

# Customers as Approvers with AME with TCA Hierarchy

Ajoy A. Devadawson  
Oracle Corporation

## Introduction

Oracle Corporation's Consulting Division, in which I am an employee, has been in the forefront in implementing enterprise applications for clients of various sizes. Currently, the Consulting Division emphasizes on consultants getting trained on Project Management to streamline project implementation to avoid missed deadlines and budget overruns.

The Project organization on which this presentation is loosely based on is a huge service-oriented organization, whose employees are mostly on the frontlines to do their business. Wherever there is a disaster in any part of the world, people and materials have to be deployed at short notice. To support the deployment, a new instance has to be cloned and snapped-off from the Enterprise database instance in US. A new instance is needed due to security and network bandwidth issues, as opposed to having a global instance.

A Consultant, to move away from command and control hierarchy and to decentralize the operations, designed the Enterprise Architecture. The decentralized Organizations/Units are highly mobile and will have Support Units, which take care of a Unit's supply, maintenance and service needs. Most of the existing Supply Chain applications will be consolidated into the new Oracle EBusiness Suite. Instead of relying too much on supply chain applications, the frontline Units satisfy their demand through Service Requests routed to back-end modules like Inventory, OM, Purchasing, EAM etc.

The CRM Service used to be a case history application and now has evolved into a serious ERP application. A typical user of CRM Service application user would be a phone or cable company whose field service personnel need details about customers when they come for installation or service of customer site installed products. The application empowers the field service personnel with all the info they need about customers like their installed items, service contracts, item ownership, case history, counters for gas or electric meters, credit information etc. In this project organization, the CRM Service application is being used in disaster like situations for all its operational needs. A need could be a supply of toilet paper to an earth moving equipment. Or, a pickup truck needs some maintenance. All these needs have to be met instantly, as the frontline Units do not have the resources to handle them, as they have to be mobile with limited on-hand resources.

Oracle Workflow delivers a complete workflow management system that supports business process based integration. Its technology enables modeling, automation and continuous improvement of business processes, routing information of any type according to user-defined business rules. The purpose of Approvals Management (AME) is to define approval rules that determine the approval processes for Oracle EBusiness Application. It provides a great deal of flexibility in creating approval processes, usually without writing any custom code.

## Trading Community Architecture (TCA)

The TCA is a data structure, originally meant for Customers in Receivables application. It has been continuously enhanced to store Vendors, Employees and Partners. The R12 has more integration with TCA, through Purchasing, Payables and Cash Management (bank reconciliation) modules.

Customer Data Management (CDM) is a complete package solution that allows companies to create a single, enterprise view of their customer base, by consolidating and synchronizing customer data from heterogeneous systems into a central/operational data store. It consolidates modules like Oracle Customers Online (OCO), Customer Data Librarian (CDL) and Customer Data Hub (CDH). By consolidating all customer data into TCA, organizations can use the functionality of CDM to standardize, cleanse and enrich their customer records.

Party is an entity you could enter into a business relationship with; even the relationship is a Party. Both Organizations and People could be Parties. The understanding of relationships between Organizations and People is essential to have a 360-degree view of your organization's business operations. For example, John Doe could be a contact in your organization's AP department, but he also could be buying your product for himself. So, there is a need for properly establishing relationships between your organization, customer's organization and customer employees. The customer-installed items can have relationships with TCA parties as Accountable and Responsible officers.

Dun & Bradstreet (D&B) organization provides services to validate and enrich customer data in TCA. For example, D&B information is integrated with party records to maintain accurate information that can be used to evaluate credit risks.

ELocations Spatial Data Integration allows for retrieving spatial information from Oracle eLocations and storing the longitude and latitude data for addresses in the TCA Registry.

## **Integrated View of Customer Model**

The Customer information stored in TCA is the centerpiece of Customer Model, in which TCA interfaces with many application suites. Marketing suite leverages the repository of customer information, to better target and personalize campaigns and refine them in real time, with powerful analytical tools. Sales suite simplifies the sales process by providing visibility into the sales cycle, helping companies plan and manage effective selling activities from getting a sales leads to quotes. Order Management is used to capture orders from multiple channels, price orders, check product availability, schedule fulfillment, plan shipments, ship delivers and track shipments. Receivables module uses TCA for business objects like Invoices, Payments and Credits. The BOM module uses customer information for Product configuration and related activities.

CRM Service suite contains modules that handle any Service needs of the customer. Oracle Installed Base is an item instance life cycle tracking application that facilitates enterprise-wide life cycle item management and tracking capability. Once an item is marked as trackable in Oracle Inventory, a real-world instance of the item is created in Install Base and any subsequent change to that instance will be recorded. A Service Request is created when a customer calls for assistance. A task is created when a Service Request needs to be acted upon, like if a technician must visit the customer site. Service Request can be entered through iSupport, TeleService, Field Service and Mobile Field Service modules in response to any customer needs. Service Contracts module provides complete contract execution solution to manage warranties, usage, subscription services, complex service agreements, entitlements etc. Field Service module provides service history relating to several areas of interest, including customer, site, product, problem or resolution. Any charges for parts or service made by the field service technicians can be billed through the Charges tab of Service Request form.

## **Relationships – Hierarchical & Lateral**

A Party Relationship is a party's role in the context of another party. Relationships can be either seeded or user defined and it can be hierarchical or lateral.

In Figure 1, the organization – “M29002 HQTRS CO CLR 37” has relationships with 2 other organizations. The organization – “M29001 CLR 37 3D” is hierarchically above and “M29003 SVC CO CLR 37” is laterally related to “M29002 HQTRS CO CLR 37”.

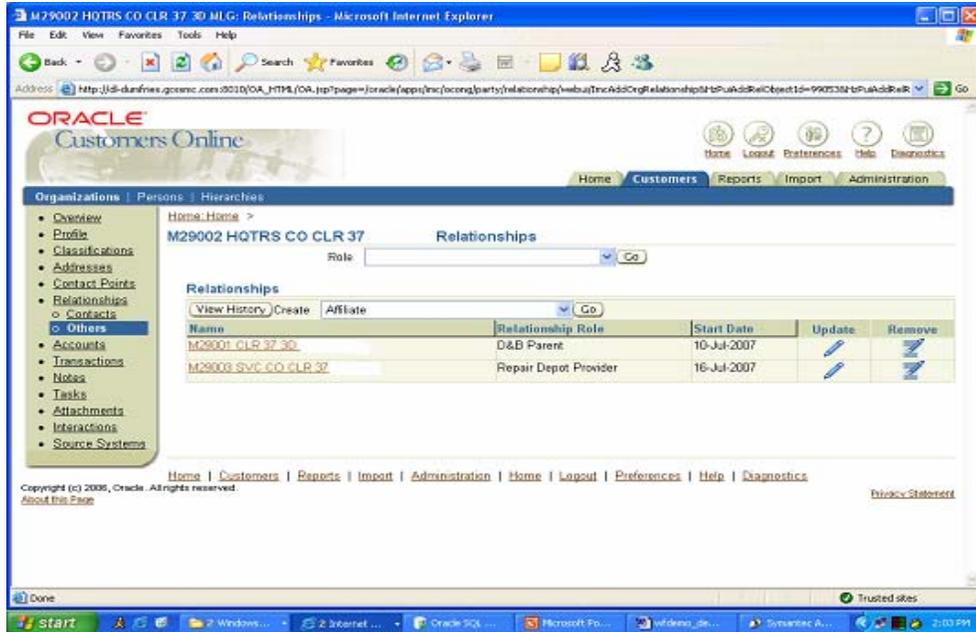
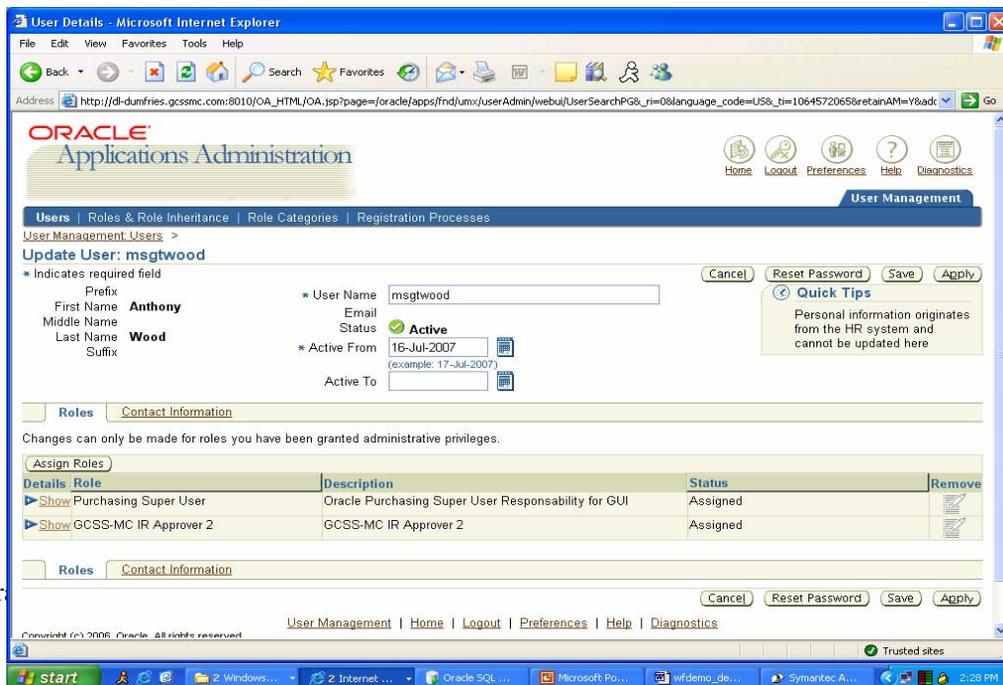


Figure 1. Hierarchical and Lateral Relationship for an Organization

## User Management (UMX) Roles

Oracle User Management (UMX) enables organizations to define administrative functions and manage users based on specific requirements such as job role or geographic location. Role Based Access Control (RBAC) can control access to application data, through a grant on UMX Role.

In Figure 2, the user Anthony Wood has been assigned a seeded Role, ‘Purchasing Super User’ and a custom role, ‘GCSS-MC IR Approver 2’. The custom role was specifically created for workflow approval eligibility.

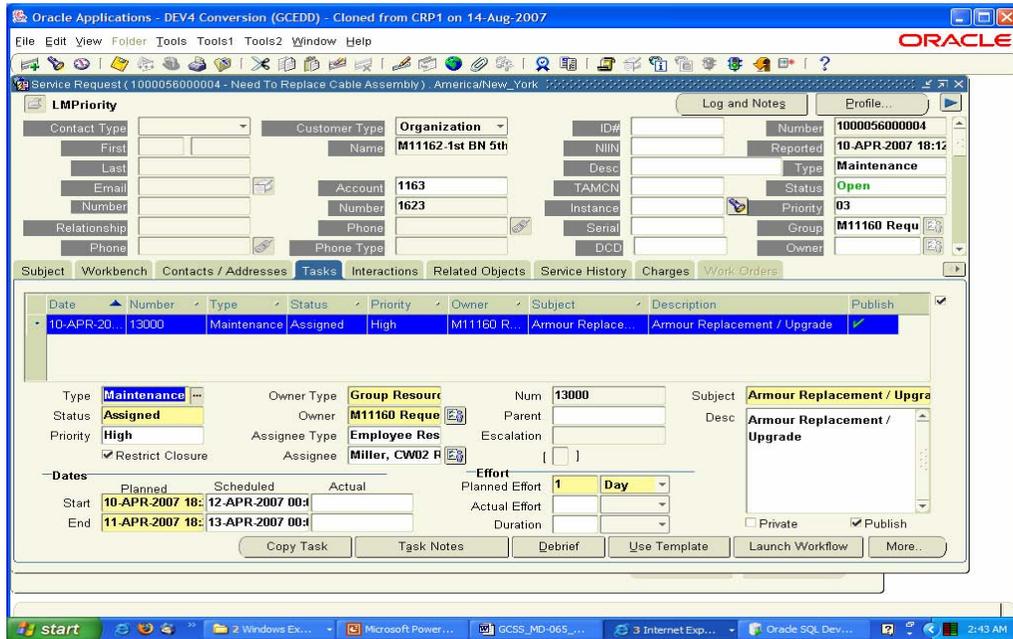


**Figure 2. User Management Roles**

## Service Request – Field Service

Service Requests (SR) can be created as a result of customer calls or planned work generated by a preventive maintenance program. It can be created through Field Service, Mobile Field Service and Tele Service modules. Customers can create Service Requests themselves, through the iSupport self-service module and go through knowledge-based solutions.

In Figure 3, a SR is created in Field Service module. The SR is of type Maintenance and a Task also is created and assigned to technician to do a particular task. A SR can also be created against a customer's installed Item Instance.



**Figure 3. Service Request – Field Service**

## Overview of AME

The Approvals Management (AME) module is a self-service Web application through which business rules can be defined for approving transactions in Oracle Applications. It comes installed with Oracle EBusiness suite. The approvals are primarily done through Rules and Rules are built through building blocks of AME, which are Attributes, Conditions, Action Types and Approval Groups. AME is a module, which is integrated into EBusiness suite, and it comes with pre-built links from the application modules.

Through AME's approval engine, the approval list can be dynamically generated, taking into account variables like termination of an approver, an employee's eligibility conditions etc. For every workflow approval, AME maintains its own history of approval information.

## How does AME Work?

AME works in coordination with Oracle Workflow. A workflow, which needs business rules for its approval, can call the AME transaction engine from inside the workflow. The call can be made from a workflow function, using a PL/SQL call. For example, AME\_API is a PL/SQL package, which comes with set of routines to utilize various AME features. This API can be used to get the next approver. Once the next approver information is received, a notification can be sent to that person to approve or reject. If the approver approves, check if that person is the final approver. If so, end the workflow. If not, find the next approver and send the notification. If the approver rejects the notification, end the workflow.

## AME with TCA/UMX

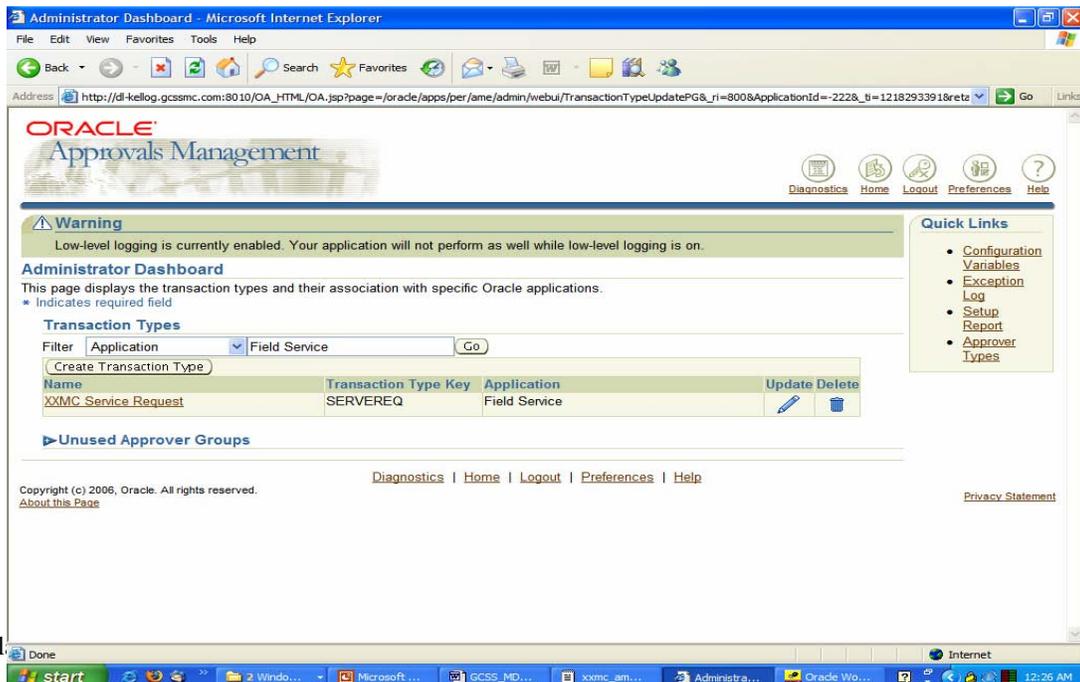
AME works with Trading Community Architecture (TCA) and User Management (UMX) modules to get the customers who will be approvers. The Customer Organizations and their Employees are stored in TCA as Parties, with relationships between employees, between organizations and between organization and employees. The UMX module is used to assign Roles to potential approvers. Roles determine the eligibility to approve a workflow notification. For example, 'SR Approver A' could be a Role, which determines if a person is eligible to approve a Service Request. If this Role is assigned to a party, that person is eligible to get the notification.

The workflow module will call AME, when it is ready to get the next approver. The AME module will dynamically generate the approver list. It builds this approver list, by using AME Conditions and Approver Groups. While it is the normal behavior of AME, the custom Action Handler will make a call to TCA and UMX to get the Customers (Customer's employees) as Approvers.

## AME Administrator Dashboard

Users can access AME Administrator Dashboard with 'Approvals Management Administrator' responsibility. To perform any technical tasks such as setting AME's configuration variables, creating new Transaction Types, the Administrator Dashboard has to be created. An application, which integrates into AME, can divide its transactions into several categories, where each category requires a set of approval rules. Each set of rules is called a Transaction Type.

The Administrator Dashboard is shown in Figure 4, in which a new custom AME Transaction Type has been created. This custom Transaction Type has been created to do approvals for Service Requests created in CRM – Service modules.



## Figure 4. AME Administrator Dashboard

### Components of AME

The AME module has to be configured for every Transaction Type that workflow has to use. The configuration can be done through the 'Approvals Management Business Analyst' role. The components to be built to implement a business are explained below.

'Attributes' are business variables with single value for a transaction. For example, "UMX\_ROLE" could be an attribute, which contains the value for UMX Role of an approver. This attribute could be used in a Rule to get the approver from TCA, with a particular UMX Role. The attribute value could be a constant or generated dynamically through a SQL statement.

'Condition' is a statement, which is either true or false, which could be used in an if-then statement. For example, a Condition could be like this, "UMX\_ROLE in ('XXMC SR Approver A')".

An 'Action' is an instruction to AME to modify a transaction's approval in the manner you specify. Every Action belongs to an 'Action Type' and Action Type is a collection of Actions having similar functionality. AME includes pre-defined Action Types, which enable it to ascend commonly used organizational hierarchies, HR Supervisor, HR Position, HR Job Level etc. An Action Type used in this presentation is 'Approval-group chain of authority', which is a general purpose one used for implementing any organizational approval hierarchy.

An 'Action Handler' is a program behind Action Type to build Approver Groups. As there is a need for building the Customer employees in TCA as Approvers, a custom Action Handler had to be written to browse through the Customer hierarchy to pick the approvers.

'Approver Group' can be either ordered set of one or more approvers or it can be a dynamically generated list, evaluated at rule evaluation time. If an Approver Group is a static one, then it has to be generated through a SQL or a database function. See the Approver Group screen for the SQL statement used to construct the Approver Group.

Creating 'Rules' is the backbone of AME implementation. Rules associate one or more Conditions to an approval in an if-then statement. Rarely will an organization's business rules match any pre-defined rules for a transaction type. So, the Rules have to be created even there is a seeded Transaction Type for a Workflow. For example, a Rule could be written like below.

```
If UMX_ROLE is equal to XXMC Request Manager then  
  Require Approval from Request Manager Rule.
```

### AME Attribute

In Figure 5, AME Attribute, 'UMX\_ROLE' is created with a Select statement, which calls a database function to return its value. The bind variable :transactionId represents the transaction key for every transaction, which can be used inside AME for any PL/SQL calls.

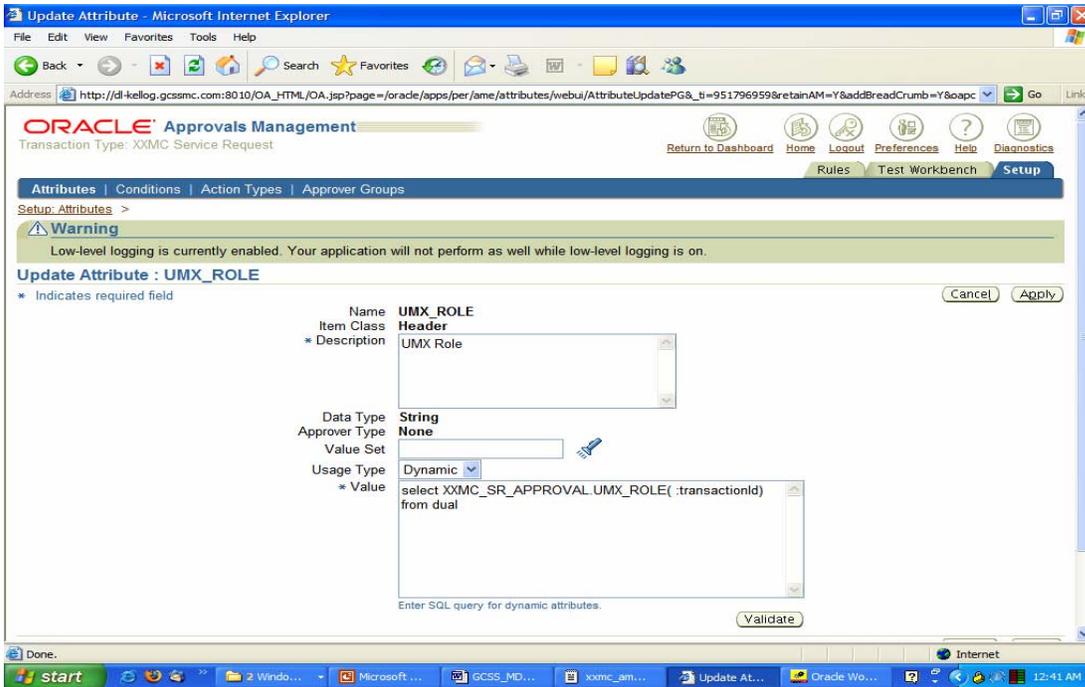


Figure 5. AME Attribute

## AME Condition

In Figure 6, the AME condition is checking for a specific Role – ‘XXMC SR Approver A’. It also shows which Rule is using this condition.

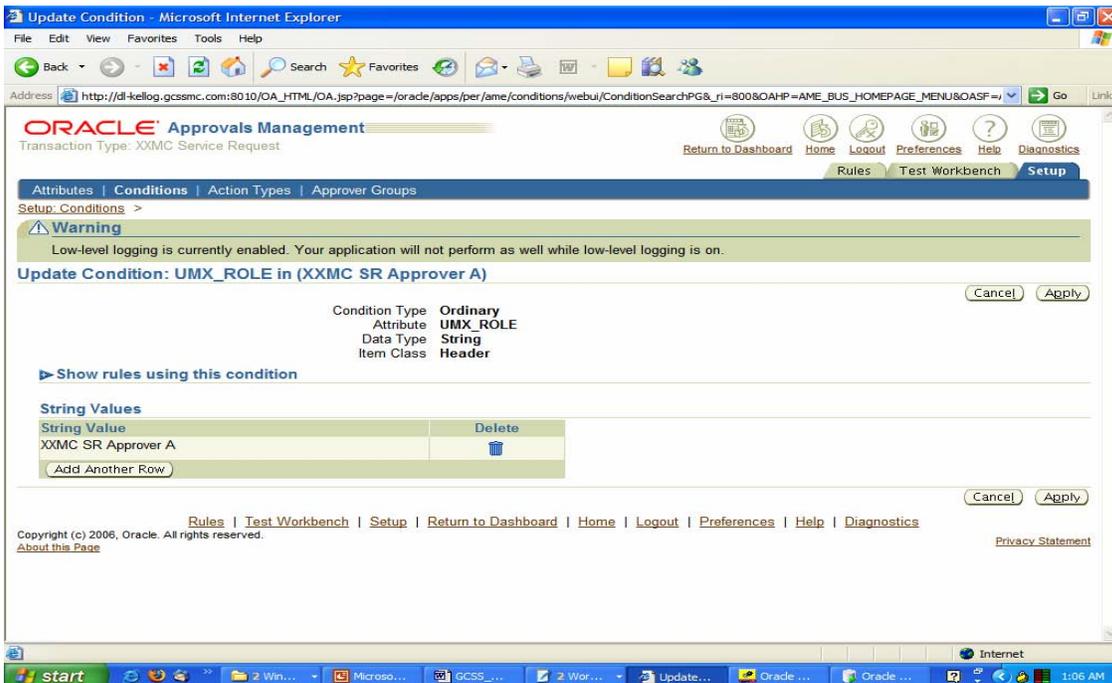


Figure 6. AME Condition

## AME Action Type

In Figure 7, the Action Type 'approval-group chain of authority' is being used in this Transaction Type. It also lists the Actions which are part of this Action Type.

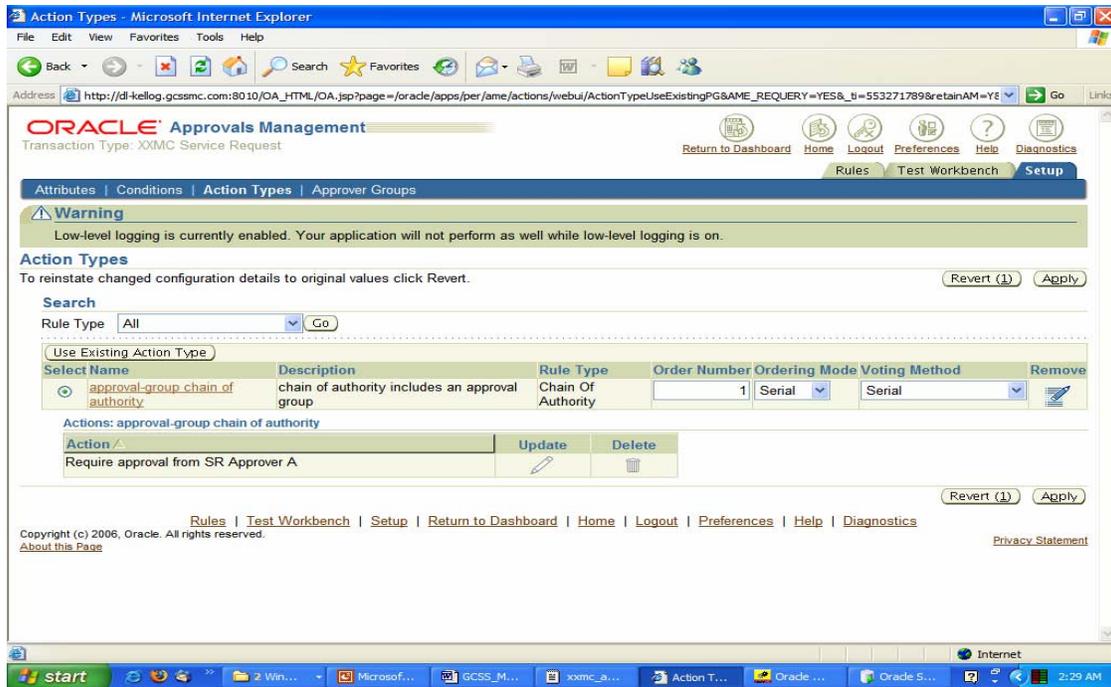
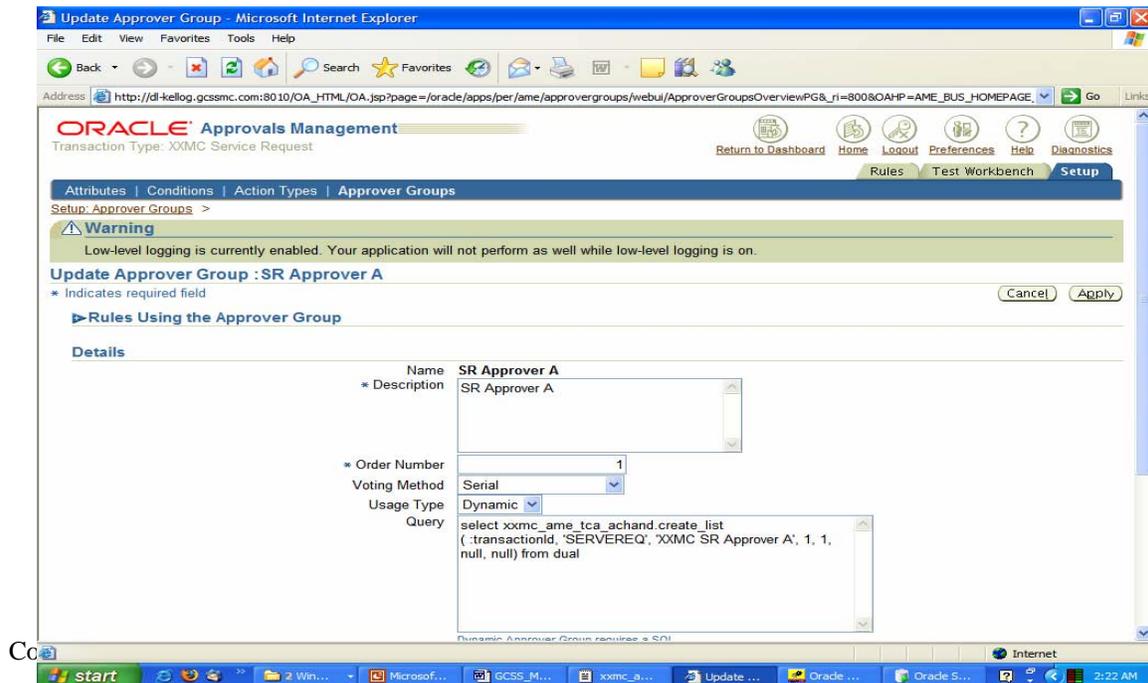


Figure 7. AME Action Type

## AME Approver Group

In Figure 8, an Approver Group, 'SR Approver A' has been created, which will be dynamically generated during rule evaluation time. It is calling the custom PL/SQL package to build the approval list. Some of the parameters used are Workflow Item Type, Role Name of the approver, level of hierarchy, no. of approvers etc.



**Figure 8. AME Approver Group**

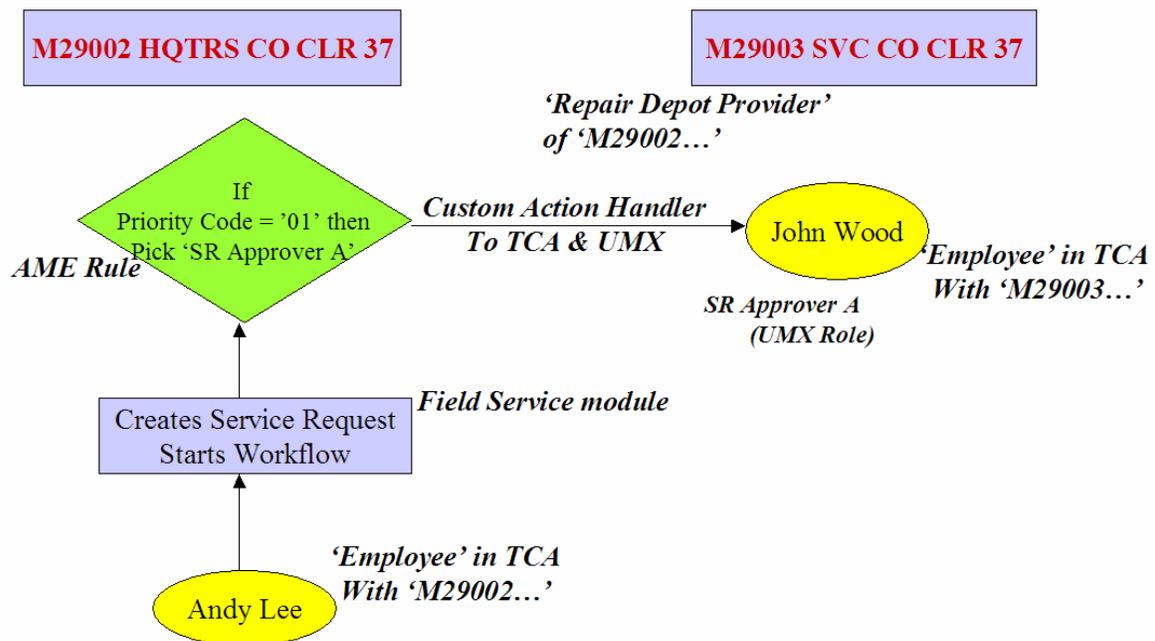
### Custom Action Handler

The Custom Action Handler called 'AME TCA Action Handler' ( Program Name: XXMC\_AME\_TCA\_ACHAND) is the backbone of implementing Customers as Approvers in your business rules for approval transactions. This Action Handler will be used by the Action Type to ascend the TCA Organization/People hierarchy to get the next approver. It has the following features:

- Builds Approver Groups of people attached to organizations in TCA.
- Recognizes Roles (UMX module) assigned to people.
- Callable from within AME module through Action Type, thereby integrating TCA and UMX modules.
- Can climb TCA hierarchy to find next approver who could be in requestor's organization or in a higher level organization.
- Can find another approver who is in another organization attached to requestor's organization through a relationship.

### Approval Structure

The Figure 9 represents the Approval Structure in which an employee of 'M29002...' creates a Service Request in Field Service module and kickstarts the workflow. The workflow calls the rules engine in AME and evaluates the rule. The rule calls the Custom Action Handler to go and pick the approver who has a particular Role and belongs to an organization. These 2 organizations are related by the 'Repair Depot Provider' relationship. The 'M29003...' is the Repair Depot Provider of 'M29002...'. Based on the AME Rule, parameters specified in the custom Action Handler, TCA Relationship and UMX Roles, an approver will be picked.



**Figure 9. Approval Structure**

## Action Handler Logic

The Action Handler written to get the Customer Approvers is a PL/SQL program to be called through the AME Action Type. The Action Handler has the following logic.

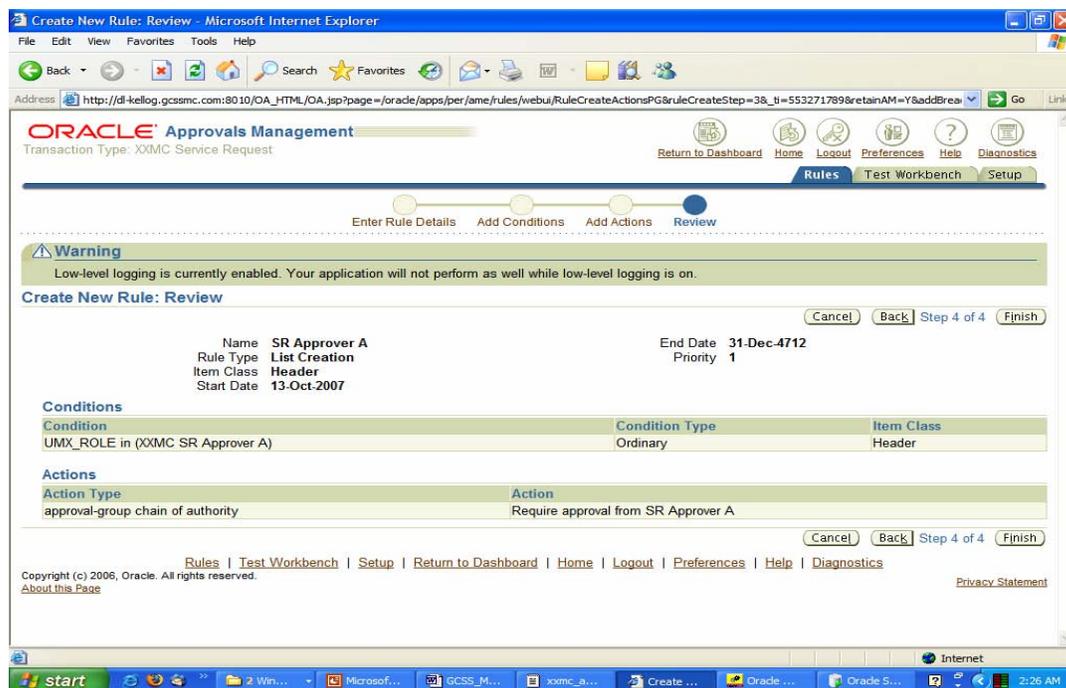
- Get the Organization of document creator from TCA, based on employee/employer relationship. In the above example in Figure 9, 'Andy Lee' is an employee of 'M29002...' organization, who creates a Service Request in the Field Service module.
- Find approvers in that Organization with proper UMX Roles. The Roles have to be assigned for the approvers, to make them eligible. The Roles could be created for different approval needs. For example, a Role 'XXMC SR Approver A' could be created for Priority Code '04' and another Role could be created for Priority Code '03' etc. Or, you could create Roles for ranges of dollar amount for Purchase Order or a Requisition.
- If approvers are not found in that Organization, go to an Organization which is related to this one. This again is dependent on your approval need. In the Figure 9, the search is on another Organization 'M29003...' who has an approver with an approval Role.
- Get approver from that Organization and send the approver information back to workflow.
- If approver, check for next approver, by call the AME API. For example, AME\_API.GET\_NEXT\_APPROVER could be used to call AME.
- If rejected, the workflow ends.

## AME Rule

In Figure 10, AME Rule is being created, which links a Condition to the Approver Group. It is a simple Rule which states that –

If UMX\_ROLE in (XXMC SR Approver A) then  
It Require approval from SR Approver A.

Even though, the Rule looks simple, there is logic behind the attribute UMX\_ROLE, there is complex logic behind the Approver Group – SR Approver A.



**Figure 10. AME Rule**

## Action Handler SQL Statements

The following sample SQL Statements can be used in the Action Handler PL/SQL package to get the Organization of the document creator and also to get the approver from the Organization. These are only sample SQL statements and so try to use them by modifying according to your environment.

### Get Organization of a Document Creator:

```
select hporg.party_id, hporg.party_name, hp.party_name
into   l_orgpartyid, l_orgparty, l_perparty
from   hz_parties hporg, hz_party_relationships hpr, hz_parties hp
where  hp.party_type      = 'PERSON'
and    hp.person_identifier = p_person_id
and    hp.status          = 'A'
and    hp.party_id       = hpr.subject_id
and    hpr.party_relationship_type = p_emp_relation
and    hpr.status        = 'A'
and    sysdate between hpr.start_date and nvl( hpr.end_date,
g_endoftime)
and    hpr.object_id     = hporg.party_id
and    hporg.status      = 'A'
and    hporg.party_type  = 'ORGANIZATION';
```

### Get Approver from a Organization:

```
select rownum arownum, hpper.person_identifier, hpper.party_name
into   l_rownum, l_person_id, l_perparty
from   hz_parties hpper, umx_role_assignments_v ura, fnd_user fu,
hz_party_relationships hpr
where  hpr.object_id      = p_orgpartyid
and    hpr.party_relationship_type = p_emp_relation
and    hpr.status        = 'A'
and    sysdate between hpr.start_date and nvl( hpr.end_date,
g_endoftime)
and    hpr.subject_id    = hpper.party_id
and    hpper.status      = 'A'
and    hpper.party_type  = 'PERSON'
and    hpper.person_identifier = fu.employee_id
and    sysdate between fu.start_date and nvl( fu.end_date, g_endoftime)
and    fu.user_id        = ura.user_id
and    sysdate between ura.active_from and nvl( ura.active_to,
g_endoftime)
and    ura.display_name  = p_appr_role
and    rownum            <= p_max_approvers;
```

### Get the Organization which is related to another Organization:

```
select hporg.party_id, hporg.party_name
into   l_orgpartyid, l_orgparty
from   hz_parties hporg, hz_party_relationships hpr
where  hpr.subject_id    = p_orgpartyid
and    hpr.party_relationship_type = p_org_relation
and    hpr.status        = 'A'
and    sysdate between hpr.start_date and nvl( hpr.end_date,
g_endoftime)
and    hpr.object_id     = hporg.party_id
```

```
and    hporg.status           = 'A'  
and    hporg.party_type      = 'ORGANIZATION';
```

## Useful References

- Approvals Management Responsibilities and Setup in AME.B – Metalink Note # 434567.1
- FAQ on AME – Note # 431815.1
- AME Developer's / Implementation Guide – Note # 289927.1
- TCA Documentation Resources, R12 – Note # 396778.1
- Oracle TCA Best Practices – Note # 269124.1
- Email: [ajoy.devadawson@oracle.com](mailto:ajoy.devadawson@oracle.com)