Oracle Workflow and PL/SQL Integration

Helen Bucksey
Senior Consultant – Xelocity Limited

Abstract

Oracle Workflow is a powerful tool that models business processes and allows process flows to be amended easily when business rules change. Oracle Workflow can be integrated with PL/SQL, allowing access to legacy business data, database functions and procedures, giving a powerful alliance of technologies.

This paper will take you through some methods of Workflow and PL/SQL integration, covering invoking Workflow from PL/SQL, invoking PL/SQL from Workflow and exchanging data between the two.

The example Workflow process and PL/SQL package used throughout this paper are for a simple employee bonus approval process. The Workflow is initiated from the PL/SQL function Process_Bonus for a specific employee.

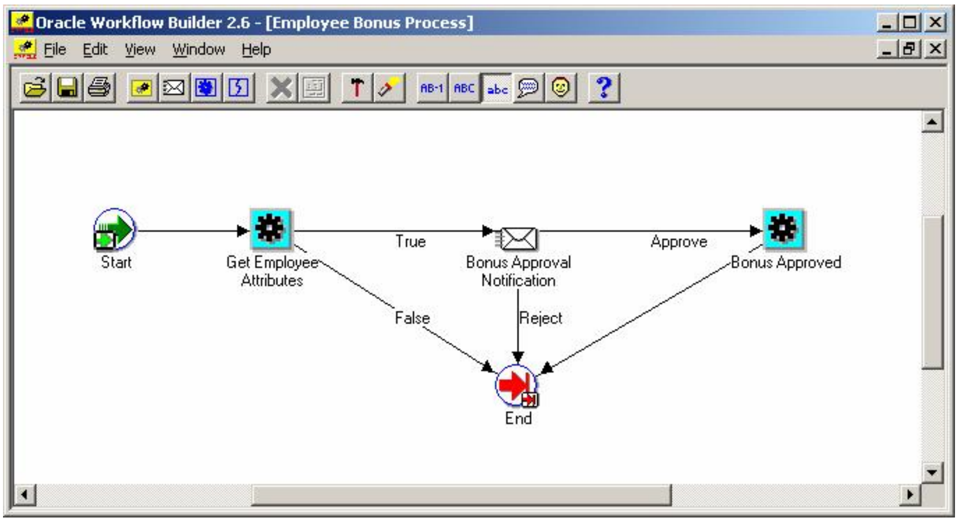


Figure 1 – The Workflow process used in this paper as an example.

Process overview:

- The Workflow function Get Employee Attributes executes the Get_WF_Attributes_For_Emp PL/SQL procedure that populates values into Workflow attributes about the employee.
- Once this is done a Workflow notification is sent to the employee's manager regarding the bonus amount for approval.
- If the bonus is approved then the Bonus Approved Workflow function executes the Bonus_Approved PL/SQL procedure that inserts a record into a database table recording the bonus details to be paid.

```
CREATE or REPLACE PACKAGE NZOUG_WF_Training AS

FUNCTION Process_Bonus(p_empno IN emp.empno%TYPE)
         RETURN BOOLEAN;

PROCEDURE Get_WF_Attributes_For_Emp(itemtype  IN VARCHAR2,
                                    itemkey   IN VARCHAR2,
                                    actid     IN NUMBER,
                                    funcmode  IN VARCHAR2,
                                    resultout IN OUT VARCHAR2);

PROCEDURE Bonus_Approved(itemtype  IN VARCHAR2,
                         itemkey   IN VARCHAR2,
                         actid     IN NUMBER,
                         funcmode  IN VARCHAR2,
                         resultout IN OUT VARCHAR2);

END NZOUG_WF_Training;
```

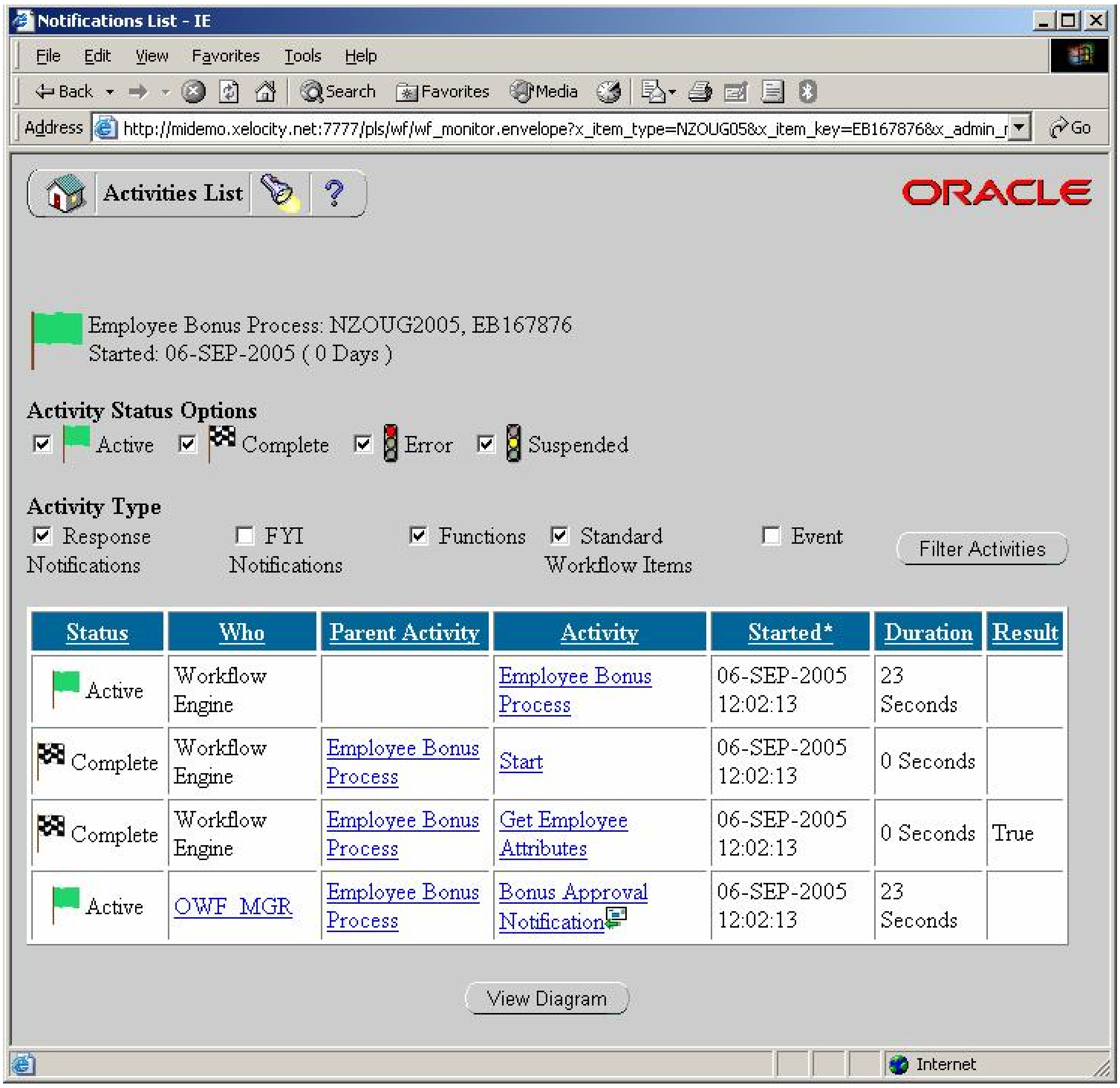
Invoking Workflow from PL/SQL

Workflow provides a flexible method of extending application code, where the process flow can be directed in a way that allows the user to have interactive control. Context-sensitive notifications and hierarchical approval mechanisms allow the user to make decisions depending on the situation and their authority level. After the initial integration with PL/SQL, changes can be made to the Workflow application logic by a super-user without involving consultants or programmers.

A series of APIs are provided in the Workflow engine that enable Workflow processes to be invoked from PL/SQL code. The WF_ENGINE package affords total control over Workflow processes and activities from PL/SQL, including launching Workflow process instances, querying process statuses and setting priorities.

```
-- First create the process for the specific employee
    OPEN cu_seq;
    FETCH cu_seq INTO l_seqval;
    CLOSE cu_seq;
    l_itemtype := ‘NZOUG05’;
    l_itemkey := 'EB' || TO_CHAR(l_seqval) || TO_CHAR(p_empno);
    wf_engine.CreateProcess(itemtype => l_itemtype,
                            itemkey  => l_itemkey,
                            process  => 'EMP_BONUS');
-- Set the process owner
    wf_engine.SetItemOwner(itemtype => l_itemtype,
                           itemkey  => l_itemkey,
                           owner    => 'OWF_MGR');
-- Set the employee number
    wf_engine.SetItemAttrNumber(l_itemtype,
                                l_itemkey,
                                'EMP_NUMBER',
                                p_empno);
-- Then start the process instance
    wf_engine.StartProcess(itemtype => l_itemtype,
                           itemkey  => l_itemkey);
```

CreateProcess() – creates a new instance of a specific runtime process.
StartProcess() – begins execution of the specified process instance.
LaunchProcess() – combines CreateProcess and StartProcess. This can be used when no instance-specific data needs to be populated.



Address: http://midemo.xelocity.net:7777/pls/wf/wf_monitor.envelope?x_item_type=NZOUG05&x_item_key=EB167876&x_admin_r

Employee Bonus Process: NZOUG2005, EB167876
Started: 06-SEP-2005 (0 Days)

Activity Status Options
☑ Active ☑ Complete ☑ Error ☑ Suspended

Activity Type
☑ Response Notifications ☐ FYI Notifications ☑ Functions ☑ Standard Workflow Items ☐ Event

Status	Who	Parent Activity	Activity	Started*	Duration	Result
Active	Workflow Engine		Employee Bonus Process	06-SEP-2005 12:02:13	23 Seconds	
Complete	Workflow Engine	Employee Bonus Process	Start	06-SEP-2005 12:02:13	0 Seconds	
Complete	Workflow Engine	Employee Bonus Process	Get Employee Attributes	06-SEP-2005 12:02:13	0 Seconds	True
Active	OWF_MGR	Employee Bonus Process	Bonus Approval Notification	06-SEP-2005 12:02:13	23 Seconds	

Figure 2 – The Employee Bonus process launched from PL/SQL, viewed from the Workflow Monitor.

Invoking PL/SQL from Workflow

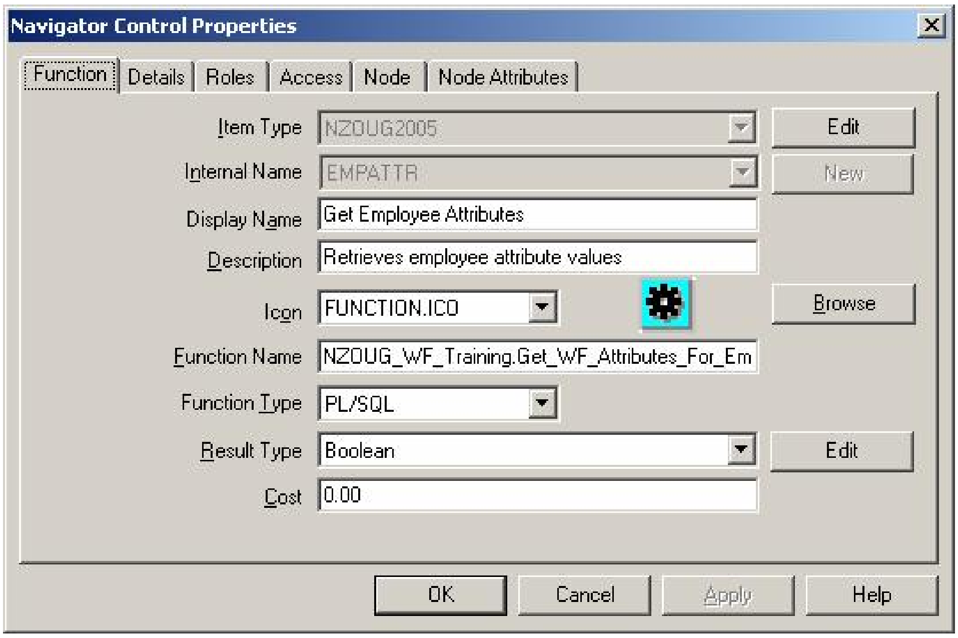
Complex calculations, reuse of existing business logic or legacy data retrieval can all be performed by PL/SQL procedures invoked from Workflow. For Oracle Applications sites, application extensions can be coded in PL/SQL and integrated into standard business flows using the Workflow method, allowing easy customisation without affecting the standard application code.

The PL/SQL procedure can be a wrapper for existing business logic, supplying appropriate parameter values from Workflow attributes or database tables. It must have five standard Workflow parameters defined:

- itemtype – internal name of the Workflow item type.
- itemkey – unique key of the specific Workflow item instance.
- actid – ID number of the activity from which this function is called.
- funcmode – activity execution mode (RUN or CANCEL).
- resultout – result to be returned from the function.

```
PROCEDURE Get_WF_Attributes_For_Emp(itemtype  IN VARCHAR2,
                                    itemkey   IN VARCHAR2,
                                    actid     IN NUMBER,
                                    funcmode  IN VARCHAR2,
                                    resultout IN OUT VARCHAR2)
IS
. . .
```

PL/SQL procedures can be executed from within Workflow by using the standard PL/SQL function type. This function type enables a Workflow process to invoke a PL/SQL procedure within its process flow and then control subsequent processing depending on the result returned by the procedure.



Item Type: NZOUG2005
Internal Name: EMPATTR
Display Name: Get Employee Attributes
Description: Retrieves employee attribute values
Icon: FUNCTION.ICO
Function Name: NZOUG_WF_Training.Get_WF_Attributes_For_Em
Function Type: PL/SQL
Result Type: Boolean
Cost: 0.00

Figure 3 – A PL/SQL function definition.

A result type can be selected to reflect the value that the procedure returns. If a value is returned it will be used to determine subsequent process flow. Any value returned from the PL/SQL procedure must match one of the result type values, i.e. for Boolean return ‘COMPLETE:T’ for True or ‘COMPLETE:F’ for False. Chapter 7 of the Oracle Workflow Guide has an example PL/SQL procedure that can be followed.

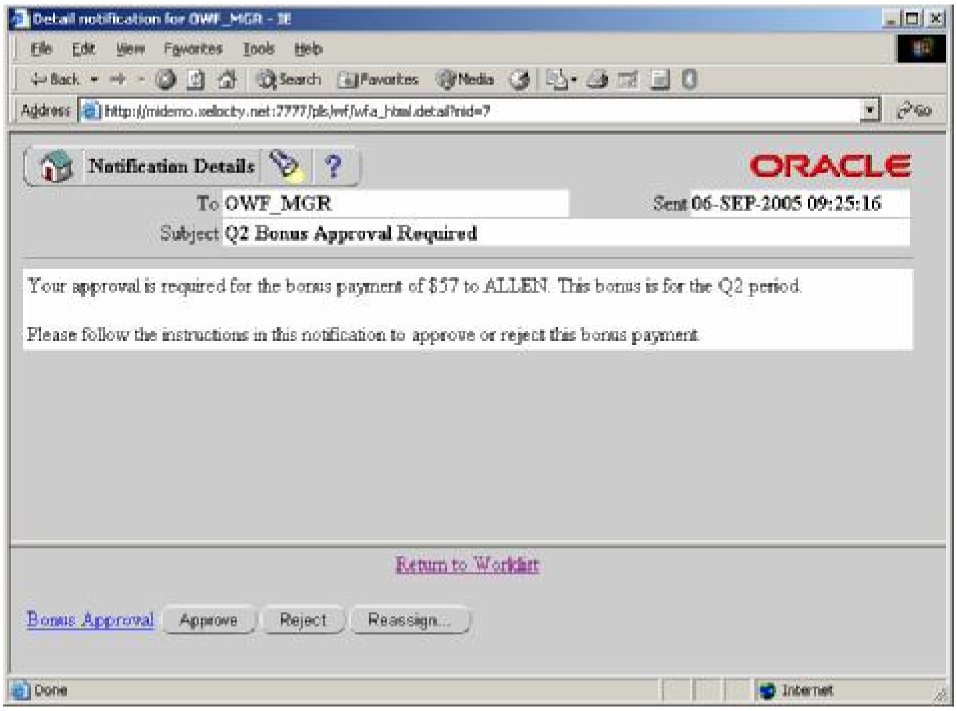
Data Exchange Techniques

Obviously any interaction between PL/SQL and Workflow will require data to be exchanged. The WF_ENGINE package provides data manipulation APIs, including GET and SET functions for all Workflow process attribute types. This extends from simple number, text and date datatypes through to complex array and document types. GET functions allow PL/SQL to retrieve current values of Workflow attributes and SET functions allow Workflow attributes to be changed from PL/SQL.

The SET function was used in the earlier example to give the employee number attribute value to the Workflow process being initiated. Here the GET function is used to find out the employee number being processed, which is used to determine other values that are then SET back into the Workflow attributes and used in notifications.

```
-- Find out which employee is being processed
    l_empno := wf_engine.GetItemAttrNumber(itemtype,
                                           itemkey,
                                           'EMP_NUMBER');
-- Populate attributes for that employee
    OPEN cu_emp(l_empno);
    FETCH cu_emp INTO l_ename,
                      l_hiredate,
                      l_gross_pay,
                      l_grade;
-- Calculate bonus based on salary grade
      l_bonus := l_gross_pay * l_grade / 100;
      wf_engine.SetItemAttrNumber(itemtype,
                                  itemkey,
                                  'BONUS',
                                  l_bonus);
      wf_engine.SetItemAttrText(itemtype,
                                itemkey,
                                'EMP_NAME',
                                l_ename);
      wf_engine.SetItemAttrDate(itemtype,
                                itemkey,
                                'HIRE_DATE',
                                l_hiredate);
. . .
```

Note that the same APIs are available for Java integration, with names beginning with a lower case letter, e.g. getItemAttr (), setItemAttrValue().


Detail notification for OWF_MGR - IE
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://midemo.velocity.net:7777/pls/wf/wfa_html.detail?nid=7
Go
Notification Details
ORACLE
Return to Worklist
Bonus Approval
Approve
Reject
Reassign...
Done
Internet

To OWF_MGR

Sent 06-SEP-2005 09:25:16

Subject Q2 Bonus Approval Required

Your approval is required for the bonus payment of $57 to ALLEN. This bonus is for the Q2 period.

Please follow the instructions in this notification to approve or reject this bonus payment.

Figure 4 – A Workflow notification using attribute values provided by PL/SQL.

Advanced Workflow

Workflow Business Events gives event-oriented process control that ties into defined events within an application. Once an event occurs any action that is subscribed to that event is invoked. Subscribed actions can be:

- Execution of custom PL/SQL code.
- Sending the business event to a predefined workflow process.
- Asynchronous messaging using Oracle Advanced Queuing.

Business events can be raised from PL/SQL or Workflow.

References

Technical white paper - Oracle Workflow Release 2.6.2 Business Event System and PL/SQL Development Guidelines.
(http://www.oracle.com/technology/products/ias/workflow/release262/wfbesdevgdln.pdf)

Oracle Workflow Guide, Release 2.6.2.
(Available from Oracle*MetaLink* http://metalink.oracle.com)