

Oracle Approvals Management (AME) Case Studies for AP, PO and HR

Susan Behn, *Solution Beacon, LLC*
Gerald Jones, *Solution Beacon, LLC*

Introduction

Oracle defines Oracle Approvals Management, otherwise referred to as AME, as a self-service application that allows users to define business rules that govern the approval process for transactions in Oracle applications integrated with AME. In other words, the AME engine provides the opportunity for an organization to transform their unique business cases into approval routing rules for a particular transaction.

This paper will discuss how AME can be utilized to create both simple and complex business cases involving the approval of Oracle Payables (AP) invoices, Purchase Requisitions (PO) and Human Resources (HR). It also will discuss the basic components of the AME application (i.e. attributes, conditions and rules) that are required as part of AME setup for any integrating application. Finally, it will discuss how workflow has been integrated with AME to drive invoice approval routings. The current AME Implementation guide provides more in-depth information on the AME engine and other advanced features which is outside the scope of this paper.

The paper assumes that the current environment is using the E-Business Suite version 11.5.10.2 or Release 12 and Oracle Approvals Management patched to mini-pack B.

AME Basics

The purpose of AME is to find approvers based on specific business rules to facilitate the approval of a transaction within an organization. Sometimes, these rules are simple. For example a simple rule regarding approval of AP invoices may require any invoice greater than \$100 to be approved by the requester's immediate manager or supervisor. However, many organizations have more complex rules regarding approval routings. An example of a complex approval routing requirement could be an organization requiring any invoice greater \$10,000 that is matched to a purchase order including computer equipment items, to be approved by the IT manager along with the requesters two immediate supervisors. The flexibility of the AME application allows a unique approval requirement such as this to be developed. Whenever approval rules are developed in AME for an application, it identifies all of the necessary approvers for a given transaction (invoice) and notifies them through workflow notifications.

AME is integrated with multiple business applications and is expanding with each subsequent release of Oracle Applications. APIs are also provided for the AME application to allow integration with third party applications.

Advantages of AME

The AME application provides many advantages for the business user looking to develop approval rules for various applications. The most obvious advantage of an automated approval process is to speed up the approval cycle. Another advantage is that business rules can be setup in AME without the business user having to write programming code or customizing the application. Another advantage of AME is that the approval routing can leverage some of the hierarchical structures that already exist in Oracle applications such as HR employee/supervisor or HR positions hierarchies. Additionally, if an organization has requirements for an invoice to be routed to a particular individual or groups of individuals, AME provides the business user the ability setup specific hierarchies on which approval routings can be based.

One of the unique features of AME is its ability to respond to any changes that may occur in an organization during the approval process of a transaction. The changes can include organizational hierarchy changes (i.e. supervisor/manager), modifications to an AME rule or even changes to the values of the current transaction.

How does AME do this? Whenever an identified approver for a transaction responds with an approval, AME reconstructs a new approver list based on the most current conditions in the application including the current AME rules for the transaction. This includes the current values of the application and any changes to the approval hierarchy on which the transaction rule is based. For example, imagine the original rule in AME requires an invoice greater than \$1,000 to be approved by the requester's two immediate supervisors. The approval for an invoice is initially sent to the most immediate supervisor. If a new rule is created prior to the final approval (by the second supervisor) requiring the review and approval by a tax accountant, a new approver list is built that includes the tax accountant approver after the first approval.

AME Components

In order for a business user to develop business scenarios in AME that determine approval routings, it is important to understand the different components within AME. These components are often required to be modified or created as part of the development of business cases. A brief description of these components will be discussed in the following paragraphs.

Configuration Variables

Configuration variables control runtime behavior in AME. Default values which apply to all transaction types are seeded. These variables can be modified for any transaction type. One example is the variable to allow FYI notifications. The default value is "No" for all transactions. For some transaction types, such as for Human Resources, this value can be set to "Yes." However, be aware that setting this value to "Yes" for transaction types that do not support this feature will cause AME to fail. Refer to MetaLink Note Id: 338508.1.

Transaction Types

A transaction type describes the type of transaction for which business rules and approval routings will be based. This can include Oracle Application transactions such as purchase requisitions, sales orders or accounting journals. Oracle provides many seeded transaction types to satisfy many of the common transactions that occur within a particular application. The creation of a new transaction type in AME is available for those business users that want to integrate custom applications with AME. However, Oracle typically does not encourage the development of new transaction types because of the significant programming effort involved to integrate with the AME application. There are exceptions such as for iRecruitment Vacancy Approval where Oracle actually recommends creating a new transaction type if the user chooses not to use the seeded rules available for this transaction type. The difference in this case is the vacancy approval process is already integrated with AME and the transaction type name is a parameter.

Attributes

Attributes within AME are business variables that represent the value of a data element of a given transaction. In the case of an AP invoice transaction, a typical attribute would be invoice amount or supplier name. Attributes can be thought of as the 'building blocks' of business case development. The reason being is that the value of an attribute(s) for a transaction can ultimately determine whether a business case (approval rule) has been met because approval rules use conditions which in turn use attributes. Most of the attributes you will need to create business rules are seeded with the transaction type. Additional attributes can be created as needed as long as they can be linked to the source transaction.

Attributes in AME can be created as being static or they can be dynamic in nature. Static attributes have a constant value that remains the same for each and every transaction associated with the attribute's transaction type. Dynamic attributes use a SQL query to retrieve the value of an attribute at runtime whenever a transaction is created. Most attributes in a transaction type are dynamic.

There are several different attribute types that exist within AME. String attributes are alphanumeric in nature and can have a total length of 100 characters. Numeric attributes are considered to be any numeric value that is acceptable in PL/SQL. This includes numbers containing decimal or sign operators (+/-). AME requires that any numeric attribute that is dynamically generated to be converted to a canonical form. This can be done by using the syntax **fnd_number.number_to_canonical** function as part of the dynamic SQL query. An example dynamic SQL query for a numerical attribute would be in the following syntax:

```
SELECT fnd_number.number_to_canonical(:requester_id)
FROM ap_invoices_all
WHERE invoice_id = :transactionId.
```

Currency attributes are used whenever the transactions of an organization involve multiple currency values. This allows for Oracle to use currency conversion between denominations when retrieving the value of an attribute. AME requires that any dynamic attribute setup as a currency attribute must include the following columns as part of the SQL query: numeric column, currency and conversion method. One caveat to mention regarding currency attributes; any AME conditions that are developed using a currency attribute must include a condition for each currency this particular transaction attribute value might have.

Boolean attributes have only two allowable values; true and false. Any dynamic attribute defined as a Boolean must return one of these two allowable results. AME provides a format string that can be used in the SQL query of a dynamic Boolean attribute. The syntax format is in the form of either **ame_util.booleanAttributeTrue** or **ame_util.booleanAttributeFalse**.

Date attributes are commonly used on transaction data that contains a date value, such as invoice date. AME requires that date attributes be returned in the format 'YYYY:MON:DD:HH24:MI:SS'. AME provides a format string that can be used in the SQL query of a dynamic date attribute. The format string **ame_util.versionDateFormatModel** can be used to return the proper date format at runtime.

All transaction types currently defined in AME use several mandatory attributes that can be thought of as *runtime parameters* because they often determine various facets of AME runtime behavior. These attributes can control AME behavior such as whether to allow an approver to appear multiple times on an approval hierarchy or whether to allow a requester to approve his/her own transactions (invoices). The following mandatory attributes are defined in AME for all transaction types:

```
ALLOW_DELETING_RULE_GENERATE_APPROVERS
ALLOW_REQUEST_APPROVAL
AT_LEAST_ONE_RULE_MUST_APPLY
REJECTION_RESPONSE
USE_RESTRICTIVE_ITEM_EVALUATION
EFFECTIVE_RULE_DATE
EVALUATE_PRIORITIES_PER_ITEM
USE_WORKFLOW
WORKFLOW_ITEM_KEY
WORKFLOW_ITEM_TYPE
REPEAT_SUBSTITUTION
```

The AME Implementation guide provides additional detail on each of these mandatory attributes and how they are interpreted by AME.

Required Attributes are similar to mandatory attributes in that they determine runtime behavior of AME. The only difference being that required attributes are defined specific to a transaction type. In the case of the Payables Invoice Approval transaction types, the following require attributes are defined.

ALLOW_EMPTY_APPROVAL_GROUPS
FIRST_STARTING_POINT_PERSON_ID and SECOND_STARTING_POINT_PERSON_ID
INCLUDE_ALL_JOB_LEVEL_APPROVERS
JOB_LEVEL_NON_DEFAULT_STARTING_PERSON_POINT_ID
NON_DEFAULT_POSITION_STRUCTURE_ID
NON_DEFAULT_STARTING_POINT_POSITION_ID
SUPERVISORY_NON_DEFAULT_STARTING_POINT_PERSON_ID
TOP_POSITION_ID
TOP_SUPERVISOR_ID
TRANSACTION_REQUESTER_PERSON_ID

The AME Implementation guide provides details of each of these mandatory attributes and how they are interpreted by AME.

Both mandatory and required attributes come seeded with default values. They can be modified to meet the needs of a specific transaction type.

Conditions

The next major component of AME setup is conditions. Conditions are used to evaluate the value of attributes in a particular transaction. The result of a condition can either be true or false. Conditions are precursors to AME business rules. The result of a condition helps to determine whether a business case (rule) has been satisfied. The conditions within AME can be better thought of as the *IF* part of an approval rule. For example,

If invoice supplier is Vendor A, then require approvals from Approver A, Approver B

In this example, AME would retrieve and evaluate the value of the attribute *invoice_supplier* to determine if the value was equal to Vendor A.

There are three different types of conditions that exist in the AME application: Ordinary-Regular, Ordinary-Exception and List Modifier. Ordinary-Regular conditions associate an attribute with a defined value or range of values (e.g. *invoice_amount > 100*). Ordinary-Exception conditions are similar to Ordinary-Regular conditions in how they are defined, but differ in that they are limited to the types of rules with which they can be associated. This will be discussed later in the document. Finally, List-Modifier conditions provide the ability to create conditions based on the existence of a specific approver in an approver list that is built by AME for a specific transaction. For example, a List-Modifier condition could be defined as follows:

If Approver B is final approver, require approver up 1 level

This condition would evaluate to true if Approver B was the last approver in an approver list built by AME at runtime.

Action Types and Actions

Actions within the AME application describe the nature of what should be done in AME if a particular condition and rule is satisfied by a transaction. It is the actions that dictate the approver list that is generated by AME for the given transaction. Actions not only provide instruction as to who the approvers are, but how many approvers are required for a given transaction and in what order should they be notified.

Action types are groupings of actions that have a similar functionality such as the approval hierarchy that should be traversed when building an approver list. An example of this would be actions that pertain to building an approval based solely on the supervisor tree in HR. The multiple actions for this action type would all pertain to traversal of the supervisor hierarchy, but would express in terms of how many levels to traverse. Typically, each action that describes how many levels of a hierarchy to move up would be defined separate unto itself. All of the defined actions would be grouped into an action type based on their relationship to the hierarchy being used.

Although the defining of a new action type is possible in the application, AME provides a broad number of action types that should satisfy most approval requirement of an organization. The action types that are defined in AME are categorized and can be grouped into four different types of hierarchies they navigate: Chain-of-Authority, List Modification, Substitution, Approver Groups & Production.

Action types and their associated actions tend to be one of the more complicated concepts to understand in AME. The following tables are intended to provide a basic understanding of action types as they are categorized in AME.

Chain-of-Authority Action Types

These action types typically utilize either the supervisor or position hierarchy defined in Oracle HR to generate an approver list for a given transaction.

Action Type	Description	Example
Absolute-job-level	Ascends the HR supervisor hierarchy until an approver with the appropriate job level is found	Require approvals up to job level 6
Relative-job-level	Ascends the HR supervisor hierarchy until an approver with a number of job level above the job level of the requester of a transaction	Require approvals at least 4 levels up
Manager-the-Final-Approver	Ascends the HR supervisor hierarchy, but only the immediate supervisor and final approver on approval list are required to approve	Require approval up to first supervisor and CEO only
Final-Approver-Only	Ascends the HR supervisor hierarchy, but only requires approval from the person that is last on the approver list	Require approval from division manager only
Dual-Chains-of – Authority	Ascends the HR supervisor hierarchy and builds two separate list chain of approvers. Must provide approval action for each chain.	Require approval from previous employee supervisor and current employee supervisor (i.e. during employee transfer)
Line-Item-Job-Level	Ascends the HR supervisor hierarchy. Enables approval chains to be built based on line level item in a transaction	Require approval from manager of an accounting code cost center distribution segment on an invoice distribution line
Supervisory-Level	Ascends the HR supervisor hierarchy based on a fixed number of required approvers. There is no correlation or dependency on job level.	Require approvals up to 3 supervisors
HR-Position	Ascends the HR position hierarchy up to a specified position	Require approvals up Accounting Manager
Position-Level	Ascends the HR position hierarchy up a specified number of positions	Require approvals up to position at level 4

List-Modification Action Types

Like the Chain-Of-Authority action type, the List-Modification action type also traverses the HR supervisor hierarchy structure. However, its intent is to modify the inherit approver list by either granting or revoking authority limits. It accomplishes this by either extending or removing individuals from an approver chain that might normally be built for a transaction. This modification of the approver chain is based on the target approver specified when using the action type. An example of this would be allowing an IT manager be the final approver of a computer purchase even though the amount of the transaction might normally require approval up to say the CFO level. However, because the IT manager may be more knowledgeable of the purchase, the approval list is shortened to allow them to be the final approver of the transaction.

Action Type	Description	Example
Final-Authority	Grants final authority to an approver that typically does not have signing authority by ending the approval chain when the approver chain reaches the designated approver	If Approver is IT Manager, allow final approval
Non-Final Authority	Revokes final approval authority to an approver that normally has sufficient signing authority by extending an approval chain beyond the final approver until a targeted or designated approver is reached	If Approver is Harry, require approval up to one supervisor

Substitution Action Types

Substitution action types allow for a target approver to be replaced by another designated approver whenever a transaction generates an approval chain involving the target approver. An example would be routing approvals to an employee in the absence of another.

Approver Group Action Types

Approver group action types allow an approver list to be built based on a predefined list of members that are part of the approval group. This list can be static, containing the specific names of individuals responsible for approvals of a given transaction. The list can also be dynamically built based on a SQL statement.

Action Type	Description	Example
Approver-Group Chain-of-Authority	Chain-of-authority list is built in the same manner as the List-Creation or List-modification action types. However, instead of using the HR supervisor or position structure, it uses a predefined approver group list to generate the chain-of-authority	If transaction item contains building material, require approvals from Mark, then Chris and then Sharon.

Pre and Post Chain-Of-Authority	Inserts an approver list either before or after the normal approver chain that might be generated for a given transaction.	If transaction includes sales tax, require approval by Sales Tax group, then require approval up to job level 3
---------------------------------	--	---

Rules

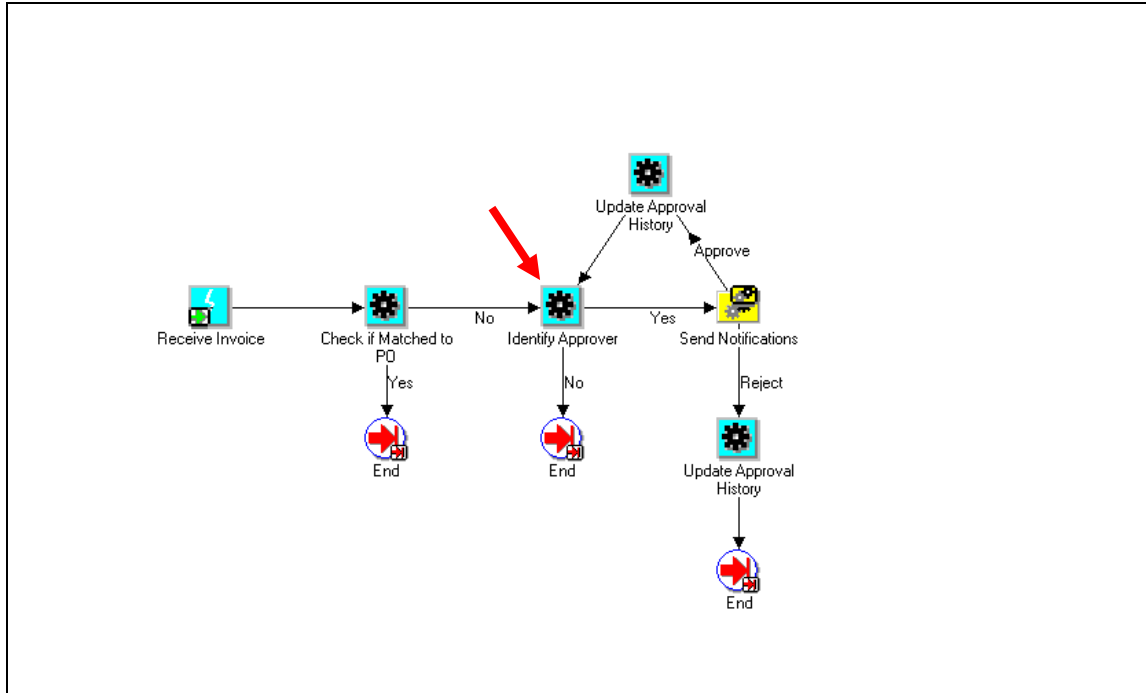
Rules could arguably be considered the essential component when defining business cases for an organization. Whenever a transaction is initiated, the rules are evaluated to determine what approval path the transaction will follow when submitted for approval. Rules are defined by associating the conditions that ultimately determine whether a particular business case has been satisfied. Additionally, rules are associated with action types to determine the approval action and thus the approver list that should be generated for the transaction.

Similar to action types, rules use rule types to determine the type list to build if the results of the rule are true. The following table briefly explains the current rule types for the Payables Invoice Approval transaction type in AME.

Rule Type	Description
List-Creation	Builds a chain-of-authority list of approvers that ascend some organizational hierarchy to generate the chain of approvers.
List-Creation Exceptions	Builds a chain-of-authority list of approvers as well. However, it is used often used to suppress a list-creation rule so as to require approvals from a certain group of approvers if an additional condition is met. An example is a rule to force approval by a particular group of approvers based on cost center of an invoice distribution account.
List Modification	Allows for rule to grant or revoke final approval based on a transaction's condition.
Substitution	Allows for delegation of an approval authority to another approver. Example is when someone is on vacation and they want to assign an individual to handle their approvals.
Pre/Post Approval Rules	Allows for additional approvers outside of a transaction's generated chain-of-authority to be added to the list of approvers.
Combination Rules	Combines actions from action types having different rule types. Often used when several rules apply to the same business transaction.
Production Rules	Outputs runtime values to a particular integrating transaction.

Invoice Approval Workflow and AME

So how does workflow integrate with Oracle Approvals Management? We will use the Invoice Approval Workflow as an example. All transactions work in a similar manner. The simplest answer is to mention that they integrate through use of the AP Invoice Approval workflow. Whenever AP is configured to use workflow, all invoices (manual and imported) are subject to invoice approval. This is done by initially setting the approval status of the invoice to *Required*. Once the invoice is validated and approval is initiated for the invoice either online or via the *Invoice Approval Workflow* concurrent program, the invoice falls into the workflow cycle. The approval logic can best be explained by reviewing the Invoice Approval workflow.



Approval Logic

When an invoice transaction falls into the approval workflow, the workflow determines if the invoice transaction is fully matched to a purchase order. If it is, then the workflow ends and the approval status of the invoice is updated to *Not Required*. However, if the invoice is not matched to a purchase order, then the workflow tries to identify the first or next individual responsible for review and approval of the invoice. The workflow node *Identify Approver* is where AP and AME are integrated. It is at this point that the workflow calls AME to determine either a) does the invoice initially meets any of the currently defined rules in AME for invoice approvals or b) are there any additional approvers left on the approval chain hierarchy.

For any AME rule satisfied by the invoice transaction, AME attempts to build an approver list based on the applied rule and the associated action type and actions that define the appropriate approvers. If a successful approver list is built, then the workflow sends a notification to the first approver in the list. The workflow itself remains active and continues to call AME as long as:

- There are more approvers left on the approver chain
- The workflow has not been rejected by any approver
- The workflow has not expired due to non-responses by an approver

As you can see, the components that are defined in AME, especially the business rules have a direct impact on the approval routings within AP invoice workflow. It is very important to plan and define your rules carefully to ensure that the organizational approval requirements are met and approval routings flow as intended.

One thing that is important to note is the behavior of AME and workflow for invoices that do not satisfy any predefined rule. By default, the approval status of any new invoice is set to *Required*. Once the invoice is sent for approval, either manually by the user online or via the Invoice Approval Workflow, program, the approval status of the invoice changes to *Initiated*. When the workflow begins, if the invoice transaction does not initially satisfy any approval rules in AME, the workflow ends and the status of the invoice remains *Initiated*. This is the behavior of the AP Invoice workflow delivered with Oracle. It is important to mention this because whenever an organization decides to require approvals of invoices, the

invoices cannot be paid until the invoice is approved. So for any invoices that fall into the category of not satisfying any approval rules, this could potentially prevent these invoices from ever being paid. There are a couple of alternatives an organization could choose to resolve this issue. The first of which is to modify the AP Invoice workflow to deal with any invoices that do not initially meet the conditions of an approval rule. Modification to the workflow is beyond the scope of this document. The second alternative has to do with the setting of the mandatory attribute *AT_LEAST_ONE_RULE_MUST_APPLY*. Setting the value of this attribute to *True* will cause the workflow to raise an exception for any invoice transactions that do not satisfy at least one defined rule. In this case, an organization could at least be aware that their rules defined in AME do not cover any all business cases that exist in regards to invoices transactions.

Implementing AME for AP Invoice Approval

In order to use AME to facilitate AP invoice approval routings, there are some setup steps that must be completed in both the AME application as well as within the Payables applications.

AME Setup

The first step in implementing AME is to install the AME application. As of release 11.5.9, AME comes seeded and is installed as part of the overall applications install. The setups that are described in this document are for the latest version of the AME applications. (As of this writing, AME.B is the latest RUP (Roll-Up Patch) available).

You can find the most recent patch for AME at site by using the Simple Search function under the Patches & Updates tab on MetaLink.

Simple Search

Advanced Search | Quick Links | Saved Searches

Search By

(ex. RDBMS Server)

Release

Patch Type

Platform or Language

Results for Platform : *AIX Based Systems (32-bit)*

Tip Consider Saving the search to make it easy to run again.

Patch	Description	Release	Updated	Size			
5055050	Human Resources Suite Family: Patchset 11i.HR_FF.K RUP 1	11i	30-JUN-2006	472M			
4428060	Approval Management Engine: Patchset 11i.AME.B	11i	30-SEP-2005	47M			
3500000	Human Resources Suite Family: Patchset Patch 11i.HR_FF.K	11i	05-JUL-2005	495M			
3333633	Human Resources Suite Family: Patchset Patch 11i.HR_FF.J	11i	14-MAR-2005	468M			
3962268	Approval Management Engine: Patchset Patch 11i.AME.A	11i	01-DEC-2004	47M			

Total: 5

The next step in setting up the application is to set up AME security. The current version of AME uses Oracle Role Based Access Model (RBAC) which is part of the new User Management model to provide access to the various AME component functions. An AME role must be attached to the user account of any person utilizing AME to develop application business rules. The following roles are predefined in AME.B.

- Approvals Management Administrator
- Approvals Management Analyst
- Approvals Management System Viewer
- Approvals Management System Administrator
- Approvals Management Process Owner

Approvals Management Business Analyst¹

Granting a role to a user does not automatically provide access to the setup components within AME. As part of the RBAC model, once a role has been granted to a user, specific access must be granted, to access functions within the role in order to 'activate' access to the functions. In terms of AME, this means granting access to either all or a specific transaction type. For example, if a business user is responsible only for the setup of the *Payables Invoice Approval* transaction type, then a specific access to this transaction type can be granted, thus allowing the user to only access and modify components of this one transaction type.

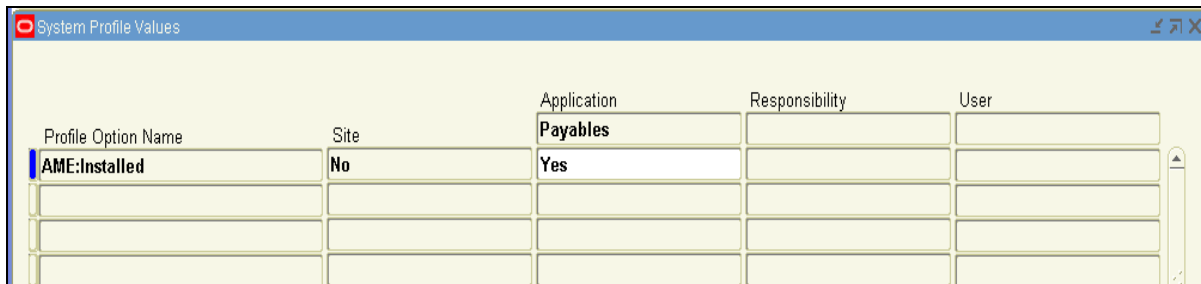
After the roles are granted and access is established, additional setup must be completed for each transaction type. The application setup varies for each transaction type. Examples below include AP expenses, AP invoices and Purchase Requisitions. Refer to the user guide for the specific application to determine the setup for other transaction types.

AME Setup for Specific Applications

The setup at the individual application levels vary by application. Refer to the product guide for individual applications for specific requirements. The setups for AP, PO Requisitions and HR iRecruitment Vacancy Approval are shown in the following sections to illustrate the wide variance of setup requirements. Remember that while there is a specific development group at Oracle for the AME product, the integration of individual products is the responsibility of the application product group that is being integrated.

AP Expense Reports Setup

The profile option *AME:Installed* must be set at the application level for *Payables* specifically to use AME for payables expenses. This profile option change can be made under the System Administrator responsibility in the applications.

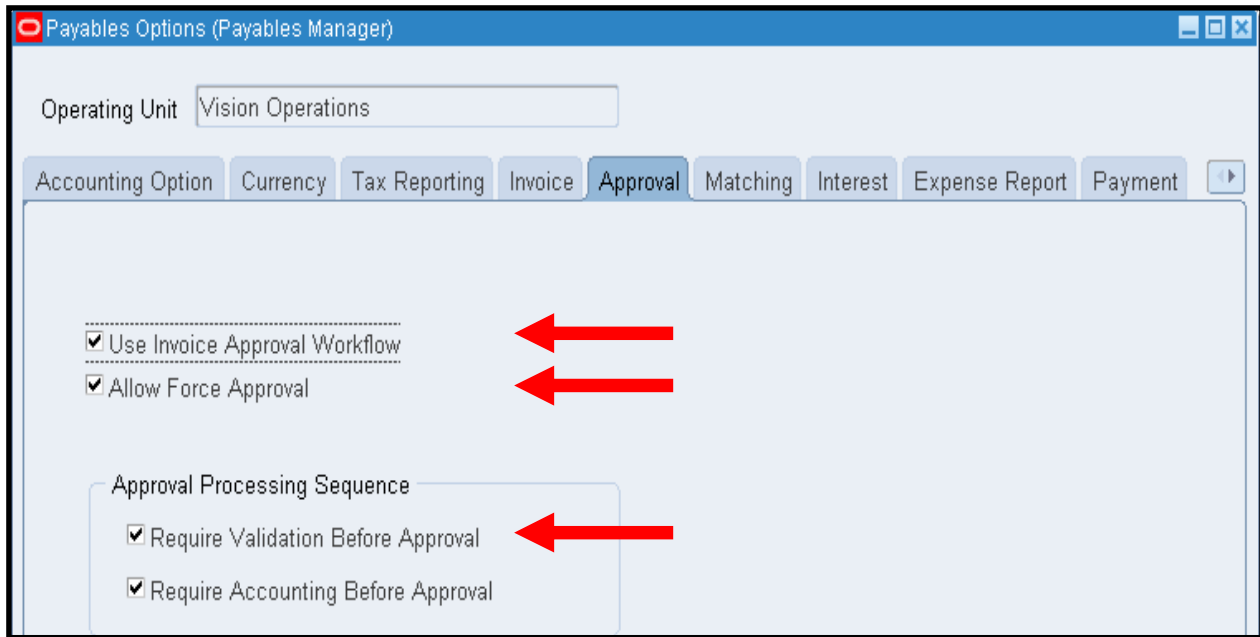


Profile Option Name	Site	Application	Responsibility	User
AME:Installed	No	Payables	Yes	

AP Invoices Setup

In addition to the setup steps that must be followed in the AME application, there are some additional steps that must be done in the Payables application to enable AP invoice approval routing. These setups are done from one form within the application module, the Payables Options form.

¹ This role comes inherit with access to all of the components needed to develop AME rules. This includes attributes, conditions, rules and the testing workbench.

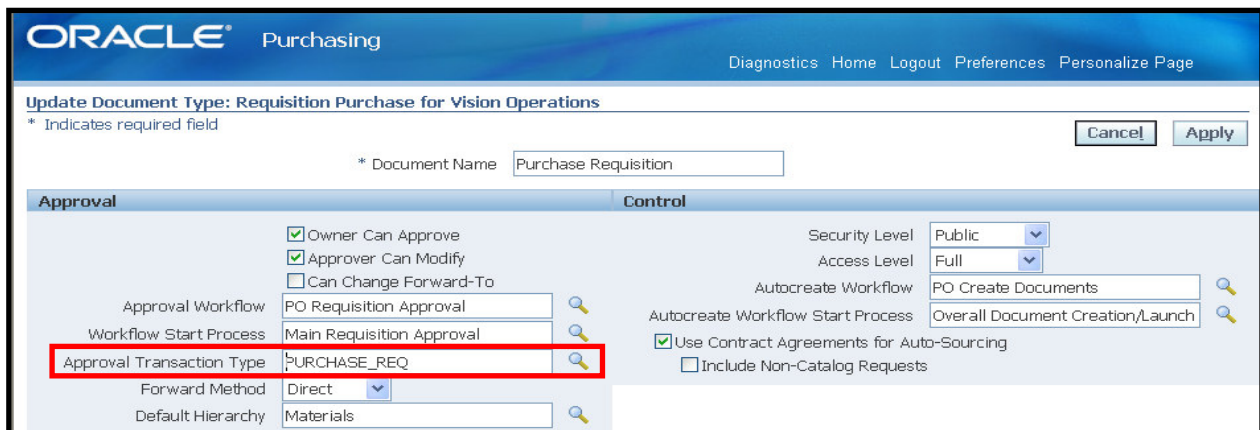


The Payable Options form is typically located under the Payables manager or equivalent responsibility in the applications. There are three options on this form that dictate how Invoice Approval is facilitated in the Payables applications.

The first option *Use Invoice Approval Workflow* is the primary option because it informs the Payables application to force all invoices to go through the invoice approval workflow. As mentioned previously, when this option is enabled, all invoices are set to *Required* and must initially fall into the workflow cycle. The next option is *Allow Force Approval*. This allows a user to automatically set the approval status of an invoice to *Approved*, which allows an invoice to be automatically approved without having to go through the workflow cycle. The last option, *Require Validation Before Approval* requires that an invoice be fully validated before it can be placed in the workflow approval cycle.

PO Requisition Setup

Use of AME for purchase requisitions is controlled via document types. Select the document type and enter the approval transaction type that applies for the document type. For purchase requisitions, enter PURCHASE_REQ. For internal requisitions, enter INTERNAL_REQ.



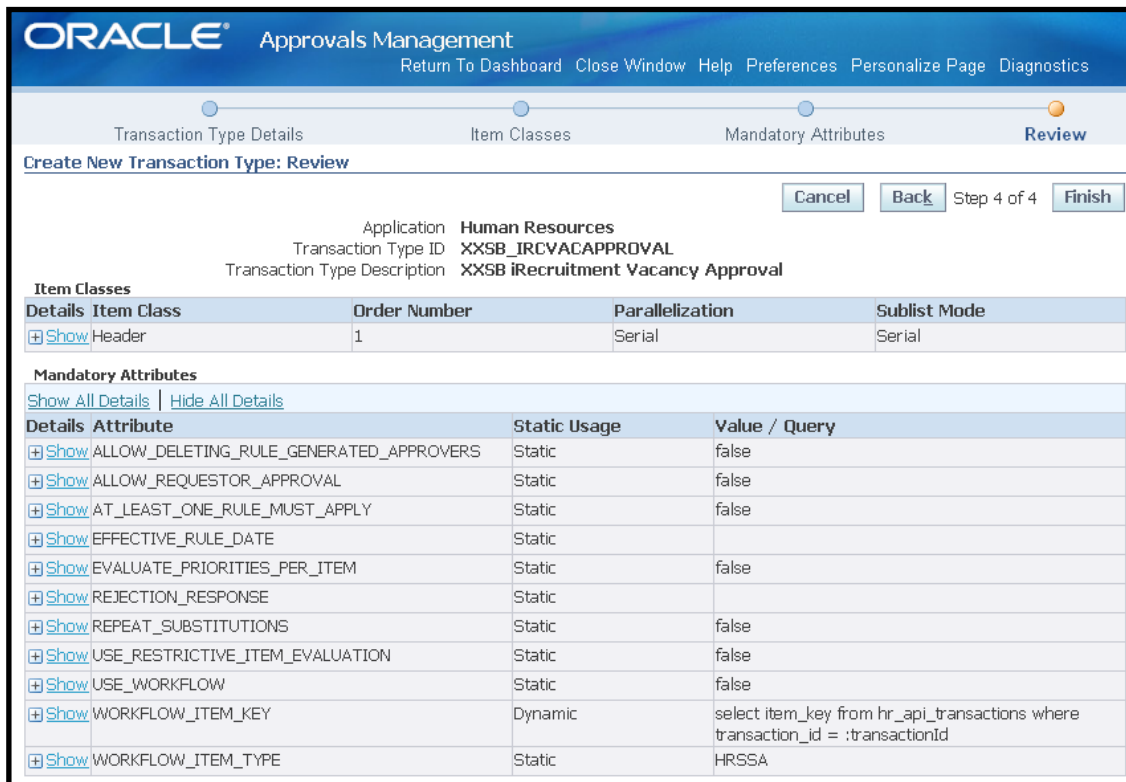
iRecruitment Vacancy Approval Setup

Vacancy Approval is one of the few exceptions where creation of a new transaction type is expected. The iRecruitment Vacancy Approval transaction type is delivered with seeded rules so unless the seeded rules work for your organization, you will want to follow the instructions in the user guide for iRecruitment to create a new transaction type for this approval process.

In the Approvals Management Administrator responsibility, click the *Create Transaction Type* button.

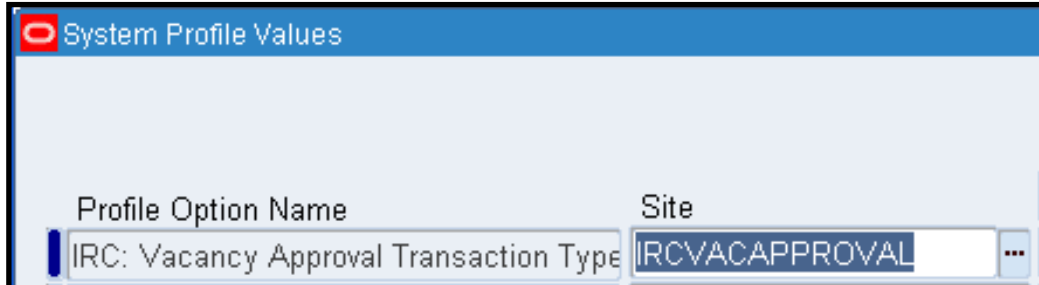


Create the transaction type with a unique key and name. Accept the default values on page 2. On page 3 of the creation process flow, update the workflow item key to *select item_key from hr_api_transactions where transaction_id = :transactionId*. Update the Workflow Item Type to HRSSA. The summary page for the transaction type creation process is shown below.

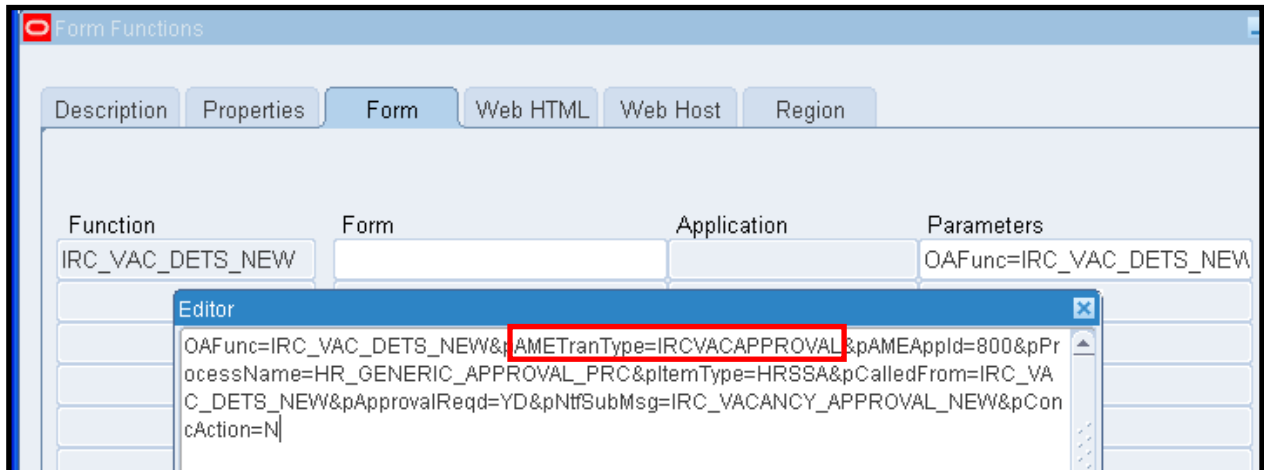


Then go to the Approvals Management Business Analyst responsibility and create the AME components as needed.

To complete the setup, there are two steps to identify the new transaction type name. First, set the profile option IRC: Vacancy Approval Transaction Type to the internal name of the transaction type to be used for Vacancy Approval.



Next, go to Application Developer Responsibility → Application → Function, search for IRC_VAC_DET_S_NET, go to the Form tab and change the value for the AMETranType to the new transaction type name in the parameters field as shown below.



Defining Business Case Scenarios

In the current version of the AME application (AME.B), the Approvals Management Business Analyst role provides access to the Business Analyst Dashboard.

ORACLE Approvals Management

Home Logout Preferences Personalize Page

Business Analyst Dashboard

This page displays the transaction types to which you have access and the rules attached to these transaction types.

Transaction Types

Name	Application	Rules	Test	Setup
AMW Audit Procedure Approval	Internal Controls Manager			
AMW Control Approval	Internal Controls Manager			
AMW Risk Approval	Internal Controls Manager			
BOM ERES Bill of Materials Creation	Bills of Material			
BOM ERES Bill of Materials Update	Bills of Material			

Recently Updated Rules

This table displays the rules whose definition and usages has changed.

Since number of days

Name	Rule Type	Transaction Type	Action	Date	User	Update
SB Sales Tax Group Approval	Pre List Approver Group	Payables Invoice Approval	Created	25-Feb-2007	GERALDJONES	
SB Tax Group Approver Rule	Pre List Approver Group	Payables Invoice Approval	Deleted	25-Feb-2007	GERALDJONES	
SB Tax Group Approver Rule	Pre List Approver Group	Payables Invoice Approval	Created	25-Feb-2007	GERALDJONES	
SB Million \$ Invoice	List Creation	Payables Invoice Approval	Created	25-Feb-2007	GERALDJONES	
SB Million \$ Invoice	List Creation	Payables Invoice Approval	Deleted	25-Feb-2007	GERALDJONES	

Approval Process Setup

- Select the transaction type**
Payables Invoice Approval
- Define the components**
Approvals Management uses these components within the approval rules.
 - Attributes**
Define attributes to fetch business facts for a specific transaction.
 - Conditions**
Define conditions to evaluate attributes within rules. If all conditions in a rule are true then the rule is active for the transaction.
 - Action Types**
Enable action types to specify the action to take if a rule is active for a transaction.
 - Approver Groups**
Define approver groups to contain approvers who are usually members of peer groups such as payroll or expenses department.
- Define the approval rules**
Approval Rules determine the approvers or FYI notification recipients required for a business transaction.
 - Rules**
- Test Workbench**
Define test cases or test real transactions to verify the approval setup, rules and associated approvers.
 - Test Workbench**

[Future Active Rules](#)

[Future Inactive Rules](#)

The dashboard can be thought of as a 'birds eye view' of the AME application. Along with displaying an overview of the various transaction types that are currently defined, the dashboard also displays any rules that have recently been defined, updated or deleted along with any rules that are slated to become active at a future date. More importantly, the dashboard provides links to all of the setup components required when defining new business case rules in AME, including attributes, conditions, approver groups and rules.

Whenever a business user begins the process of defining rules that represent organization business cases, it is important to have an understanding of the transaction type of which business rules will be based. As part of this understanding, a user should determine two important elements of the transaction type:

- What does the transaction type's transaction id represent?
- How does the transaction type determine the requester of a transaction?

The answer to the first question would require some research (i.e. MetaLink, Application specific guides, etc.) to discover what value in a particular transaction is used to represent the transaction id. In the case of the *Payables Invoice Approval* transaction type, the invoice_id in AP_INVOICES_ALL is used as the transaction type. The importance of knowing the value of the transaction id lies in the fact that the dynamic attributes use the transaction id as part of the WHERE clause of the SQL statement used to retrieve their value. Remember, an attribute must return a single value. In the case of invoice transaction, using the invoice_id will ensure that a single value will be retrieved.

As far as determining the requester initiating a transaction, there is a required dynamic attribute defined for most if not all transaction types that contains the logic to retrieve this value. The required attribute is

TRANSACTION_REQUESTOR_PERSON_ID. In the Payables Invoice Approval transaction type, the value of this attribute is retrieved by the following SELECT statement:

```
select requester_id
from ap_invoices_all
where invoice_id = :transactionId
```

This means that the person populated in the requester field on the invoice header in Payables will be flagged as the initiator of a transaction. Any approver lists that are built from an invoice transaction will begin using the requester id as the basis.

For each of the following business cases, the paper assumes the HR supervisor hierarchy is used as the basis for building approval lists. Additionally, there is the assumption that only one currency (USD) is used.

Business Case # 1: Require Approvals up 1 level from invoice requester for any invoice between \$500 and \$10,000. This applies only to invoices that are not matched to purchase orders and not related to projects.

For this case, the following components need to be defined:

Attributes: No new attributes are required. All necessary attributes are seeded.

Condition: PO_MATCHED_INVOICE = N

Condition: PO_MATCHED_INVOICE in (N)	
Condition Type	Ordinary
Item Class	Header
Attribute Name	PO_MATCHED_INVOICE
Data Type	String
Condition	PO_MATCHED_INVOICE in (N)

Condition: PROJECT_RELATED_INVOICE = N

Condition: PROJECT_RELATED_INVOICE in (N)	
Condition Type	Ordinary
Item Class	Header
Attribute Name	PROJECT_RELATED_INVOICE
Data Type	String
Condition	PROJECT_RELATED_INVOICE in (N)

Condition: SUPPLIER_INVOICE_AMOUNT >= \$500 and <= \$10,000

Condition: SUPPLIER_INVOICE_AMOUNT is greater than or equal to 500 and less than or equal t...	
Condition Type	Ordinary
Item Class	Header
Attribute Name	SUPPLIER_INVOICE_AMOUNT
Data Type	Currency
Condition	SUPPLIER_INVOICE_AMOUNT is greater than or equal to 500 and less than or equal to 10000,USD

Rule: Supervisor Rule

If PO_MATCHED_INVOICE = 'N' and PROJECT_RELATED_INVOICE = 'N' and SUPPLIER_INVOICE_AMOUNT >= \$500 and <= \$10,000, then require approval up to the first superior.

Rules		Test Workbench	Setup
Rules >			
Rule: Simple Invoice between \$500 and \$10,000 Requires Manager Approval			
Name	Simple Invoice between \$500 and \$10,000 Requires Manager Approval	End Date	31-Dec-4712
Rule Type	List Creation		
Item Class	Header		
Start Date	28-Sep-2004		
Conditions			
Condition	Condition Type	Item Class	
PO_MATCHED_INVOICE in (N)	Ordinary	Header	
PROJECT_RELATED_INVOICE in (N)	Ordinary	Header	
SUPPLIER_INVOICE_AMOUNT is greater than or equal to 500 and less than or equal to 10000,USD	Ordinary	Header	
TRANSACTION_SET_OF_BOOKS_ID is equal to 1	Ordinary	Header	
Actions			
Action Type	Action		
Supervisory level	Require approvals up to the first superior.		

Business Case # 2: For project related invoices, require pre-approval from key project members with the 'Project Manager' and 'Director' roles. Also require 1 supervisor level approval for invoices between \$500 and \$10,000 and 2 supervisor levels for invoices over \$10,000.

In Oracle projects, project roles are defined and assigned to projects. An employee is identified for these key roles.

For this case, the following components need to be defined:

Attributes: No new attributes are required. All necessary attributes are seeded.

Condition: SUPPLIER_INVOICE_AMOUNT >= \$500 and <= \$10,000 (defined in Business Case 1)

Condition: SUPPLIER_INVOICE_AMOUNT > \$10,000

Condition Type	Ordinary
Item Class	Header
Attribute Name	SUPPLIER_INVOICE_AMOUNT
Data Type	Currency
Condition	SUPPLIER_INVOICE_AMOUNT is greater than 10000,USD

Approver Group: SB_PROJECT_MANAGER (shown below) and SB_PROJECT_DIRECTOR

This is a dynamic approver group to select the project managers associated with all projects on an invoice distribution line. Multiple project managers may be returned if different projects are referenced.

The approver group for the project director would be created similarly with the appropriate project_role_meaning.

Important to remember is that an approver group allows an organization to setup hierarchies that include specific individuals required to be included in an approval list. Additional people can be added or removed as needed.

Rule: If the invoice is not matched to a purchase order and the invoice is project related and the invoice is between \$500 and \$10,000, then require pre-approval from the project manager and project director and require approval from one superior.

Condition	Condition Type	Item Class	Remove
TRANSACTION_SET_OF_BOOKS_ID is equal to 1	Ordinary	Header	
PO_MATCHED_INVOICE in (N)	Ordinary	Header	
SUPPLIER_INVOICE_AMOUNT is greater than or equal to 500 and less than or equal to 10000,USD	Ordinary	Header	
PROJECT_RELATED_INVOICE in (Y)	Ordinary	Header	

Action Type	Action	Remove
Supervisory level	Require approvals up to the first superior.	
pre-chain-of-authority approvals	Require pre-approval from SB Project Manager	
pre-chain-of-authority approvals	Require pre-approval from SB Project Director	

The definition of this rule really demonstrates the flexibility of the AME application when defining complex or unique approvals. There are a couple of items worth noting regarding this setup.

The first of which is the rule type associated with the rule which is a *combination* rule type. This important to note because a combination rule type allows a rule to include different action types that may use different approval types as discussed earlier in the document. For this rule, a combination rule type was necessary to allow the pre-approval groups to be notified (SB Project Manager and SB Project Director) prior to the notifying the immediate supervisor of the invoice requester.

The other item worth noting is that rule can use as many conditions as necessary to satisfy the most complex or unique approval requirements of an organization.

Business Case # 3: For all requisitions require 1 supervisor level approval for requisitions between \$0 and \$25,000, 2 supervisor level approvals for requisitions between \$25,000 and \$50,000, 3 supervisor level approvals for requisitions between \$50,000 and \$100,000, 4 supervisor level approvals for requisitions over \$100,000. Vehicle purchases require director level or above approval and IT purchases require pre-approval from the IT Manager.

For this case, the following components need to be defined:

Attributes: No new attributes are required. All necessary attributes are seeded.

Conditions:

- REQUISITION_TOTAL >= \$0 and <= \$25,000
- REQUISITION_TOTAL > \$25,000 and <= \$50,000
- REQUISITION_TOTAL > \$50,000 and <= \$100,000
- REQUISITION_TOTAL > \$100,000

Condition ▲	Item Class	Condition Type
REQUISITION_TOTAL is greater than 100000,USD	Header	Ordinary
REQUISITION_TOTAL is greater than 25000 and less than or equal to 50000,USD	Header	Ordinary
REQUISITION_TOTAL is greater than 50000 and less than or equal to 100000,USD	Header	Ordinary
REQUISITION_TOTAL is greater than or equal to 0 and less than or equal to 25000,USD	Header	Ordinary

Conditions:

- ITEM_CATEGORY = AUTOMOTIVE.VEHICLE
- ITEM_CATEGORY in (COMPUTER.PC, COMPUTER.SERVER)

Condition ▲	Item Class	Condition Type
ITEM_CATEGORY in (425.00, AUTOMOTIVE.VEHICLE)	Line Item	Ordinary
ITEM_CATEGORY in (COMPUTER.PC, COMPUTER.SERVER)	Line Item	Ordinary

Approver Group: IT_APPROVER

The approver group below is a static approver group. Approvers can be people defined in HR or users defined in the Create user form. Users that are not linked to employees are the only records available as FND User approver types. If a user is linked to an employee, that user will only appear in the HR People list of values. This allows for significant flexibility in AME. If there is a requirement to send an approval request to a distribution list, use of a user with an email address that is a distribution list is one option.

Approval Group Return to Top					
Details	Name	Description	Type	Order Number	Voting Regime
Hide	IT Approver	Approver required for all IT Purchases	Static	1	Serial
Members					
Name	Approver Type		Order Number		
Stockman, Pat	HR People		1		

Rules 1-4: – Supervisor level rules are created for each requisition total range as shown in the example below.

Rules	Test Workbench	Setup
Rules >		
Rule: VO Supervisor Rule 1, 0-25K		
Name	VO Supervisor Rule 1, 0-25K	End Date 31-Dec-4712
Rule Type	List Creation	
Item Class	Header	
Start Date	21-Jul-2004	
Conditions		
Condition	Condition Type	Item Class
REQUISITION_TOTAL is greater than or equal to 0 and less than or equal to 25000,USD	Ordinary	Header
Actions		
Action Type	Action	
Supervisory level	Require approvals up to the first superior.	

Rule 5: A pre-approval rule as shown below is created to require pre-approval based on any line item with the category COMPUTER.PC or COMPUTER.SERVER.

Rules Test Workbench Setup		
Rules >		
Rule: Computer Purchases Rule		
Name	Computer Purchases Rule	End Date 31-Dec-4712
Rule Type	Pre List Approver Group	
Item Class	Header	
Start Date	01-Nov-2007	
Conditions		
Condition	Condition Type	Item Class
ITEM_CATEGORY in (COMPUTER.PC, COMPUTER.SERVER)	Ordinary	Line Item
Actions		
Action Type	Action	
pre-chain-of-authority approvals	Require pre-approval from IT Approver	

Rule 6: The final rule for the purchase requisition case is for the job level approval. Job levels are based on the approval authority for the employee's job. This is set via the Job form available in Purchasing at Setup → Personnel → Jobs. The example below shows the job level for a director and the rule requiring at least a job level of 5 or higher for the purchase of a vehicle. AME will include all approvers in the supervisor chain until an employee is found with a level 5 or higher.

Job	
Job Group	Vision Corporation
Name	DIR300.Director
Dates	01-JAN-1987 -
Approval Authority	5
<input type="checkbox"/> Additional Employment Rights <input type="checkbox"/> Benchmark Job	
Benchmark Job Name	
Further Information	Exempt.Professionals...Officials and M [30]
<input type="button" value="Evaluation"/> <input type="button" value="Requirements"/> <input type="button" value="Valid Grades"/>	

Name Vehicle End Date 31-Dec-4712		
Rule Type	List Creation	
Item Class	Header	
Start Date	01-Nov-2007	
Conditions		
Condition	Condition Type	Item Class
ITEM_CATEGORY in (AUTOMOTIVE.VEHICLE)	Ordinary	Line Item
Actions		
Action Type	Action	
absolute job level	Require approvals up to at least level 5.	
<input type="button" value="Cancel"/> <input type="button" value="Back"/> Step 4 of 4 <input type="button" value="Finish"/>		

Business Case # 4: For all vacancy approvals, the following approvals are required:

- Send an FYI Notification to the hiring manager
- Require approval from the hiring manager's VP
 - If there is no VP then it should go to the Division Controller
- Require approval from the Controller for this hiring manager
- Require approval from the Hiring Manager's SVP which is based on the people group
 - There may not be a SVP. If not, go to the next level
- Require approval from the Hiring Manager's EVP or CFO, COO or President also identified by the people group

Custom approver groups were used to determine the various VPs and controllers needed to approve vacancies. Otherwise, the setup of rules was similar to those shown in earlier examples. Key custom approver groups are shown in the following table.

Group Name	Desc	Usage Type	Order #	Voting	Dynamic Query
XXSB_IRC_HIRING_MANAGER	This group will contain the personId for the hiring manager of the vacancy	Dynamic	1	Serial	select 'person_id':llXXSB_irc_notifications_pkg.get_irc_manager_id(:transactionId) from dual
XXSB_IRC_HIRING_MGR_VP	Hiring manager's Vice President	Dynamic	3	Serial	SELECT 'person_id':llXXSB_IRC_NOTIFICATIONS_PKG.GET_IRC_VP_PERSON_ID(:transactionId) FROM DUAL
XXSB_IRC_DIVISION_CONTROLLER	Get the division controllers of the hiring manager	Dynamic	4	Serial	SELECT 'user_id':lluser_id FROM FND_USER WHERE USER_NAME = (SELECT DECODE(PAF.ASS_ATTRIBUTES5, '001', 'IRC_DIV_001', '002', 'IRC_DIV_002', '003', 'IRC_DIV_003', '005', 'IRC_DIV_005','IRC_DIV_003') FROM PER_ALL_ASSIGNMENTS_F PAF WHERE PERSON_ID = XXSB_IRC_NOTIFICATIONS_PKG.get_irc_manager_id(:transactionId)) AND SYSDATE BETWEEN PAF.EFFECTIVE_START_DATE AND PAF.EFFECTIVE_END_DATE AND paf.PRIMARY_FLAG = 'Y')
XXSB_IRC_FINAL_VAC_APPROVAL	This group will contain the final approver list for vacancy approval	Dynamic	5	Serial	SELECT person_id FROM ((SELECT 'person_id:' ppf.person_id person_id ,ppf.full_name ,ind_level ,ppg.group_name FROM per_all_assignments_f paf2 ,per_all_people_f ppf ,pay_people_groups ppg ,(SELECT paf1.person_id ,LEVEL ind_level FROM per_all_assignments_f paf1 START WITH paf1.person_id = XXSB_irc_notifications_pkg.get_irc_manager_id (:transactionid) AND paf1.primary_flag = 'Y' AND SYSDATE BETWEEN paf1.effective_start_date AND paf1.effective_end_date CONNECT BY PRIOR paf1.supervisor_id = paf1.person_id AND paf1.primary_flag = 'Y' AND LEVEL > 1 AND SYSDATE BETWEEN paf1.effective_start_date AND paf1.effective_end_date AND LEVEL > 1) loc WHERE paf2.person_id = loc.person_id AND SYSDATE BETWEEN paf2.effective_start_date AND paf2.effective_end_date AND paf2.people_group_id = ppg.people_group_id AND SYSDATE BETWEEN ppf.effective_start_date

					<pre> AND paf2.person_id = ppf.person_id AND ppg.group_name = 'Senior Vice President') UNION ALL (SELECT * FROM (SELECT 'person_id:' ppf.person_id person_id ,ppf.full_name ,ind_level ,ppg.group_name FROM per_all_assignments_f paf2 ,per_all_people_f ppf ,pay_people_groups ppg ,(SELECT paf1.person_id ,LEVEL ind_level FROM per_all_assignments_f paf1 START WITH paf1.person_id = XXSB_irc_notifications_pkg.get_irc_manager_id (:transactionid) AND paf1.primary_flag = 'Y' AND SYSDATE BETWEEN paf1.effective_start_date AND paf1.effective_end_date CONNECT BY PRIOR paf1.supervisor_id = paf1.person_id AND paf1.primary_flag = 'Y' AND LEVEL > 1 AND SYSDATE BETWEEN paf1.effective_start_date AND paf1.effective_end_date AND LEVEL > 1) loc WHERE paf2.person_id = loc.person_id AND SYSDATE BETWEEN paf2.effective_start_date AND paf2.effective_end_date AND paf2.people_group_id = ppg.people_group_id AND SYSDATE BETWEEN ppf.effective_start_date AND ppf.effective_end_date AND paf2.person_id = ppf.person_id AND ppg.group_name IN ('Executive Vice President' ,'COO' ,'CFO' ,'President') ORDER BY loc.ind_level ASC) WHERE ROWNUM = 1)) </pre>
--	--	--	--	--	--

AME Testing Workbench

One of the very powerful features of the AME application is the Testing Workbench. The workbench provides the ability to test the business rules that have been defined in AME against test or *real* transactions. This allows you to preview the results of your AME definitions to verify certain aspects such as:

- Are attribute values, particularly custom attributes retrieving values correctly?
- Does the invoice satisfy the appropriate rule?
- Is the proper approver chain(s) being generated for the transaction based on the rule chosen?

The testing workbench can be accessed from the AME Dashboard. The AME Dashboard can be found under the Approvals Management Business Analyst role discussed earlier in the document.

ORACLE Approvals Management

Home Logout Preferences Personalize Page

Business Analyst Dashboard

This page displays the transaction types to which you have access and the rules attached to these transaction types.

Transaction Types

Name	Application	Rules	Test	Setup
AMW Audit Procedure Approval	Internal Controls Manager			
AMW Control Approval	Internal Controls Manager			
AMW Risk Approval	Internal Controls Manager			
BOM ERES Bill of Materials Creation	Bills of Material			
BOM ERES Bill of Materials Update	Bills of Material			

Recently Updated Rules

This table displays the rules whose definition and usages has changed.
Since number of days:

Name	Rule Type	Transaction Type	Action	Date	User	Update
SB Sales Tax Group Approval	Pre List Approver Group	Payables Invoice Approval	Created	25-Feb-2007	GERALDJONES	
SB Tax Group Approver Rule	Pre List Approver Group	Payables Invoice Approval	Deleted	25-Feb-2007	GERALDJONES	
SB Tax Group Approver Rule	Pre List Approver Group	Payables Invoice Approval	Created	25-Feb-2007	GERALDJONES	
SB Million \$ Invoice	List Creation	Payables Invoice Approval	Created	25-Feb-2007	GERALDJONES	
SB Million \$ Invoice	List Creation	Payables Invoice Approval	Deleted	25-Feb-2007	GERALDJONES	

Future Active Rules

Future Inactive Rules

Approval Process Setup

- Select the transaction type**
Payables Invoice Approval
- Define the components**
Approvals Management uses these components within the approval rules.
[Attributes](#)
Define attributes to fetch business facts for a specific transaction.
[Conditions](#)
Define conditions to evaluate attributes within rules. If all conditions in a rule are true then the rule is active for the transaction.
[Action Types](#)
Enable action types to specify the action to take if a rule is active for a transaction.
[Approver Groups](#)
Define approver groups to contain approvers who are usually members of peer groups such as payroll or expenses department.
- Define the approval rules**
Approval Rules determine the approvers or FYI notification recipients required for a business transaction.
[Rules](#)
- Test Workbench**
Define test cases or test real transactions to verify the approval setup, rules and associated approvers.
Test Workbench

The first step in using the Test Workbench involves defining a new test case in AME. Defining a test case is simple as it involves supplying a name for the test case and description (optional).

ORACLE Approvals Management
Transaction Type: Payables Invoice Approval

Return to Dashboard Home Logout Preferences Help Personalize Page

Rules Test Workbench Setup

Test Workbench >
Create Test Case

Cancel Save for Later (1) Run Test Case (2)

Name: OAUG Test Case
Description: Test Case to demonstrate AME setup at OAUG 2007

Expand All | Collapse All

Focus Test Data	Value	Add/Delete Item
Parameters		

Cancel Save for Later (1) Run Test Case (2)

Rules | Test Workbench | Setup | Return to Dashboard | Home | Logout | Preferences | Help | Personalize Page

Copyright (c) 2006, Oracle. All rights reserved. [About this Page](#) [Privacy Statement](#)

After entering a name for the test case and description, choose the *Save for Later* button to save the test case definition. Although a *Run Test Case* button is available at this point, it is best to save the definition to the database first and then execute a test case after receiving the confirmation page below.

ORACLE Approvals Management
Transaction Type: Payables Invoice Approval

Return to Dashboard Home Logout Preferences Help Personalize Page

Rules Test Workbench Setup

Confirmation
Test case OAUG Test Case has been saved successfully.

Test Workbench

Run Real Transaction Test (1)

Test Cases

Select Test Case: Run Copy Create

Select Name	Description	Last Updated	Update	Delete
OAUG Test Case	Test Case to demonstrate AME setup at OAUG 2007	27-Feb-2007		

Run Real Transaction Test (1)

Getting Started

- To create a new test case for the current transaction type, click Create.
- To view the applicable rules and approver list for an existing test case, select the test case and click Run.
- To duplicate an existing test case that you can modify later, select the test case and click Copy.
- To view the applicable rules and approvers for a real transaction, click Run Real Transaction Test.

Rules | Test Workbench | Setup | Return to Dashboard | Home | Logout | Preferences | Help | Personalize Page

Copyright (c) 2006, Oracle. All rights reserved. [About this Page](#) [Privacy Statement](#)

The best approach to demonstrating the AME Test Workbench is by entering a new transaction in the application – in this case, an invoice in the Payables application. Then return to AME and execute a test against this invoice. The invoice will be created to test the second business case rule defined earlier in the document. Our test case will allow us to verify whether our business case rule has been defined properly and if the approver list is built correctly by AME.

The first step to test our business case is to create an invoice in Oracle Payables. The invoice has been created as follows.

Invoice Workbench (Payables Manager)										
Batch Control Total					Batch Actual Total					
Sbehn										
Type	PO Number	Project	Requester	Trading Pa	Supplier Num	Supplier Site	Invoice Date	Invoice Num	Invoice	Invoice Amount
Standard			Behn, Susan D	Dell Comp	5092	DELL	31-OCT-2007	OOW_AME_CASE1	USD	2,600.00

Num	Type	Amount	GL Date	Account	Project	PO Number
1	Item	2,600.00	31-OCT-2007	01-110-7530-0000-000		

The invoice has been created with a total amount of \$2,600. This invoice is not matched to a PO and is not related to a project. Based on our rule in case 1, we would expect AME to require approval of Susan Behn's supervisor who is Pat Stock. In order to use this transaction for an AME test, the invoice Id must be retrieved via SQL.

Now that an invoice has been created, we can execute a test from the workbench to see if our AME components have been defined correctly and produce the results we expect. To execute a test against this invoice, navigate to the testing workbench. From the workbench, choose a test case against which the testing will be conducted. Choose the *Run Real Transaction Test* button.

ORACLE Approvals Management

Transaction Type: Payables Invoice Approval

Rules | **Test Workbench** | Setup

Test Workbench

Run Real Transaction Test (1)

Test Cases

Select Test Case: Run | Copy | Create

Select Name	Description	Last Updated	Update	Delete
QAUG Test Case	Test Case to demonstrate AME setup at OAUG 2007	27-Feb-2007		

Run Real Transaction Test (1)

Test Workbench

The Workbench enables you to define test cases to verify your approval setup. Approval Management treats a test case as a real transaction, it evaluates your defined rule and generates an approver list for your review.

Getting Started

- To create a new test case for the current transaction type, click Create.
- To view the applicable rules and approver list for an existing test case, select the test case and click Run.
- To duplicate an existing test case that you can modify later, select the test case and click Copy.
- To view the applicable rules and approvers for a real transaction, click Run Real Transaction Test.

Rules | Test Workbench | Setup | Return to Dashboard | Home | Logout | Preferences | Help | Personalize Page

Copyright (c) 2006, Oracle. All rights reserved. [About this Page](#) [Privacy Statement](#)

The next screen prompts the user for the transaction id which AME uses to evaluate previously defined rules and generates an approver list. This transaction id as mentioned previously is the invoice_id from AP_INVOICES_ALL

Real Transaction Test

* Indicates required field

Cancel | Save for Later (1) | Run Test Case (2)

* Transaction Id Go

TIP Transaction ID is the ID of a real transaction which AME uses to evaluate rules and generate approvers.

It is important to note that upon entering a transaction id, you must choose the *Go* button which retrieves information about your transaction. In particular, it retrieves the values of all attributes that have been defined for the current transaction type. The values of the some of the attributes are shown below to demonstrate how AME retrieves and displays the values of the attributes. These values are consistent with the data entered for your invoice transaction. Expand All and/or scroll down to see additional attributes and values.

Focus Transaction Data	Value
Parameters	
Mandatory Attributes	
Header Attributes	
Line Item Attributes	
Cost Center Attributes	
Project Code Attributes	
Distribution Attributes	
Distribution ID: 1369669	
SUPPLIER_INVOICE_DISTRIBUTION_LAST_UPDATE_DATE	31-Oct-2007
SUPPLIER_INVOICE_DISTRIBUTION_PA_ADDITION_FLAG	E
SUPPLIER_INVOICE_DISTRIBUTION_PA_QUANTITY	
SUPPLIER_INVOICE_DISTRIBUTION_PO_BUYER_EMP_NUM	
SUPPLIER_INVOICE_DISTRIBUTION_PO_MATCHED	N
SUPPLIER_INVOICE_DISTRIBUTION_PO_REQUESTER_EMP_NUM	
SUPPLIER_INVOICE_DISTRIBUTION_REFERENCE_1	
SUPPLIER_INVOICE_DISTRIBUTION_REFERENCE_2	
SUPPLIER_INV_DISTRIBUTION_EXPENDITURE_ITEM_DATE	
SUPPLIER_INV_DISTRIBUTION_LINE_TYPE_LOCKUP_CODE	ITEM
SUPPLIER_INV_DIST_PROJECT_ACCOUNTING_CONTEXT	
JOB_LEVEL_NON_DEFAULT_STARTING_POINT_PERSON_ID	
SUPERVISORY_NON_DEFAULT_STARTING_POINT_PERSON_ID	
SUPPLIER_INVOICE_DISTRIBUTION_ASSETS_TRACKING_FLAG	N
SUPPLIER_INVOICE_DISTRIBUTION_COST_CENTER	110
SUPPLIER_INVOICE_DISTRIBUTION_CREATION_DATE	31-Oct-2007
SUPPLIER_INVOICE_DISTRIBUTION_DESCRIPTION	
SUPPLIER_INVOICE_DISTRIBUTION_EXPENDITURE_TYPE	
SUPPLIER_INVOICE_DISTRIBUTION_GL_ACCOUNT	7530
SUPPLIER_INVOICE_DISTRIBUTION_GL_BALANCING	01

Buttons: [Cancel](#) | [Save for Later \(1\)](#) | [Run Test Case \(2\)](#)

After reviewing the values retrieved for the various attributes of the transaction, choose the *Run Test Case* button to execute and evaluate the rules and action defined for the transaction.

Rules | Test Workbench | Setup

Test Workbench > Real Transaction Test >

Run Real Transaction

To review the process stages that Approvals Management took to arrive at the applicable rules and approver list, click View Approval Process Stages.

Transaction ID: 155959

[View Approval Process Stages \(1\)](#)

Applicable Rules		Class	Rule Type	Category
Show	Simple Invoice between \$500 and \$10,000 Requires Manager Approval	Header	List Creation	Approver

Final Processed Approver List				
Details	Order Number	Approver Type	Approver	Status
Show		HR People	Stock, Pat	

[Return to Test Workbench](#) | [View Approval Process Stages \(1\)](#)

Based on the results of the test, it appears that the business test case was defined properly. The Supervisor Rule was applied to this transaction because the invoice had an invoice amount between \$500 and \$10,000. Additionally, the approval list has been built correctly. The approver is Susan's supervisor.

The View Approval Process Stages button provides additional details about how the final list of approvers was determined. For example, if multiple rules apply and approver appears more than once in the approver list, the additional occurrences of that approver will be suppressed. Also, expanding the + sign for the rule or approver list will provide details on the conditions that were met and the approver list.

As previously stated, the Testing Workbench can be a very useful during the process of implementing and developing AME rules for invoice approval routing. Having the ability to evaluate and see the results of your AME setups using *real* transaction prior to implementation in a production environment is quite valuable.

Conclusion

It was the intent of this paper to provide the reader with enough high level understanding of AME functionality and how organizations can use AME to control their approval requirements for invoice approval routing in Oracle Payables and Oracle Purchasing. As with any other Oracle application, mastery of the application comes through practice and experimentation. Hopefully, the paper has demonstrated how thorough planning of business case rules and further understanding of AME can allow business users to develop their most unique or complex approval requirements in this powerful application.

Bibliography

1. *Oracle Approvals Management Implementation Guide Rel 11i* Part No. B25324-01, MetaLink Note 336901.1
2. *Oracle Approvals Management Developers Guide Release AME.B*, MetaLink Note 289927.1
3. *How to Implement the Invoice Approval Workflow*, MetaLink Note 269057.1
4. *Oracle Payables Users Guide 11i* Part No. A81180-07
5. *Oracle Approvals Management Not Enabled? What does it take to Enabled It?*, MetaLink Note 413300.1
6. *About Oracle Approvals Management Minipack B*, MetaLink Note 336004.1