
Workflow - Replacing Flexbuilder and Alerts for Oracle Applications

Dana Tate
KPMG LLP

Introduction

Workflow is an Oracle tool designed to add functionality to the Oracle Applications in an easily maintainable and object oriented way. Additional business rules can be built into the applications to better fit each company's business practices. Account generator workflows have replaced Flexbuilder in Release 11, and workflows can replace Alerts too. Therefore it is important for future users of the applications to understand Workflow and its features.

Workflows are business processes made up of functions, notifications, and other processes. The functions are written to programatically determine a business flow. Functions are used when a person isn't required. Notifications are used when a person needs to determine the business flow by answering a question, such as whether to approve or reject a expense report. Account Generator workflows can have logic built into functions to determine the segment values to use. Alerts can be replaced as well, since Workflow allows notifications to be sent when certain conditions occur. Those notifications can also timeout if no action is taken on the notification and the workflow can escalate the notification. This is a big difference between Alerts and Workflow. This paper will explain the components of Workflow and how to alter the seeded processes to perform the Account Generator workflow. Examples of Alert applications will also be discussed.

Workflow Terms

Attributes are the variables used by messages and functions. They hold the data being passed through the workflow process. The internal name of an attribute is what is referenced in a message or function and the type defines the data that can be entered in that attribute and how it is formatted.

Lookup Types are the categories, or list of values, that hold the results that can occur in branching of a process. Type may be Approval and the valid lookup codes for that type may be Approved and Rejected.

Messages are reusable text blocks that are used to build a notification. The text is entered along with the internal names of the message attributes that are to be included in the message. The message attributes are the variable fields in the messages. They are typically the same attributes that are defined as the item type attributes. Message attributes can be informational only and are then signified as a Send attribute. If the message attribute is signified as a Respond attribute with the internal name of RESULT, then it will require an answer, or result, which will decide the path of the workflow.

Performer is the person or role that will receive the notification within a process. A role is defined usually as a responsibility within the Oracle Applications.

Functions are PL/SQL procedures that do not require a person to interact to determine the result of the function. Functions are used to do queries based on data that is being passed through the workflow. Usually the function will have a result that equals a lookup type or the result may be the value of a segment in your accounting flexfield.

Notifications are made of messages and lookup types. These notifications are sent to a performer, or person that is required to perform some action. The action may be a decision to be made or it may be for informational purposes only.

Processes are composed of functions, notifications, and other processes. Processes have no limit on the number of levels within the process. All processes usually end in a result. If the Runnable box is unchecked, then only other Runnable processes can call this process. To create a process, drag and drop functions and notifications into the

Workflow Overview

The classic example of a workflow is the approval process of expense reports. The steps to this workflow are as follows:

1. The employee enters an expense report into the system.
2. A function determines the supervisor of that employee by performing a query on HR tables and passes the name retrieved in the attribute Manager Name.
3. A notification is sent to the performer, where the Item Attribute is equal to Manager Name. The notification message asks the manager to respond whether the expense report should be approved or rejected. The result is stored in the attribute approval.
4. If the approval attribute equals Approved, then a function is run to switch the status of the expense report to allow it to be imported into AP. Also a notification is sent to the preparer attribute of the expense report letting the person know that the expense report was approved.
5. If the approval attribute equals Rejected, then a notification is sent to the preparer attribute of the expense report letting the person know that the expense report has been rejected and the reason for rejection is included. The status of the expense report remains unchanged.

Obviously this is the simplest form of the workflow. This workflow however doesn't consider that the manager may be out of town and can't answer the notification within the defined seven day turn around time. In this case a timeout of three days can be placed on the notification to the Manager. If no action is taken within those three days, the workflow can determine via the same function in step 2, who the manager's boss is. This is accomplished by simply sending the function the preparer as the manager. Then the rest of the workflow will work the same.

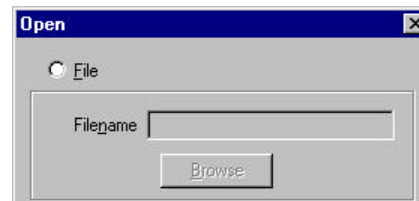
In this example it is easy to see that the PL/SQL code within a function can be written as a reusable object that changes depending on the attributes that are sent to it. Workflow processes are embedded in all the following modules of the Oracle Applications:

Accounts Payable
Engineering
Fixed Assets
General Ledger
Human Resources
Project Accounting

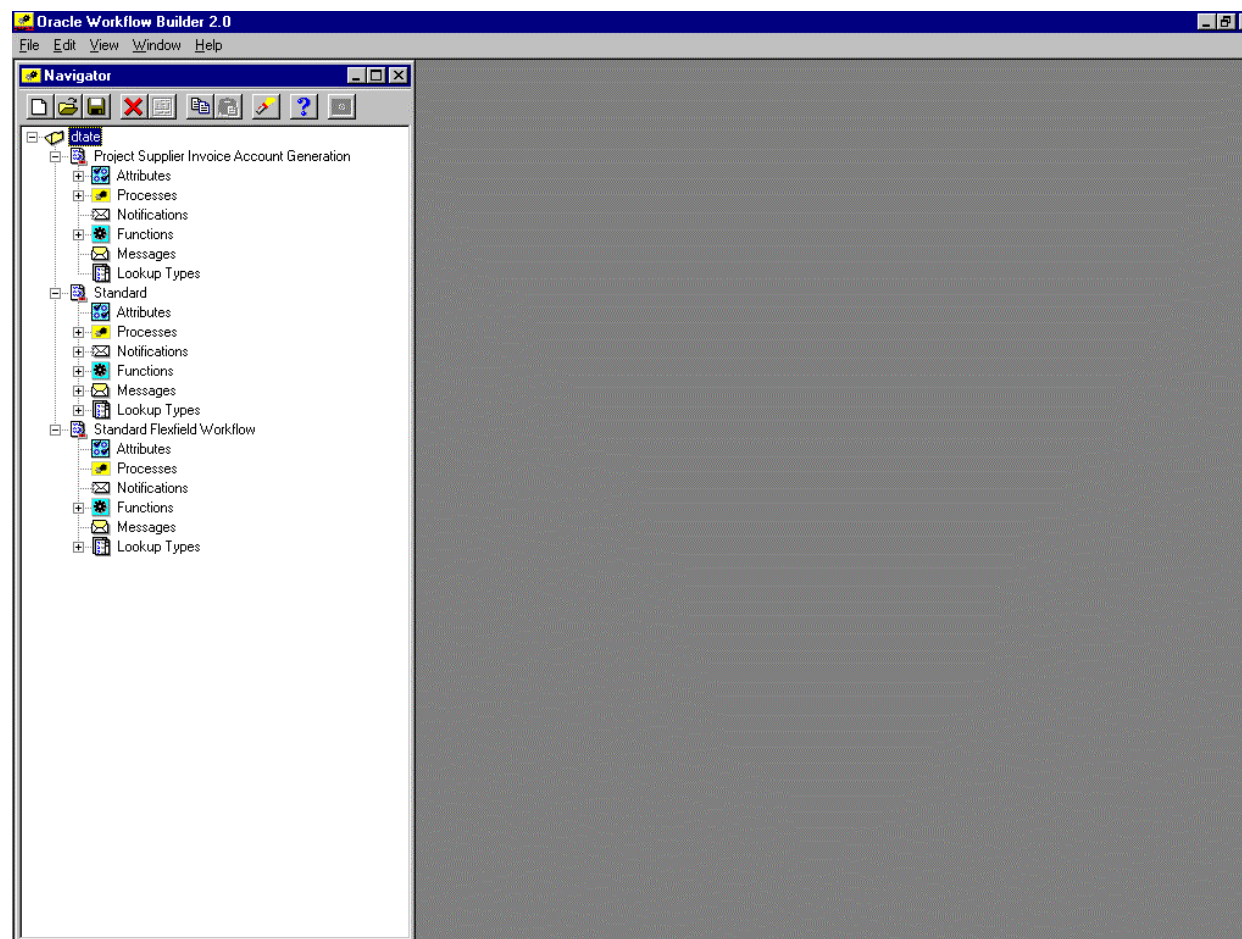
Purchasing
Service
Web Customers
Web Expense Reporting
Web Requisitions
Web Suppliers

Account Generator Workflow Example

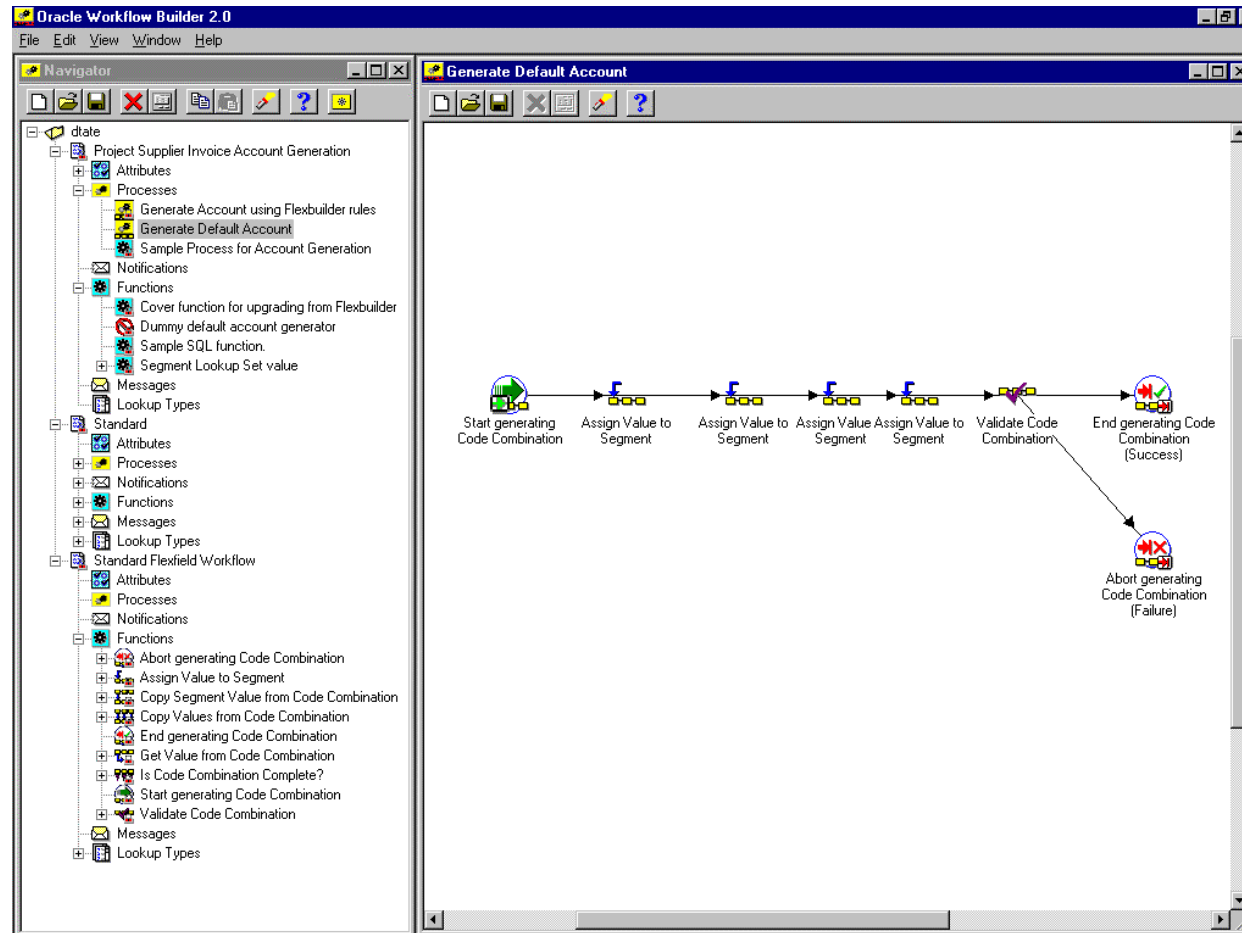
The example Account Generator workflow used will be the item type Project Supplier Invoice Account Generation. To open this workflow, Accounts Payable must be installed on the database server and Workflow Builder must be installed on a client workstation. Start the Workflow Builder application and then click on the File Menu, then Open. The following screen will be displayed:



The user name and password need to be valid database logons and the connect string is defined in the TNSNAMES.ORA file in the \\${ORACLE_HOME}\Network\Admin directory of the client PC. When OK is clicked a list of the Item Types installed on the database will be displayed on the right side of the screen under Hidden. Select the ones to show and move them to the Visible box and then click on OK. The Workflow Builder will then open the item types selected. The screen will look similar to the one below if the Project Supplier Invoice Account Generation item type was selected.



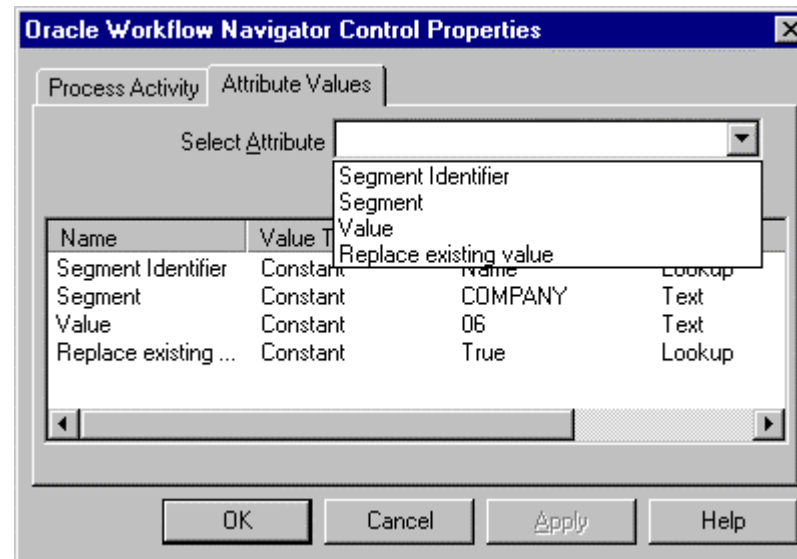
The Standard item type and the Standard Flexfield Workflow item type are each included with each Account Generator workflow. To see the functions available under the Standard Flexfield Workflow click on the plus sign beside the word Functions. These functions are available to be used in all Account Generator workflows. To see the processes defined in the Project Supplier item type, click on the plus sign beside Processes. To view the Generate Default Account process, double click on the process name and the process will be displayed in the window on the right side of the Workflow Builder. This particular process was designed to generate an accounting flexfield with four segments, Company, Account, Department, and Product Line. Each segment is being assigned via a function called Assign Value to Segment. Then the combined accounting flexfield is being validated in the GL code combination table and if successful, end with the result Success, else end with the result Failure. The Validate Code Combination function is the procedure FND_FLEX_WORKFLOW_APIS.VALIDATE_COMBINATION.



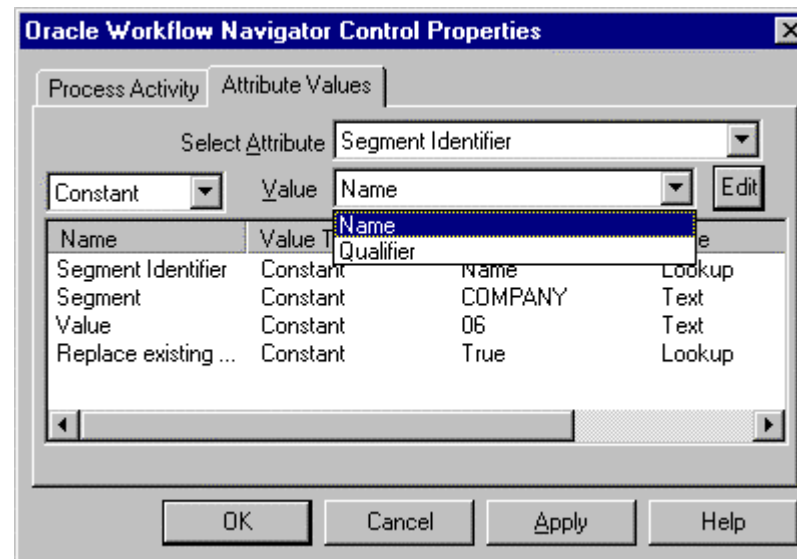
To assign values to the various segments, right click on one of the Assign Value to Segment nodes and then click on properties. The dialog box similar to the one below is displayed when clicking on the first node with this name:

The screenshot shows the 'Oracle Workflow Navigator Control Properties' dialog box. It has a title bar with a close button. The 'Process Activity' is set to 'Attribute Values'. Below this, there are several fields: 'Item Type' is 'Standard Flexfield Workflow' with an 'Edit' button; 'Function' is 'Assign Value to Segment' with an 'Edit' button; 'Label' is 'FND_FLEX_ASSIGN_TO_SEGMENT'; 'Start/End' is 'Normal'; and 'Comment' is '2: Assign value to Company Segment using a constant'. The 'Performer' field is partially visible at the bottom.

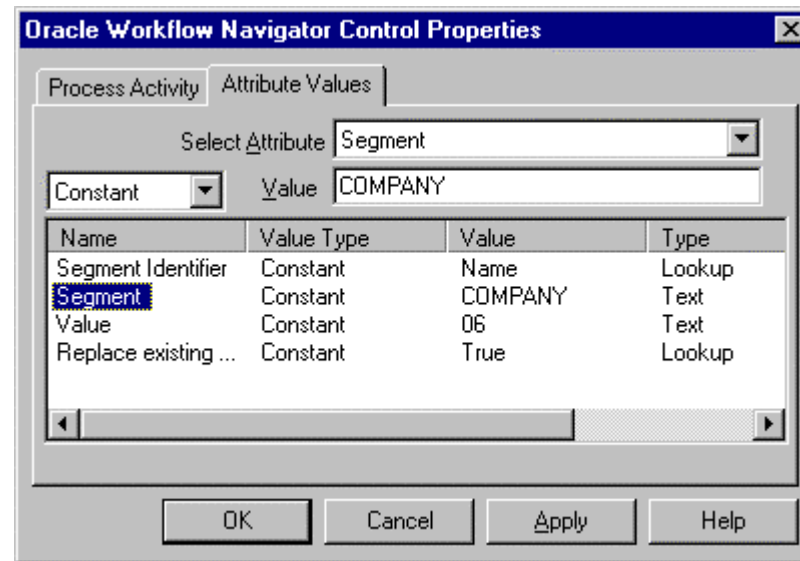
To adjust the values, click on the Attribute Values tab and the following window is displayed.



To select the attribute to change, click on the Select Attribute pick list and click on the Segment Identifier. The following fields are displayed.



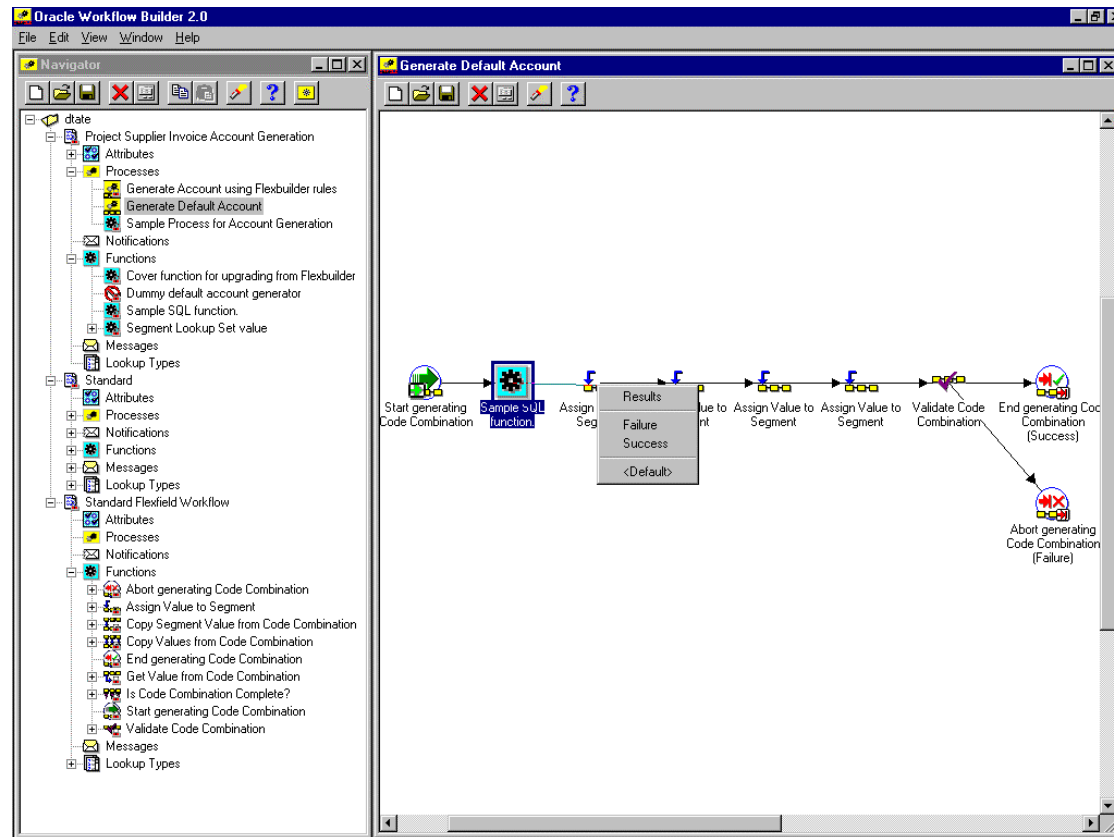
Usually the Segment Identifier is a Constant, and the value is either Name or Qualifier. The Name of the segment will be the Accounting Flexfield Segment Name, but the Qualifier would be whether the segment is the Balancing Segment, Natural Account Segment, or the Cost Center Segment as defined on the Accounting Flexfield Qualifier screen. In this example the Name is selected, so the Segment value will be displayed as the Constant COMPANY.



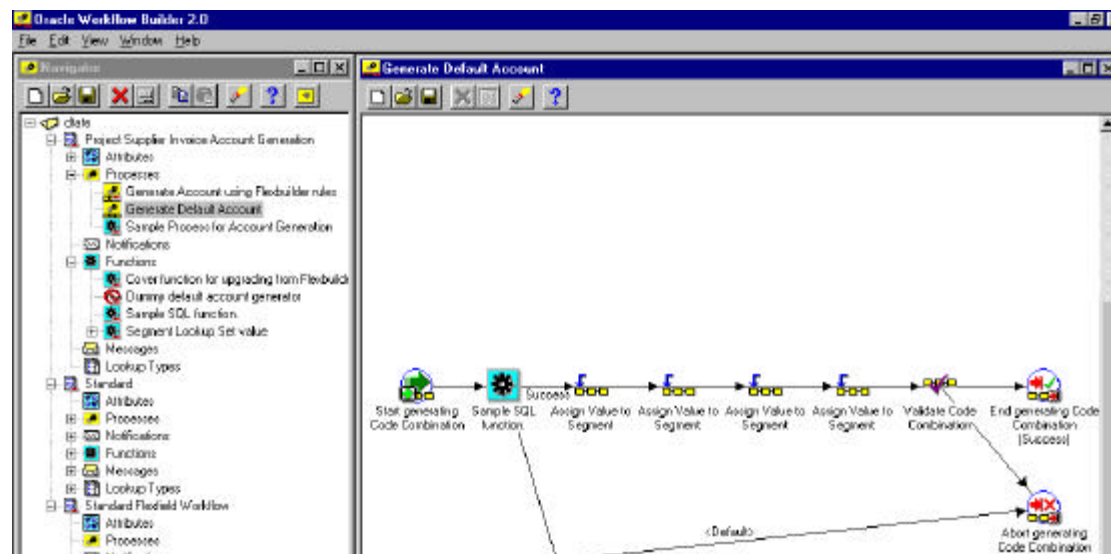
The Value for the Segment in this example is a constant but an item attribute could also be applied as the value. To enter the constant value, type the value 06 in the Value field. If you wanted to select an item attribute, then instead of Constant, Item Attribute would be selected. The list of available Attributes would display in the Value field. The Replacing existing value attribute should usually be a constant with the Value of True because the existing value of the attribute is being replaced. Then the values added to the flexfield will be retained. This completes assigning the value to the first segment of the accounting flexfield. Each segment of the Accounting Flexfield must have one of these functions assigned to it. Since this example uses a four segment accounting flexfield, the Assign Value to Segment function is reused four times with different parameters.

If the values are not constant and need to be calculated based on data included within the process, a function can be placed before the Assign Values to Segments nodes to make the query to determine the values to assign. In this case usually a couple of the segments are calculated and the others are constant. For example, the company and the department may be calculated based on the project being invoiced. A query could be written looking at the project tables for the project number and retrieving the company and department to be charged. Oracle provides a Sample SQL function to customize this query. The function used in the Project Supplier Invoice Account Generation for the Sample SQL is defined as using the procedure called PA_WF_FB_SAMPLE_PKG.PA_WF_SAMPLE_SQL_FN. This function shows a sample query and how to assign the results to an attribute whose value can then be assigned to a segment by using the Item Attribute value instead of the constants.

The workflow process Generate Default Account would also need to be altered to add the Sample SQL function to the workflow before the Assign Value to Segment functions. To do this, drag and drop the Sample SQL function to the process window. Then delete the transition between the Start node and the first Assign Value to Segment by clicking on the line and then pressing the delete key. Then connect the Start node to the Sample SQL function by right clicking on the Start node and dragging the line over the Sample SQL function node and releasing it. Then to connect the Sample SQL function to the First Assign Value to Segment node, right click on the Sample SQL node and drag the transition line over to the first Assign Value to Segment node and release. A dialog box will be displayed asking whether this transition should be Success, Failure, or default result. This transition should be the Success path. The screen will look like the one on the following page when creating the transition between the Sample SQL function and the first Assign Value to Segment node.



A transition should also be created between the Sample SQL function and the End Node designating Default, which means anything other than Success goes through this transition. If the Sample SQL function is not successful, the default transition should be to go to the End with a Failure result type. The workflow should look as the one below.



The account generator workflow is now edited, and is ready to be saved back to the database. To do this, click on File then Save. If there are any errors in the workflow, it will display them and not upload the workflow until they are fixed.

Alerts as Workflows

Alerts are typically used to determine exception conditions or to report inconsistencies within the data. For example, an alert can be created so that when an employee's salary is altered within Payroll, an email is sent to a human resources manager. The same process can be accomplished via a workflow process. A function could be written to query the database for any changes to the Payroll system. If the function returns a value of True, then someone's record has been changed and a notification could be sent to the HR manager. If the function returns a value of False, then no notification is sent. This is a simple workflow process, but the difference between Alerts and Workflow is that the notification that is sent can be required to be acknowledged within one day or be escalated to the HR manager's supervisor by sending a notification to that supervisor as well. Alerts can't do this. This escalation feature is very popular with Oracle Service users, especially if the service agreements state that a callback will occur within two hours on any service call. Once the service call is assigned to a representative, a workflow escalation process can be designed to escalate the request if the representative hasn't responded to the notification yet. Customers will be happy, and reports can also be written to extract the data on the number of times a call is escalated and from where it was escalated. The Workflow tables are listed in the AOL technical reference manual and a tool such as Discoverer can query that data and create the report.

Workflow's added functionality and flexibility make it an easy alternative to Alerts. The timeout/escalation feature within one workflow process is much better than having to write multiple Alerts to accomplish the same function.

Conclusion

Workflow is a very useful tool to enhance the functionality delivered with the Oracle Applications. Business rules unique to each company can be easily defined by editing the existing workflow processes. The additional functionality that Workflow gives the Oracle Applications allows for customization to be done outside of the apps. As with any customizations to the Applications, the changes should be saved in a directory that is protected from upgrades of the Oracle Applications. You can save these processes to the database or to files, which makes version control and storing of customizations easier.

About the Presenter

Dana Tate is a consultant in the Financial Services/Real Estate and Hospitality/ERP Practice of KPMG LLP. As such she has gained experience on multiple Oracle implementations of Oracle Financials and Self Service Web Applications, such as Web Employees, Web Customers, and Web Suppliers. Most of these implementations have been for international companies. She has taught Oracle Education courses for both Web Application Dictionary and Workflow. Dana is also the 1999 President of the Dallas Oracle Users Group in Dallas, Texas, USA. Her experience also includes project manager for multiple custom designed applications on Oracle databases over six years at a major financial institution. Other presentation experience includes OAUG Europe 1999 and OAUG Spring 1999.