

ERP- Oracle Apps

Lesson 18: Customization using Custom.pll

August 14, 2014 Proprietary and Confidential -1-

IGATE
Speed. Agility. Intelligence.

Lesson Objectives

- List the coverage for this lesson
- Custom Library Overview.
- Architecture of the CUSTOM Library.
- Routines in the CUSTOM library.
- Adding packages to the CUSTOM library.
- Coding Restrictions.
- Events passed to the CUSTOM library.
- Examples.
- Zoom LOV.
- Summary.
- Review Question



Custom Library Overview

- An oracle forms allow PL/SQL library.
- Allow the extension of oracle applications without modifying oracle applications code.
- Receives events from forms in oracle applications. Can be used to:
 - Code zoom
 - Enforce business rules.
 - Disable fields.
 - Modify form appearance.

August 14, 2014 Proprietary and Confidential < 3 >

IGATE
Ignited Agility. Inspired Innovation.

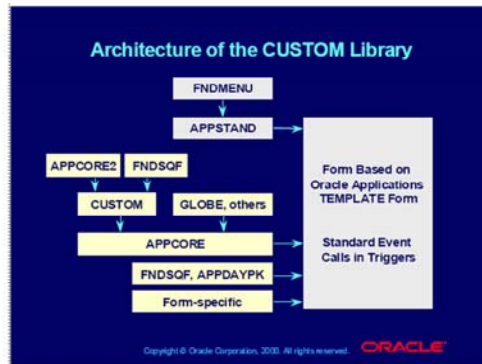
CUSTOM.pll is used to add extensions to Oracle's form Functionality. Some of the common scenarios where CUSTOM.pll can be used are:-

1. Enabling/Disabling the fields
2. Changing the List of Values in a LOV field at runtime
3. Defaulting values
4. Additional record level validations
5. Navigation to other screens
6. Enabling Special Menu

The CUSTOM.pll library is a standard Oracle Forms PL/SQL library that is supplied by Oracle with the Oracle Applications. This is Oracle's built-in feature that allows the customer to enhance the standard functionality of the Applications by implementing site-specific business rules. Every Oracle Forms -based eBusiness screen, and any custom form developed using the Oracle Application development standards, will access the CUSTOM library.

This makes an ideal point of creating business rules that effect the entire organization. This is the only method of forms enhancement whose functionality is supported by Oracle World Wide Support. Although any enhancements coded by the customer are not directly supported by Oracle World Wide Support.

Architecture of the CUSTOM Library



August 14, 2004 Proprietary and Confidential - 8 -

IGATE
Speed. Agility. Intelligence.

All the forms that are based on the oracle applications TEMPLATE from reference objects from the APPSTAND.fmb form .libraries attached directly to the form include APPCORE,APPYPK,FNDSQF and any other form specific libraries.

Since the CUSTOM library is a typical Forms PL/SQL library, it can include standard PL/SQL logic, Oracle Forms built-in commands and may have other Forms PL/SQL libraries attached to it. The base CUSTOM library is located in the \$AU_TOP/resource directory on your forms server. Explore the CUSTOM.pll using the Oracle Forms Designer module to examine the sample code that exists in library. Once any enhancements are made, the library must be generated into an executable library (CUSTOM.plx) that then can be used by the Oracle Applications Forms runtime module. Since the CUSTOM library is loaded once for a given session, a user must log out of the application and sign-on again before any changes will become apparent. It is suggested that you also place a copy of the CUSTOM library in your customization directory as a safety precaution. The CUSTOM PL/SQL library does have its limitations. It can not contain SQL commands. It can not have certain Oracle Applications Forms global libraries attached to it (such as APPCORE). The CUSTOM library is designed to be used solely with the Oracle eBusiness Applications and does not support the Oracle Self-Service Web Applications (OSSWA). Since the CUSTOM library's standard location is in the \$AU_TOP/resource, it can be overwritten during an upgrade of the Applications.

Architecture of the CUSTOM Library

- **CUSTOM is called from events common to all in all oracle application.**
- **Event triggers in the TEMPLATE from call APPCORE routines which in turn call the CUSTOM library.**
- **CUSTOM contains 3 routines:**
 - ZOOM_AVIALABLE
 - STYLE
 - EVENT

August 14, 2014

Proprietary and Confidential

« 5 »

IGATE
Ignite Applications Intelligence

The CUSTOM library allows extension of Oracle Applications without modification of Oracle Applications code. You can use the CUSTOM library for customizations such as Zoom (such as moving to another form and querying up specific records), enforcing business rules (for example, vendor name must be in uppercase letters), and disabling fields that do not apply for your site.

You write code in the CUSTOM library, within the procedure shells that are provided. All logic must branch based on the form and block for which you want it to run. Oracle Applications sends events to the CUSTOM library. Your custom code can take effect based on these events.

Routines in the CUSTOM Library

- **ZOOM_AVIALABLE:**
Code whether zoom is available a particular block on a form
- **STYLE:**
Choose to have your code executable before , after,or in place of the code provided. STYLE is available only for product specific events that support custom execution styles.
- **EVENT:**
Code specific business logic which is based on events occurring in form.

August 14, 2014Proprietary and Confidential< 5 >**IGATE**
Ignite Applications

ZOOM_AVIALABLE

This function allows you to specify if zooms exist for the current context. If zooms are available for this block, then return TRUE else return FALSE. This routine is called on a per-block basis within every Applications form from the WHEN-NEW-BLOCK-INSTANCE trigger. Therefore, any code that will enable Zoom must test the current form and block from which the call is being made. By default this routine must return FALSE.

Appcore2 Library

- **APPCORE cannot be attached to CUSTOM because CUSTOM is attached to APPCORE.**
- **APPCORE2 library duplicates most APPCORE routines in**
 - APP_ITEM_PROPERTY2
 - APP_DATE2
 - APP_SPECIAL2
- **Procedures and functions are the same, just the package names are different (add a 2)**
- **You must mutually attach APPCORE2 to CUSTOM, if you need any of this packages.**

August 14, 2014

Proprietary and Confidential < 2 >


 IGate
 Ignited. Applied. Integrated.

STYLE

This function returns a integer value. This function allows to override the execution style of Product specific events, but it doesn't effect generic events like when-new-form-instance. Possible return values are:

1. custom.before
2. custom.after
3. custom.override
4. custom.standard

By default it returns custom.standard.

EVENT

This procedure allows you to execute your code at specific events including:

- ZOOM
- WHEN-NEW-FORM-INSTANCE
- WHEN-NEW-BLOCK-INSTANCE
- WHEN-NEW-RECORD-INSTANCE
- WHEN-NEW-ITEM-INSTANCE
- WHEN-VALIDATE-RECORD

By default this routine must perform 'null;'

Adding packages to the custom library

- You cannot change the specification of the CUSTOM package in any way.
- You can place your code within the existing package body , or add your own packages.
- You may attach other libraries to the CUSTOM library.
 - FNDSQF already attached.
 - Attach APPCORE2 to instead of APPCORE.

August 14, 2014 Proprietary and Confidential v 8.0

IGATE
Ignite Your Imagination

The CUSTOM library is shipped with the FNDSQF library already attached however, early versions of release 10SC did not include the attachment.

Coding Restrictions

- You cannot use any SQL in the CUSTOM library or any library attached to it.
- Use record groups for SELECT statements, call stored procedures for other DML operations.
- The CUSTOM library is global to all oracle applications.
- There is single instance of the package variables regardless of the no of forms being in that session.

August 14, 2014 Proprietary and Confidential 49

IGATE
Ignited Applications Intelligence

Since the CUSTOM library cannot contain SQL commands, that type of logic must be passed off to stored procedures or functions within the Oracle database.

Within the CUSTOM library, you are free to write almost any code supported by the Developer/2000 toolset, so long as you follow all Oracle Applications coding standards. However, there is an exception: you cannot call any APPCORE routines from CUSTOM. Almost all APPCORE routines start with the prefix "APP".

If you use Zoom or the CUSTOM library to invoke forms that you have developed, those forms must adhere completely to all of the Oracle Applications coding standards.

Attention: To invoke another form, use the function security routines. Do not use CALL_FORM since the Oracle Applications libraries do not support it.

After you write code in the CUSTOM procedures, compile and generate the library using Oracle Forms. Then place this library into \$AU_TOP/res/plsql directory (or platform equivalent). Subsequent invocations of Oracle Applications will then run this new code.

Events passed to the CUSTOM library

- Zoom.
- WHEN_NEW_FORM_INSTANCE.
- WHEN_FORM_NEVIGAT.
- WHEN_NEW_BLOCK_INSTANCE.
- WHEN_NEW_RECORD_INSTANCE.
- WHEN_VALIDATE_RECORD.
- WHEN_NEW_ITEM_INSTANCE.
- SPECIAL n and KEY Fn.
- EXPORT.

Events passed to the CUSTOM library (contd..)

➤ AOV product –specific events.

- WHEN_LOGIN_CHANGED.
- WHEN_PASSWORD_CHANGED.
- WHEN_RESPONSIBILITY_CHANGED.

➤ Other product-specific events.

- Check the documentation for your product to see a list of events passed to the CUSTOM.
- product documentation also contains list of forms that accept parameters for querying (used by zoom).

August 14, 2014 Proprietary and Confidential 499

IGATE
Ignite Your Imagination

WHEN-LOGIN-CHANGED – when a user logs on as a different user

WHEN-RESPONSIBILITY-CHANGED – when a user changes responsibilities

WHEN-PASSWORD-CHANGED – when a user changes their password

Example: Force a field to uppercase

```
begin
  If(event_name='WHEN_NEW_FORM_INSTANCE')
  then
    If (form_name='APXVENDR') then
      app_item_property2.set_property('VENODR.NAME'.CASE_RESTRICTI
ON, UPPERCASE);
    end if;
  end if;
end if;
```

Example: Change a button label

```
Begin
If(event_name='WHEN_NEW_FORM_INSTANCE')
then
If (form_name='APIVENDR') then
app_item.property2.set_property('vendr.details',LABEL,'More
Details');
end if;
end if;
end if;
```

Example: Hide a Field

```
begin
  If(event_name='WHEN_NEW_FORM_INSTANCE')
  then
    If (form_name='APIVENDR') then
      app_item.property2.set_property('venodr.note',DISPL
      AYED, PROPERTY_OFF);
    end if;
  end if;
end if;
```

Example: Change a Field prompt

```
begin
  If(event_name='WHEN_NEW_FORM_INSTANCE')
  then
    If (form_name='APXVENDR') then
      app_item_property2.set_property('vendor.name'.PRO
      MPT_TEXT, 'supplier name');
    end if;
  end if;
end if;
```

Example: Prevent inserts/updates to a block

```
begin
  If(event_name='WHEN_NEW_FORM_INSTANCE')
  then
    If (form_name='APIVENDR' and block_name='VENDOR') then
      Set_block_property(block_name,insert_allowed,property_false);
      Set_block_property(block_name,update_allowed,property_false);
    End if;
  End if;
End if;
```


Special Menu Example

- (display the special15 entry on the tools menu)

```
begin
  If(event_name='WHEN_NEW_FORM_INSTANCE')
  If(form_name='DEMXXOER') then
    App_special2.instantiate('SPECILA15','print order
again',',',TRUE,'LINE');
  End if;
End if;
End if;
```

Special Menu Example

➤ (toggle display the special15 tools menu entry.)

```
If(event_name='WHEN_NEW_FORM_INSTANCE')  
If(form_name='DEMXXOER' and block_name='ORDERS')then then  
App_special2.enable('SPECILA15',PROPERTY_ON);  
Elseif(form_name= 'DEMXXOER' and block_name='LINES')then  
App_special2.enable('SPECILA15',PROPERTY_OFF);  
End if;  
End if;  
End if;
```

Special Menu Example

➤ (code the logic for the special15 menu entry)

```
If(event_name='SPECIALS')then
If(form_name='DEMXXEOR' and block_name='ORDERS') then
/*add your print order logic here*/
Raise FORM_TRIGGER_FAILURE;
End if;
End if;
End if;
```

Coding CUSTOM.ZOOM_Available

- (If zoom is available for this block then TRUE; otherwise return FALSE.)

```
Function zoom_available return boolean is
form_name
varchar2(30):=name_in( 'system.current_form');
Block_name varchar2(30):=name_in( 'system.cursor_block');
Begin
If(form_name='FNDSCAUS' and block_name='USER') then
Return TRUE;
else return FALSE;
End if;end zoom_avialable;
```

CUSTOM.EVENT for ZOOM

```
begin
  If (event_name='ZOOM')then
    If (form_name='FNDSCAUS' and block_name='USER') then
      Param_to_pass:=name_in('user.description');
      Fnd_function.execute(FUNCTION_NAME=>'DEM
        DEMXXOER',OPEN_FLAG=>'Y',SESSION_FLAG=>'Y',OTHER_PARA
        MS=>'DESCRIPTION="||param_to_pass||" ');
      End if;else
      Null;end if;end;
```

Support

- There is a new LOV and parameter in all 11i forms:
- APPCORE_ZOOM(LOV)
- APPCORE_ZOOM_VALUE (parametre)
- Use these when you have more than one zoom from a perticular block.

Zoom LOV

- Create a record group in CUSTOM library and populate it with the names of available zooms for the block.
- Attach the record group to the APPCORE_ZOOM LOV.
- Call show_lov to show the APPCORE_ZOOM LOV.
- When the user picks a zoom, a value is returned into the APPCORE_ZOOM_VALUE parameter in the form.
- Retrieve the value and branch your zoom code accordingly.

Summary

- Custom Library Overview.
- Architecture of the CUSTOM Library.
- Routines in the CUSTOM library.
- Adding packages to the CUSTOM library.
- Coding Restrictions.
- Events passed to the CUSTOM library.
- Examples.
- Zoom LOV.



August 14, 2014 Proprietary and Confidential + 24 +

IGATE
Speed. Agility. Intelligence.

Add the notes here.

Review Question

➤ **Question 1: CUSTOM contains how many routines?**

- Option 1: 1
- Option 2: 2
- Option 3: 3



➤ **Question 2: APPCORE can be attached to CUSTOM library**

- True/False

➤ **Question 3: Where CUSTOM.pll is located?**

- Option 1: \$AU_TOP/resource Directory.
- Option 2: Fnd_top.
- Option 3: Custom_top.

Add the notes here.