Multi Org Access control in Oracle R12 New Release

Syed Imdad Huseny
Team Lead,
Accenture Private Limited,
Chennai, Tamil Nadu, India.
imdadji.i@gmail.com

I.Sumaiya Thaseen
Assistant Professor (Senior), SITE,
VIT University
Vellore, Tamil Nadu, India.
isumaiyathaseen@vit.ac.in

Abstract— **Every organization implements shared services to efficiently process business transactions. These transactions can be accessed, processed and reported for more than one operating units within a single responsibility. MOAC is a new feature that allows user to enter, process data and generate reports from a single responsibility in addition to data security. This paper presents the MOAC functionalities, features, benefits and implementation steps in Oracle R12.This feature has been analyzed and compare with Oracle 11i.**

Keywords- MOAC;Oracle R12;Operating units;Security Profile;

I. INTRODUCTION

MOAC is expanded as Multi-Org Access Control. It means accessing multiple operating units within a single application responsibility. This new Feature in R12 enables companies that have implemented or implementing shared services operating model to efficiently process business transactions by allowing them to access, process and report on data for an unlimited number of operating units within a single applications responsibility. Users are no longer required to switch applications responsibilities when processing transactions for multiple operating units.Multi Org model in Oracle Release 12 has changed from what was there in R11. It is now called known as **Multi Org Access Control**. It allows assigning a node of organization hierarchy or a list of operating units to the responsibility. Thus we can assign multiple operating units to a single responsibility. This is made possible either through Organization Hierarchy or by an Organization List.

Oracle Release12 multi-org model uses Security profiles. This is achieved the by the following steps

- A Security profile will be created in HRMS (Human Resource Management Service) responsibility. This security profile will contain the list of operating units (or legal entities) which the responsibility can access.
- A security profile will be attached to responsibility as profile option.

A. Security Profile

Data security is maintained using "Security profiles" that determine the data access privileges associated to responsibilities granted to a user. Because of this multiple tasks across operating units can be performed without changing responsibilities, the simple case can be best described as diagram in fig 1, where 3 users from three different Operating units required three separate responsibilities to perform the task.

B. Features of MOAC

1. R12 [3] does support a hybrid environment where some users operate in MOAC and others may not.
2. Features unconstrained by multi-org today are still unaffected in R12 MOAC.
3. Upon upgrade it is not essential to implement MOAC however should test for boltons, personalization, customizations and 3rd party apps with some users MOAC enabled.
4. MOAC may require additional internal controls or security.

C. *Benefits of MOAC*

1. Multi-Org Access Control [3] feature allows user to enter, process data and generate reports from a single responsibility.
2. This is achieved by providing the Operating Unit field on the forms/pages and while running the concurrent processes.
3. To Set this feature user should define the security profile containing operating units and set it at MO: Security Profile
4. The MO is set to default the Operating Unit on forms/pages: Default Operating Unit profile

Prior to Release12, Org id was a hidden column. But now all screens that use security profile multi-org will have an enterable field. This can be defaulted to a specific value as per profile settings.

II. MOAC IMPLEMENTATION

In R11, the scenario is as follows:

1. A table is created in PO(Purchase Order) Schema, named PO_HEADERS_ALL
2. A synonym named PO_HEADERS_ALL is created in APPS schema, referring to PO.PO_HEADERS_ALL
3. Create a view PO_HEADERS in APPS schema, as "select * from po_headers_all where org_id=client_info"

In R12 [3],

1. A table is created in PO Schema, named PO_HEADERS_ALL
2. A synonym named PO_HEADERS_ALL is created in APPS schema, referring to PO.PO_HEADERS_ALL
3. Another synonym named PO_HEADERS is created in APPS, referring to PO_HEADERS_ALL
4. A Row Level security is applied to PO_HEADERS, using package function MO_GLOBAL.ORG_SECURITY.
5. This can be double-checked by running SQL select * from all_policies where object_name='PO_HEADERS'.
6. The effect of this policy is that, whenever you access PO_HEADERS, Oracle RLS will dynamically append WHERE CLAUSE similar to below

SELECT * FROM PO_HEADERS WHERE EXISTS (SELECT 1 FROM mo_glob_org_access_tmp oa WHERE oa.organization_id = org_id)

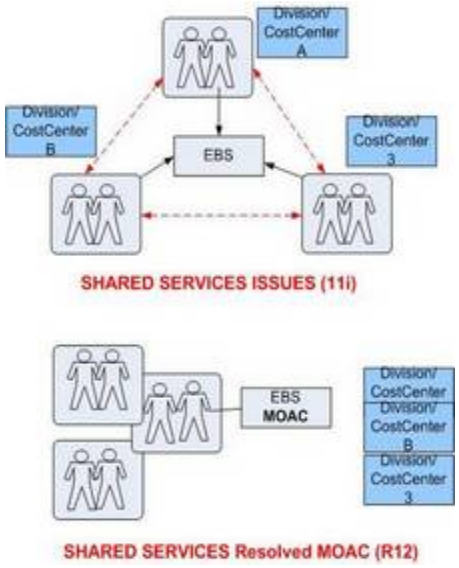


Figure 1: Comparison of shared services in Oracle 11i and R12

The simple predicate using CLIENT_INFO is used for the following case:

- **Access control is not enabled for older releases of the applications, it is not backward compatible:** enable the multiple organizations access control feature cannot be enabled for all products simultaneously because multiple organizations views are shared between products at different levels. For example, if the option chosen is upgrade Payables but choose to keep an earlier version of Purchasing then Payables is access control enabled, but Purchasing is not. Therefore, Purchasing must replace the views it shares with Payables, such as PO_VENDOR_SITES, and PO_HEADERS, with secured synonyms. The secured synonyms must work as before for Purchasing, since it has not upgraded Purchasing therefore Purchasing still relies on CLIENT_INFO.

The simple predicate using current_org_id is used for the following cases:

- **Access control is limited to only one operating unit:** In this case, the access mode is 'S'. An example is when a user can access to only one operating unit through the MO: Security Profile or the MO: Security Profile is not set and the user access depends on MO: Operating Unit.

- **Access control is enabled with access to multiple operating units:** The security profile provides access to multiple operating units, but in the scope of a transaction since the operating unit is controlled, a simple predicate eliminates additional changes to the server and client side code.

The complex predicate is used for these cases:

- Access is enabled and the security profile gives access to multiple Operating Units. The access mode is set to 'M' for this caseUnits

A. Security Policy

1. Security Policies [2] [4] can be applied to database object to control access to specific rows and columns in the object

2. Security Policies can be different for each DML action
   a) Select
   b) Insert
   c) Update
   d) Delete

Real world examples:

Consider the following Security Profile and Operating Units

TABLE I. SECURITY PROFILE DETAILS

Security profile	Org id	Org name
62	299	Canada
63	2	US
64	299	North America

Sample Query
Select ORG_ID, count(*) from OE_ORDER_HEADERS group by ORG_ID;

Security Profile = 62 (Canada)
299, 1000
Security Profile = 63 (US)
2, 7000
Security Profile = 64 (North America)
299, 1000
2, 7000

B. *Background of MOAC*

Multiple organizations architecture (Multi-Org) [1] was introduced in Release 10.6 to secure the data by operating unit. In Release 10.7, Oracle added a column ORG_ID to each base table to partition the data by operating units. The partitioned tables are renamed with the suffix, '_ALL', and their corresponding secured views are created in Applications (APPS) schema. The following diagram shows a single organization view in the APPS schema.

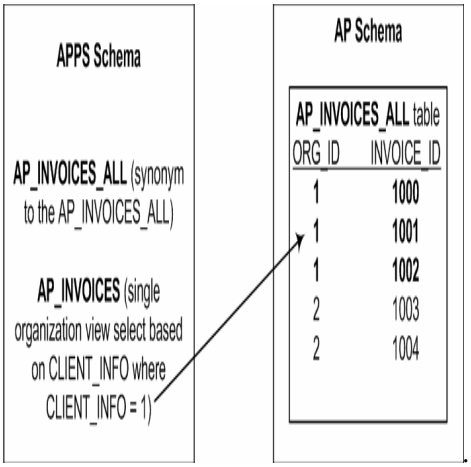


Figure 1.1 : APPS Schema

Multiple organizations views restrict access by filtering records for an operating unit assigned to the application responsibility set for the "MO: Operating Unit" profile option. This profile option value is cached in application context, and is initialized when calling the FND initialization routine. The FND CLIENT_INFO predicate includes all multiple organizations views and SQL statements that require multiple organizations security. The FND_CLIENT_INFO function retrieves the ORG_ID value stored in the application context. This value is valid for a session, unless explicitly changed by the calling procedure.

Use the _ALL table in the SQL statement to retrieve information irrespective of the operating unit. To increase the flexibility and performance in a multiple organizations environment and provide the same level of data security, the DBMS Virtual Private Database (VPD) feature replaces the CLIENT_INFO function.

C. *Virtual Private Database*

This is a security feature of the Oracle Database Server 10G. The Virtual Private Database (VPD) feature allows developers to enforce security by attaching a security policy to database objects such as tables, views and synonyms. It attaches a predicate function to every SQL statement to the objects by applying security policies. When a user directly or indirectly accesses the secure objects, the database rewrites the user's SQL statement to include conditions set by security policy that are visible to the user.

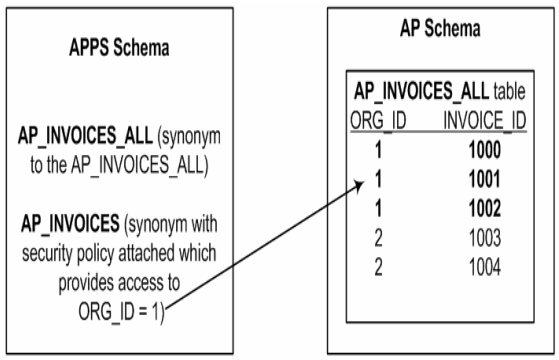
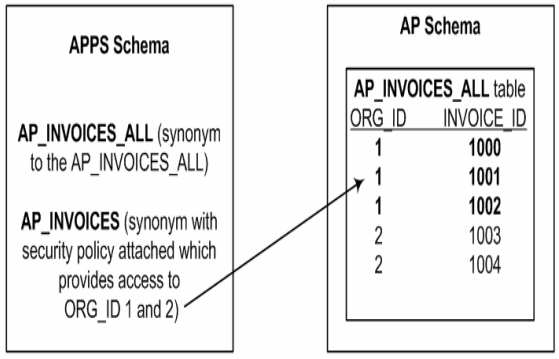


Figure 2: Database Schema - Access to one operating unit



D. *Multiple Organizations Security Policy Predicate*

Synonyms replace single organization views that contain the CLIENT_INFO predicate attached to them. When

installing, it must attach a security policy function to the multiple organizations synonyms. This indicates that the security is in place irrespective of the tools used to access the data.

The security policy function returns different predicate based on the number of accessible operating units. An application context attribute "ACCESS_MODE" is set based on the accessible operating units. Context sensitive security policy is used for multiple organizations access control to minimize the coding impact. The multiple organizations code in previous releases works in the context of only one operating unit. It was not anticipated that multiple organizations access would be supported. A solution to code impact is to change the policy predicate whenever needed. For example, when a form is opened using a responsibility that can access multiple operating units and when an operating unit is selected, the operating unit context is established and there is no need to modify the code that is used for validation from that point onwards, if the synonyms return data for the selected operating unit.

If the access mode is M (Multiple), then the policy predicate issues an EXISTS sub-query to a global temporary table. The global temporary table is a new feature in Oracle 8*i*. The table stores and manipulates data specific to a SESSION or TRANSACTION. If the access_mode is S (Single), then a simple equality predicate is used for performance reasons, since it is cost effective in comparison to the temporary table. An access mode A (All) is incorporated to bypass the security for functionality that needs full table access. If the access mode is not set or is NULL, then a simple predicate that uses the CLIENT_INFO value for ORG_ID is used for the policy predicate to support backward compatibility.

E. Multiple Organizations Initialization.

The profile options MO: Security Profile or MO: Operating Unit [1] [3] populate the multiple organizations global temporary table. The profile option MO: Security Profile takes precedence over MO: Operating Unit. It can be combined under one application menu:

- Products at different levels
- Products that are access control enabled
- Products that are not access control enabled (i.e. in transition)

In such cases, initializing the multiple organizations depends on the application of the calling module and not the application tied to the responsibility, since the profile Option MO: Security Profile must be ignored for products that are not access control enabled or are in the transition phase.

A new table (FND_MO_PRODUCT_INIT) is introduced which contains a value Y for products that are enabled with the multiple organizations access control feature. The multiple organizations initialization API uses the module owner to initialize the temporary table depending on the value for the product in the FND_MO_PRODUCT_INIT table.

TABLE II. FND_MO_PRODUCT_INIT TABLE

Application_Short_Name	Status
AR	Y
JTF	Y
<Custom application short code>	Y or N

.

Legend: Y indicates multiple organizations access control is enabled, N indicates otherwise.

Use the shared services API to register products that are enabled with access control. For example to enable or remove access control for Payables (SQLAP), enter the following code:

To enable access:
FND_MO_PRODUCT_INIT_PKG.register_application('SQLAP','SEED','Y');

To delete your application entry:
FND_MO_PRODUCT_INIT_PKG.remove_application('SQLAP');

The Payables system administrator must then seed a row in the Multiple Organizations table to indicate that Payables is enabled with access control.

F. Some Packages for setting this profile

a. MO_GLOBAL.INIT:

It will check if new Multi Org Security Profile is set, to decide if new Security Profile method will be used. If the new MO security profile is set, then mo_global.init inserts one record, for each Organization in Org Hierarchy, in table mo_glob_org_access_tmp

Calling of mo_global.init :

This package procedure will be called as soon as the login is completed or as soon as the user switches responsibility. Just like FND_GLOBAL.INITIALIZE is called. It is safe to assume that Oracle will invoke MO_GLOBAL.INIT after FND_GLOBAL.INITIALIZE. mo_glob_org_access_tmp table is a global temporary table. Hence after Multi Org is initialized for the user session, the session will have X number of records in table mo_glob_org_access_tmp. X is the number of organizations assigned to MO Security profile.

b. MO_GLOBAL.ORG_SECURITY:

The purpose of Row-Level-Security is to hide certain data[based on some conditions]. RLS does so by appending a where clause to the secured object.

1. MO_GLOBAL.ORG_SECURITY is a function that returns a predicate for the WHERE CLAUSE
2. The where clause will be appended to Table/Synonym/View for which Multi Org Row Level security is enabled.

c. MO_GLOBAL.SET_POLICY_CONTEXT:

This procedure has two parameters

p_access_mode

Pass a value "S" in case if the user wants current session to work against Single ORG_ID.

Pass a value of "M" in case the user wants the current session to work against multiple ORG_ID's.

p_org_id

Only applicable if p_access_mode is passed value of "S".

To set SQL session context:
In R12 we can set SQL session context for multiple Operating Units with the following:

```
BEGIN
execute mo_global.set_org_access(NULL,64,'ONT');
END;
```

Restriction of dbms_client_info.set_client_info:

This will become redundant functionally. Hence mo_global package is used. This package already exists in 11.5.10 instance. And if the package is opened the user can find Row Level Security. If the user has enabled the MultiOrg Security Profile feature also then dbms_client_info.set_client_info will still work, but will produce unexpected results.

G. Subledgers that leverage MOAC

1. Oracle Inventory
2. Oracle Purchasing
3. Oracle Payables
4. Oracle Receivables
5. Oracle Assets
6. Oracle Work in Process
7. Oracle Projects

III. SETUPS AND SCREENSHOTS

A. Create a Define Global Security Profile[3]

Responsibility : Human Resources
Navigation : Security–>Global Profile

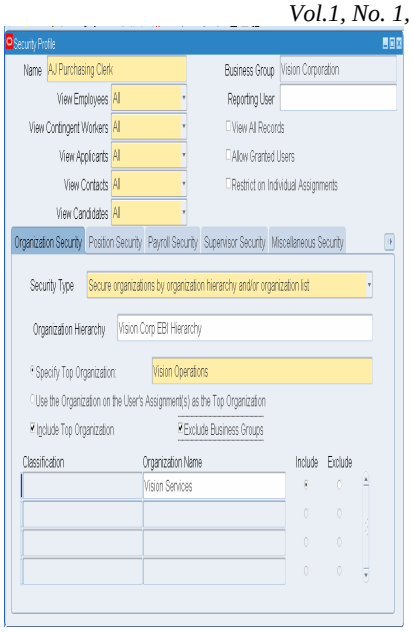


Figure 3 : Global Security Profile

B. Run Security List Maintenance program

Responsibility : Human Resources
Navigation : Security–>Global Profile

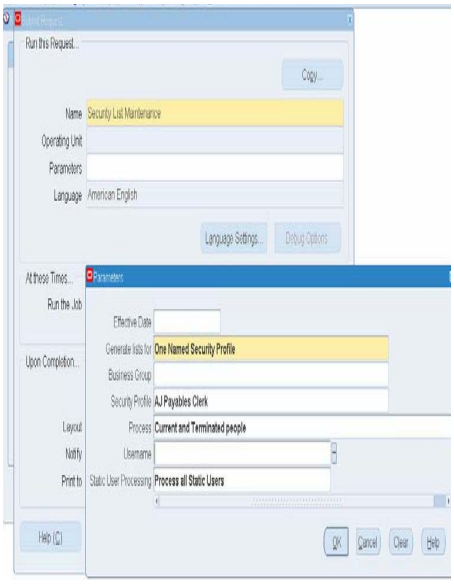


Figure 4 : Security List Maintenance

Run after any change to a security profile
• Generate Lists for
– All Security Profiles
– All Security Profiles in One Named Business Group
– One Named Security Profile

C. Assign the Security Profile to the MO: Security Profile

Responsibility : System Administrator
Navigation : Profile->System

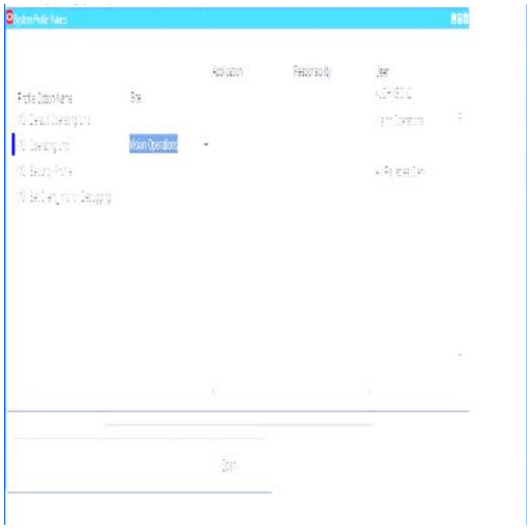


Figure 5 : Assigning Security Profiles

MO: Default Operating Unit

One can define another profile option called MO: Default Operating Unit which is optional and allows user to specify a default operating unit that will be the default when the user opens different subledger application forms.

D. Setup – User Preferences

- User can change default Operating Unit without navigating to Profile Options
- Default operating unit will populate first in Operating Unit Field, if required this can be changed at any time.

Figure 6 : Setup of User Preferences

Impact on concurrent programs

New column MULTI_ORG_CATEGORY in FND_CONCURRENT_PROGRAMS
Values for this column are
– 'S' = Single Org
– 'M' = Multi Org
In upgrade, for programs where definition has been customized, value is null and concurrent program will NOT run.
The following is the Script to set value
– UPDATE FND_CONCURRENT_PROGRAMS SET MULTI_ORG_CATEGORY = {'S' or 'M'} WHERE CONCURRENT_PROGRAM_NAME = '{Program Name}'

CONCLUSION

MOAC is a new feature in Oracle R12 release which minimizes the user's process time to enter the data and create the reports without changing responsibilities. Data security is an additional feature for MOAC. Hence any organization employing shared service can make use of this functionality to efficiently process business transactions.

REFERENCES

[1] "Oracle –General ledger user Guide". http:// docs.oracle.com/ cd/ B34956_01/current/acrobat/120glug.szip

[2] http://forums.oracle.com

[3] "Oracle Application Multiple Organization Implementation Guide". http:// docs.oracle.com/ cd/ B34956_01/current/acrobat/120funmo.pdf .

[4] www.oracleappshub.com

AUTHORS PROFILE

Syed Imdad Huseny, Project lead at Accenture is a OCP certified with over 9 years of experience in implementation of multiple ERP products like Oracle eBusiness suite, good experience in managing and delivering projects, which includes planning, designing, effort estimation, scoping, delivery management and team mentoring. Good exposure in client interfacing and project coordination in offshore-onsite model, both from onsite and offshore. Skilled in managing, leading and motivating teams to accomplish tasks within specified timelines, comply with client specifications and quality processes defined by the organization

Sumaiya Thaseen is an Assistant Professor (Senior) in VIT University with 5 years of experience and also pursuing her PhD degree. A life member of Computer Society of India (CSI). Her areas of interests are adhoc networks, network security and data mining.