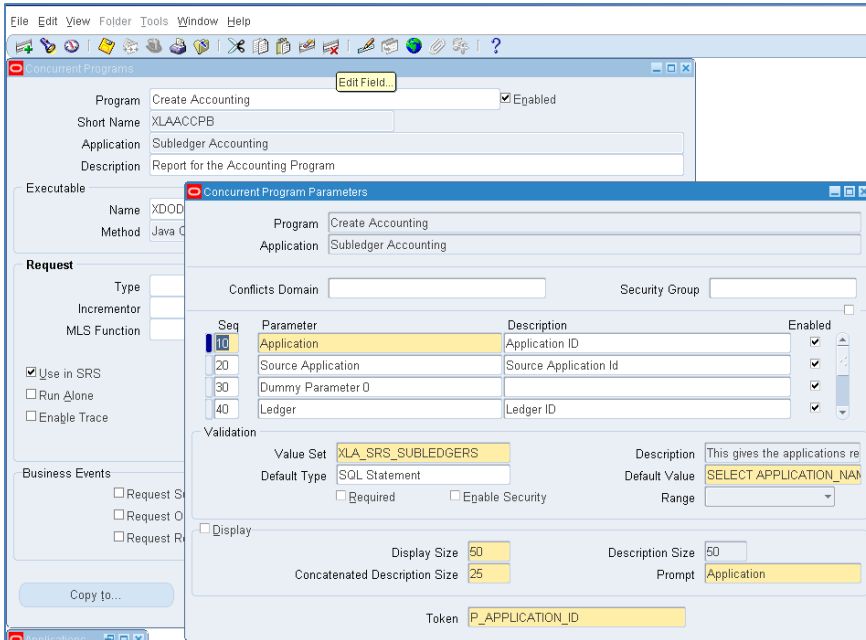
Responsibility Name	Application	Responsibility Key	Description	Effective Dates
JTH Custom Asset Manager	JTH_TEST	JTH_CUSTOM_ASSET		From: 28-JUN-2011
Fixed Assets Manager	Assets	FIXED_ASSETS_MANAGER	Fixed Assets Manager	From: 01-JAN

Available From	Data Group	Request Group	Menu	Web Host Name	Web Agent Name
<input type="radio"/> Oracle Applications <input type="radio"/> Oracle Self Service Web Applications <input type="radio"/> Oracle Mobile Applications	Name: Standard, Application: Assets	Name: All Reports and Programs, Application: Assets	FA_MAIN		

Type	Name	Description
Function		
Menu	FA_PURGE	Purge menu for Oracle Assets Administr

In this example, I have created a custom responsibility “JTH Custom Asset Manager” with the same data group, menu, and request group as the standard responsibility “Fixed Assets Manager”.

In R12, Oracle introduced their Sub-Ledger Architecture (SLA) and many of the programs and reports are standardized across the subledgers. One example is the Create Accounting program. Here is a screen shot of that program’s definition:



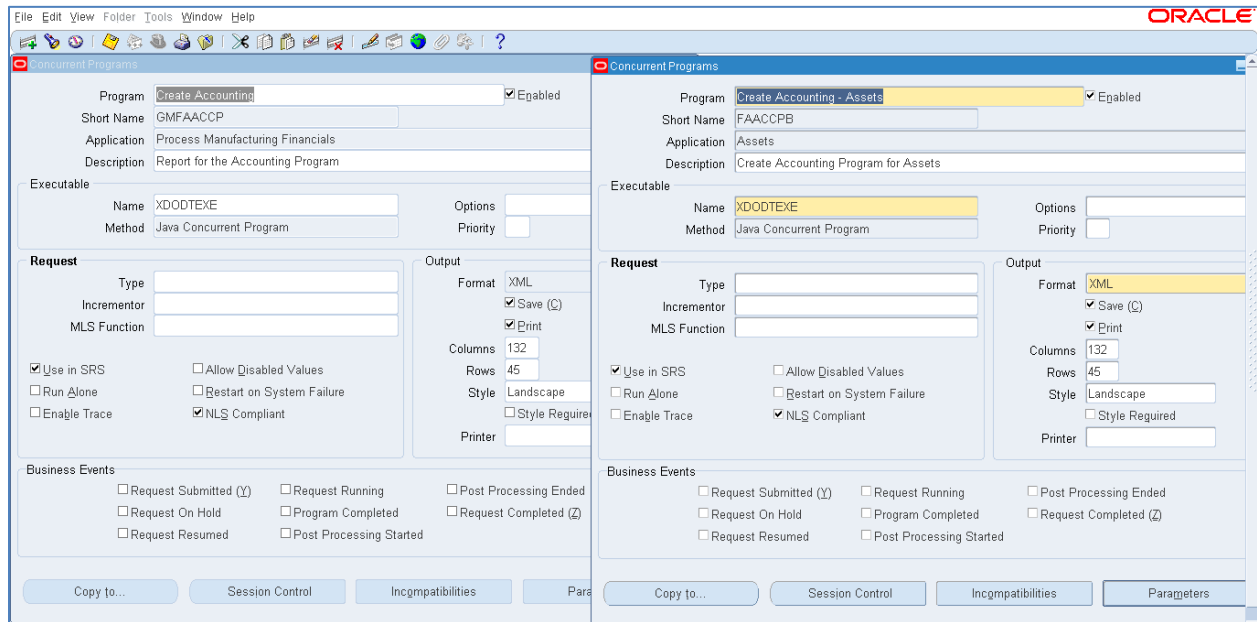
The screenshot shows the 'Concurrent Program Parameters' window in Oracle. The main window is titled 'Concurrent Programs' and shows details for the 'Create Accounting' program. The 'Concurrent Program Parameters' window is open, showing the following details:

- Program:** Create Accounting
- Application:** Subledger Accounting
- Short Name:** XLAACCPB
- Description:** Report for the Accounting Program
- Executable:** Name: XDOD, Method: Java C
- Request:** Type, Incrementor, MLS Function, Use in SRS, Run Alone, Enable Trace, Business Events (Request S, Request O, Request R)
- Parameters Table:**

Seq	Parameter	Description	Enabled
10	Application	Application ID	<input checked="" type="checkbox"/>
20	Source Application	Source Application Id	<input checked="" type="checkbox"/>
30	Dummy Parameter 0		<input checked="" type="checkbox"/>
40	Ledger	Ledger ID	<input checked="" type="checkbox"/>
- Validation:** Value Set: XLA_SRS_SUBLEDGERS, Description: This gives the applications re, Default Type: SQL Statement, Default Value: SELECT APPLICATION_NAM, Range: [dropdown]
- Display:** Display Size: 50, Description Size: 50, Concatenated Description Size: 25, Prompt: Application
- Token:** P_APPLICATION_ID

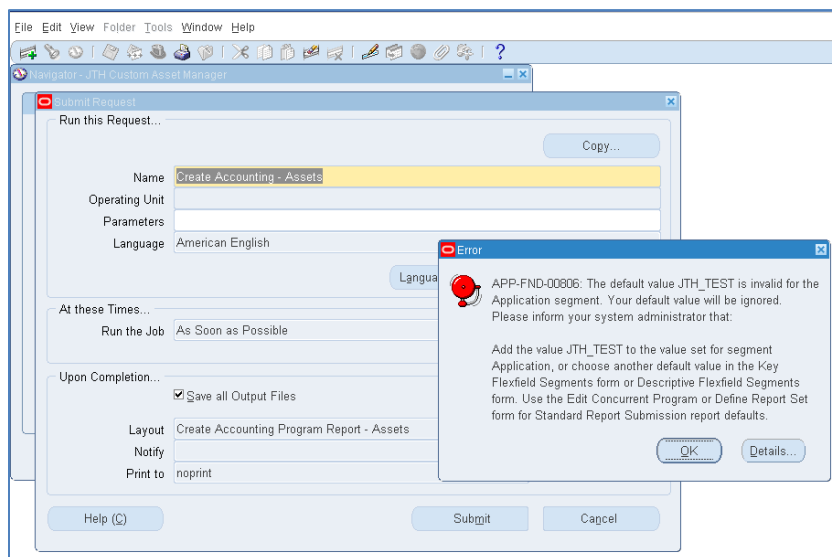
You can see that the “Application” is a parameter this is obtained via this SQL statement (from the Default Value field) “SELECT APPLICATION_NAME FROM FND_APPLICATION_VL WHERE APPLICATION_ID= nvl(:\$PROFILES\$.XLA_APPL_ID:NULL,:\$PROFILES\$.RESP_APPL_ID)”. The application is derived from the Application defined on the Responsibility. So... if the application is “JTH_TEST” as is the case in the “JTH Custom Asset Manager” responsibility above, the program wouldn’t know what data to process. The program needs the application passed from the definition of the responsibility or from another source.

Here are screen shots for Create Accounting and Create Accounting – Assets programs:



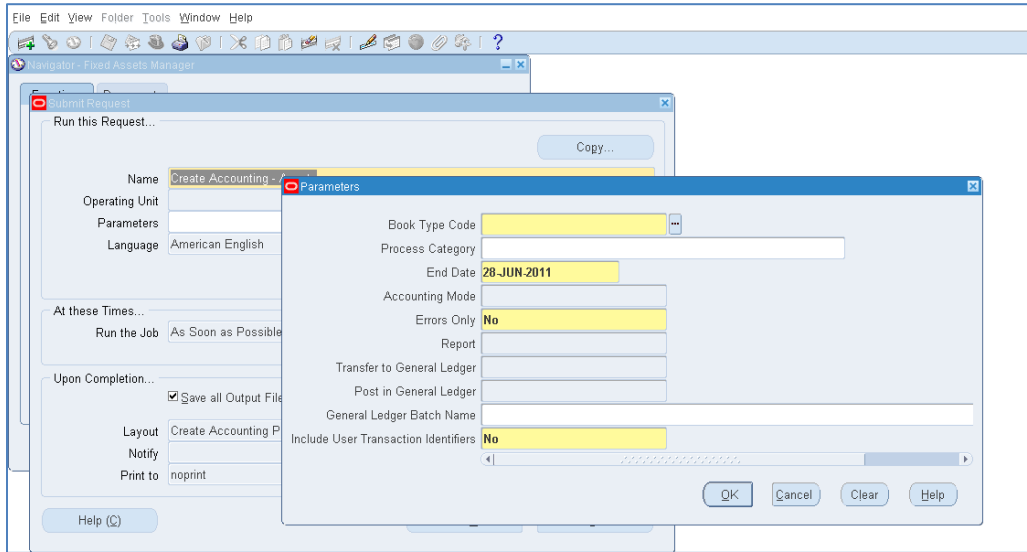
You can see that Oracle has used the same executable “XDODTEXE” (underlying code) for both programs.

Here is an example of what happens when you try to run an SLA program using a responsibility that is associated with a custom application. In this example, I am running another SLA program called “Create Accounting – Assets”

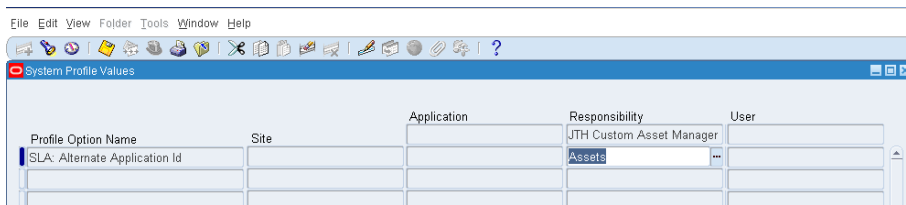


The program cannot be executed because it cannot derive the Application.

However, when running it from the standard “Fixed Asset Manager” responsibility, the program derives the Application (as a hidden parameter) and passes that value as a parameter. Here is a screen shot of the program’s parameters:



To solve this issue, Oracle introduced a new profile option called “SLA: Alternate Application Id” that can be set at the Responsibility level as follows:



This profile option can be used to identify the Application that the Responsibility is associated with so the SLA programs will ‘know’ which data to process.

If you are a company that is upgrading to R12 and you have associated your custom Responsibilities with a custom Application (as opposed to the standard application – e.g. Assets), you will need to set the profile option “SLA: Alternate Application Id” for each of your responsibilities at the Responsibility level as shown above.

The patch number that you may need to apply in order to load the “SLA: Alternate Application Id” is [Patch 8332032](#). Find out more at My Oracle Support using notes 833986.1 or 1233453.1.