Oracle Applications Release 12 Upgrade Project: An Overview of the Joys and the Pains

Marcella Sackett, Ph.D.
Traust Consulting

Introduction

Oracle Applications Release 12 was released in January 2007 with a completely redesigned Financials module and the addition of new features to the Manufacturing modules. The R12 applications are considered "The Global Release" for the E-Business Suite by implementing tools and functions that increase a corporation's visibility to their organization from a global perspective. This is achieved through the introduction and/or enhancement of a Global Architecture and Global Business Functions.

This white paper is divided into two main sections: the first is provides a general overview of the new design, features, and functionality of Oracle Applications Release 12 (R12); the second section provides the insights and experiences of an upgrade project from 11.0.3 to R12.

Overview of Release 12: "The Global Release"

Release 12 of Oracle Applications presents entirely new methodologies, forms, processes, and functions to enhance the global view of a business' financial and operational information. This global approach is obtained through two areas: Global Architecture and Global Business Functions. The Global Architecture is achieved through modifications in the areas of application functions and database structures. The Global Business Functions include thousands of new capabilities and the introduction of new modules that restructure or streamline business processes.

Global Architecture – Application Functions

Examples of the Global Architecture include the introduction of new application functions such as Mulit-Org Access Control (MOAC), Subledger Accounting (SLA), and Ledger Sets. It also includes database features such as additions to the Trading Community Architecture (TCA) and the Oracle Applications Tablespace Model (OATM).

Multi-Organization Access Control (MOAC) is a new feature to the E-Business Suite that includes the capability for a centralized access point to application functions to all operating units within an organization. Previously, if an end user needed to access data for different operating units, they would need to exit their existing screens and either use the change organization function or to change their

responsibilities to a custom responsibility pointing to a single operating unit. These options lead to either increased risk at data entry due to lack of visibility as to which organization was active or increased management and overhead of access to the different operating units within an organization. It also limits the end user's capabilities to analyze information for all operating units. Thus, single-point access to all operating units improves efficiency and global visibility.

The modules impacted by MOAC are varied and include modules where multi-operating unit visibility is beneficial: Payables, Procurement, Supply Chain Management, Order Management, Customer Data Management, and Receivables. The functions impacted within these modules are detailed in Table 1

Area/Module	MOAC Functions
Payables/Payments	Funds Disbursements and Receipts
Procurement	Requisitions, Demand Purchase Orders
Supply Chain Management	Receiving
Order Management	Order Management
Customer Data Management	Trading Community Architecture (TCA) Credit Card Numbers
Receivables	Billing and Collections

Table 1. Modules impacted by Multi-Organization Access Control

One example where MOAC is now utilized is within the Payables/Payments module. When MOAC is enabled for a Payables/Payment responsibility, then the data entry and the funds disbursement can be performed for all operating units in a centralized responsibility. However, MOAC is an option and is not required. This provides the system administrators the flexibility of choosing to provide multi- or single-organization access through specific responsibilities.

The flexible security options provided by MOAC extend beyond the option of controlling access to the operating units at the responsibility level. It also includes new templates for a variety of functions within a responsibility to allow for single- or multi-operating unit access. One example of this is in the Payments responsibility where the Payments Administrator can choose to have payment batches created for one or more specified operating units. The combination of multiple operating units into a single payment batch increases efficiency and the flexibility of the templates provides flexible security.

Other areas of the application utilizing MOAC include reporting and several of the user interfaces (forms). Reports now offer the option for single- or multi-operating unit execution by the addition of an operating unit parameter to many reports. Also, several user interfaces have also been updated to include an operating unit field so that the end user is well aware of which operating unit s/he is working within

Subledger Accounting is another new feature of the Oracle Applications Release 12 and is the basis for the primary changes to the Financials modules. The Subledger Accounting (SLA) provides a centralized repository for accounting transactions. All transactions pass through the SLA and thus accounting models and rules are applied to each transaction in a consistent manner. In fact, now that the General Ledger data all passes through the SLA, the accounting close process has been much more consistent and streamlined across all of the modules. Since all of the data transacts through the SLA, there are several benefits that result from these changes (see Table 2) including increased auditability and the ability to be compliant with accounting rules from both the local and the corporate perspectives. The increased auditability derives from the fact that the SLA contains complete drill down capabilities from the General Ledger to the individual transactions. For instance, the end user can start at a summarized , Payables journal entry and drill down via the SLA all the way to the individual Payables invoice transactions. Another benefit driven by the consistency of the data in the SLA is the ability to report the data in multiple representations. These representations are derived through Ledgers and Ledger Sets. These new reporting capabilities replace the traditional "Set of Books". A Ledger is touted as the "repository of financial truth" in which all of the transactions contain the same 4 "c's": accounting method, chart of accounts, calendar, and currency. The Ledger Sets, on the other hand, are a grouping of Ledgers that must have at least the same chart of accounts and calendar. These Ledger Sets provide the corporation with the ability to view their global financial information in a single, cohesive unit. Through the use of Ledgers and Ledger Sets of data in the SLA, an organization is readily able to comply with corporate accounting rules for standardization, local accounting rules for improved compliance, and improved global compliance.

Benefits of Subledger Accounting (SLA)	
Benefit	**Achieved Via**
Corporate Rules	Accounting Standardization
Local Rules	Improved Local Compliance
Increases Control	Common Data Model Data Model and Repository
Increases Auditability	Full Drill Down Capabilities
Streamlines Closing Process	Common Posting Engine
Global Compliance	Multiple Accounting Representation

Table 2. Benefits of Subledger Accounting

Global Architecture – Database Structures

The global architecture, as it pertains to database structures, was further expanded in R12 from the basis established in 11*i*. The Trading Community Architecture (TCA) now encompasses the Payables module by including supplier data and the Cash Management module as it pertains to both internal and supplier

banks and bank accounts. In fact, it should be noted that the banking setups have been moved to the broader, new module of Payments. Thus, the TCA now includes the following “parties”: customers, suppliers, employees, and banks. In order to facilitate this movement towards TCA, Oracle has included views of the original tables to assist with any backwards compatibility concerns. Inclusion of these modules within the TCA facilitates the maintenance of information for the suppliers and the banks across all operating units. One change can be automatically propagated to all operating units.

The Oracle Application Tablespace Model (OATM) was first introduced as an option with 11*i* applications continues to be encouraged as part of the R12 applications. This model consolidates the 300+ tablespaces within the database down to a total of 12 tablespaces. This includes one tablespace for the base tables, one tablespace for index management, and 10 tablespaces for database management such as undo, temporary, and system tablespaces. The reduction in the number of tablespaces contains several benefits including an increased ease in management and efficient space utilization.

Global Business Functions

In conjunction with the enhanced Global Architecture associated with Release 12, there are several new enhancements to the business functions within the applications. These new features include several new modules, the release of new tax and bank models, and enhancements to the manufacturing modules. Some of the new modules included with R12 are the following: Payments, E-Business Tax, Sourcing Optimization, Project Portfolio Planning, Service Desk, Case Management, Clinical Data Repository, Customer Loyalty, Utility Biller, Fiinancial Consolidation Hub, and Healthcare Intelligence. In conjunction with these new modules, Oracle lists that Release 12 contains over 2,400 additional new capabilities.

With Release 12 being a global release, Oracle has also redefined some of its models in order to accommodate this globalization. Two of the main areas of change include the new tax model as part of the E-Business Tax module and the new banking model. E-Business Tax is the centralization of all of the tax information within the E-Business Suite. This new model is so all encompassing that it literally includes all of the modules that contain tax information except for tax associated with the Payroll module. For instance, some of the modules that utilize the new tax engine are Purchasing, Internet Procurement, Payables, Inter-Company Invoicing, Order Management, Receivables, General Ledger, and Internet Expenses. This centralized repository of tax information increases the ease of maintenance, especially when it is combined with the third party options of either Vertex or Taxware. However, the third party additions are not required and tax can still be manually maintained and customized to meet the needs of each corporation. Another benefit of the new E-Business Tax module is that it contains a Tax Simulator function which allows the end user to test transactions containing tax without creating actual transactions within the system.

A new banking model is also introduced as part of Release 12. This new model ties bank accounts to the legal entity level rather than to each operating unit. In addition to the obvious increased ease of maintenance, the new model enhances the new Payments module. After each operating unit has entered their own payables invoices or through the use of MOAC, invoices for all operating units can be entered, a single payment instruction can be created for each bank being utilized. Thus, a single payment instruction, previously termed a payment batch, can be created for the organization as a whole for all payments transacting through the same bank.

The Financials area is not the only set of modules to be impacted in the globalization of the E-Business Suite. Enhancements in the manufacturing areas include the introduction of the Manufacturing Execution System (MES); a converged, single source of global inventory; a third party option of Demantra; and Transportation Management. The Manufacturing Execution System (MES), designed for both discrete and process manufacturing, is the “one stop shop” for shop floor operations. It includes a centralized location for WIP, Quality, Time and Labor Tracking, on-line instructions and requirements for individual operations, and a clock in/clock out capability.

The converged, single source global inventory model for mixed-mode (discrete and process) manufacturing now stores inventory information in a common data model. Of the two modes of manufacturing, the process flow mode is the most impacted. Also, this globalization still requires the creation of separate organizations for the process and the discrete data.

Two additional areas of opportunity include the option to use the Transportation Management module or the point solution of Demantra. Transportation management assists with freight cost reduction and the benefits of this module include that it is 100% web-based, is highly configurable, and can support a global business. Demantra, which is not part of the E-Business Suite, is an advanced demand management tool This tool assists with the integration of the often hard to manage areas of demand planning such as trade promotion management and “real time” sales operations planning.

Summary of the “Global Release”

With the release of Oracle Applications Release 12, the E-Business Suite (EBS) is now truly a “Global Release”. This lofty goal was achieved primarily through a complete redesign of the Financials module and the addition or enhancement of a variety of functions. These changes all results in an increased efficiency in daily tasks, new functionality, new terminology, new processes, and the ability to have a global view of the a corporation.

Upgrade to Release 12: the Joys and Pains

Summary of the Upgrade Project

The E-Business Suite upgrade project was performed for a mid-sized, discrete manufacturing firm in Minneapolis. The upgrade was a big leap from Oracle Applications 11.0.3, rdbms 8.0.6 to Oracle Applications Release 12, rdbms 10g. The footprint of the existing Oracle Applications was relatively small and consisted of General Ledger, Payables, Inventory Bill of Materials, Material Planning/MRP, Engineering, Purchasing, and Cash Management. Also, the number of customizations in the system was minimal; however, there existed one critical customization that utilized the Demand Interface. This interface fed information from a custom, legacy order entry and shipping system into Oracle Applications to generate demand and to relieve inventory.

Project Overview

Decision to Upgrade to Release 12

Prior to choosing to upgrade to Release 12, the company considered upgrading to 11.5.10. Several factors were taking into consideration when R12 became an option such as the stability of 11.5.10 as compared against the opportunities available in R12. The decision to move to R12 rather than 11.5.10 was achieved primarily by considering the long-term, strategic vision of the company which included significant growth and expansion. Another aspect was that as an early adopter of R12, the client, who is relatively small-sized in the world of Oracle Application users, would have the opportunity to utilize the Critical Accounts Support team within Oracle. The goal of the project was to only use this team if deemed necessary due to slippage in timelines driven by issues in the application. The primary benefit of upgrading to R12 was that it fit with the strategic vision of the client. Prior to deciding to upgrade the Oracle Applications, the client had spent time selecting an ERP software solution and in designing their future state of their IT infrastructure to match and accommodate the strategic vision of the corporation as a whole. After deciding the stay with Oracle Applications as their ERP system, the client designed a multi-year rollout to migrate their legacy system components over to the E-Business Suite. Since the client was starting with a relatively small footprint of modules that did not yet include some of the larger, more complex modules such as Order Management and Accounts Receivable, it was decided that upgrading the smaller footprint would be more straightforward than upgrading these larger modules at a later date. Each of these individual factors led to the decision to upgrade to Release 12.

Module Footprint, Infrastructure Plans, and Team Structure

The module footprint for this client is relatively small with 8 modules in 11.0.3 including General Ledger, Payables, Inventory, Bill of Materials, Material Planning/MRP, Engineering, Purchasing, and Cash Management. Due to the changes in the structure between the 11.0.3 and the R12 application, four

additional modules were added to the footprint: Payments, E-Business Tax, Legal Entity Configurator, and Subledger Accounting.

As part of the upgrade project, the client also integrated the purchase of new hardware which included a platform change from HPUX to Linux. The goal was also to move from a 32-bit platform to a 64-bit platform.

The team to support the upgrade project contained team members from both the client and the consulting firm. On the client side, the project team consisted of a Project Sponsor, two IT staff members, and Subject Matter Experts (SMEs) in the areas of Finance, Purchasing, Material Control, Engineering, and Planning. The consulting firm provided the DBA, developer, a Manufacturing Business Analyst, a Financials Business Analyst, and a Project Manager who also performed Business Analyst tasks. The focus of the consulting team was to create the new environment(s), perform map and gap analyses of the functions, business processes, and reports on the existing version of the applications as compared to what was available in Release 12 of the applications. In addition to these tasks, test cases and new business processes were created. The goal of having the consulting team perform all of these tasks rather than the SMEs was to reduce the overall time commitment required by the firm's internal staff.

Original Project Timeline

The original project plan (Figure 1) was set to be a 6.5 month project. The project consisted of five distinct phases: Analysis, Design, Solution Pilot, Construction, and Transition. The Analysis Phase included business requirements analysis, infrastructure planning, and the initial DBA planning for the upgrade. During the Design Phase, the test cases were to be created and the initial DBA "playbook" for the upgrade completed in conjunction with a first build. The Solution Pilot was slated to have the DBA complete the first upgrade dry run (UDR) for the consulting team to use for detailed analysis of the R12 environment. Another deliverable during the Solution Pilot phase was that the consulting and the client team were to begin the business process design for the cutover/go live period. The Construction Phase from September through October was designed for two additional upgrade dry runs (UDR-2 and UDR-3) along with User Acceptance Testing (UAT). The final period, the Transition Phase, was slated for finalization of processes for the cutover by including a final practice simulation, actual "Go Live", and a two week period for stabilization.

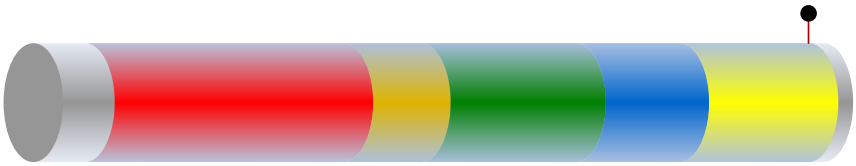

Figure 1. Original Project Timeline and Phases

The Reality of the Upgrade: The Joys and Pains

Timeline Struggles

The joys and pains that were to be experienced became evident early in the project. The first sign of pain was experienced through timeline slippage. The reasons for the slippage were due to three main areas: departure of the only DBA, migration to new hardware, and the difficulty of the first stage of the upgrade from 11.0.3 to 11.5.10.

Infrastructure Concerns

The issue of the departure of a key staff member is not atypical for a project. In this case, the turnover was not the only reason for the initial time slippage. The issue was that the time required to create their "upgrade playbook" was going to take significantly longer than originally estimated. The creation of the new environment was on the critical path of the project timeline because without a system with actual, client data, it was difficult for the business analysts and the developer to meet their deliverables timelines. The delay in the "upgrade playbook" design was driven by two main factors: the migration to new platforms, especially the migration to 64-bit for both the database and the application tiers, and the upgrade steps from 11.0.3 to 11.5.10 (11.5.10 is required as a base application to move to R12). Risk mitigation came into play immediately after these issues were realized. Unfortunately, the only way to mitigate the risk of the issues surrounding the upgrade passing through 11.5.10 was to increase the timeline to allow for issue resolution. The issue of the 64-bit servers for both the application and the database tiers was able to be mitigated by choosing to upgrade on 32-bit servers for this project and then move to 64-bit at a later date in conjunction with another project.

May June

Module Concerns and a Joy

After the original risks associated with infrastructure were addressed, the second issue that became evident was that there were issues with specific modules within the application. Specifically, most of the issues involved the invoice workbench within the Payables module. The root cause of the issues this this module stemmed from the complete restructuring of all of the Financials modules through the introduction of Subledger Accounting (SLA) and the introduction of Payables invoice lines. The issues with the invoice workbench and its underlying functions were so extensive that it was several months (approximately 4 months) before we could enter an invoice from beginning to end. The long length of time was driven by the fact that the issues seemed to reside in multiple layers of the code so as soon as one issue were resolved through Service Request work that another series of issues would arise at the next level. The concerns surrounding the Payables module were so extensive that we were able to leverage one of our original decision points for choosing to adopt Release 12 and we started working with a Customer Service Representative on a bi-weekly basis. The Customer Service Representative worked with a list of prioritized Service Requests to focus efforts within Oracle Development. After a period of time, it became evident to both the client/consultant team and the Oracle representative that the project required the attention of an additional layer deeper within the Oracle organization. It was at that point that the project was assigned to a Critical Accounts Manager who was able to leverage both the analysts and the development team. Through the efforts of the Critical Accounts Manager, the Service Requests associated with the upgrade project were prioritized, managed, and escalated internally within the Oracle team. During this period of issue resolution within the payments module, the invoice workbench form version level increased from x.x.23 to x.x.112. This level of version increase is indicative of the fact of the layers of complexity and thus the layers of issues surrounding the invoice workbench in the Payables module. However, these issues also led to the silver lining of being assigned a Critical Accounts Manager and learning and understanding in more detail the internal workings of Oracle.

Final Timeline

The timeline slippages were inevitable due to the issues that arose. Adding additional resources to the project on the client or the consultant team would not have sped up the resolution of the issues involving the Payable module. The project timeline became reliant upon the Oracle development team who resolved issues as quickly as possible.

The new timeline (see Figure 2) ended up including two extensions and an additional upgrade dry run to the timeline. These extra phases of the project resulted in the addition of 3 months to the original schedule. The benefits of extending beyond the original timeline were extensive. They included increased user time on the new system, increased practice time for the business processes during the cutover to the new system, increased number of practice rounds for the DBA, and resolution of nearly all of the open issues.

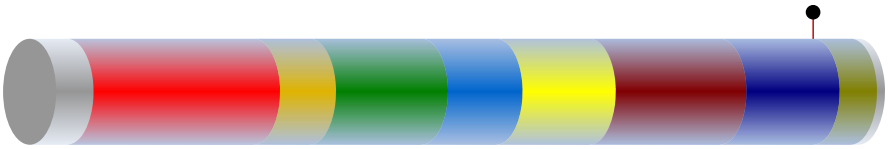

Figure 2. Final Project Timeline and Phases

Thoughts/Recommendations on R12 Upgrades

Having successfully completed an R12 upgrade, there are several key thoughts and recommendations to present to other Oracle customers who are considering upgrading to R12. The recommendations vary from preparatory steps to the planning of the project timeframe.

Project Timeframe Considerations

As with any project, there are several criteria and points to consider when determining a project timeline. Several points for consideration include the size of the current footprint as compared to the R12 footprint; customizations/extensions in the system and where they reside in the application; and the team members' experience level with R12.

May - Jul
Analysis

May June July

Current Footprint vs. Future (R12) Footprint

In the case of upgrading to Release 12, one area of consideration is the current size of the module footprint as compared to the final size of the module footprint. This project involved the relatively small initial footprint of 8 modules, but with the addition of new features and functionality in R12, a total of 12 modules were part of the upgrade. The additional modules that were implemented will be a part of almost any R12 project. These modules include E-Business Tax, Subledger Accounting, Legal Entity Configurator, and Payments. These new modules each have their own setups and configurations that must be learned. The primary method utilized on this project was initially through implementation manuals, user guides, and quite often Online Web Conferences (OWC) with analysts from Oracle (see "Experience with R12" section for more details).

Customizations

Also, customizations are a key point of consideration. In this project's case, the customizations to the system were focused on an interface between a legacy order entry system and the obsolete Demand Interface. Due to the obsolescence of the Demand Interface, the project also needed to include the additional design, development, and setup of the backend of Order Management to feed information into MRP.

If your organization contains customized reports and forms, it is critical that an evaluation be performed of where the customizations lie and how the customizations will be impacted by the redesign of the application. For instance, a custom report on supplier information is unlikely to require modifications because although the supplier information is now stored in the Trading Community Architecture (TCA), Oracle has provided backwards compatibility to the TCA through views with old table names (ie. po_vendors is now a view that feeds and reads information into the TCA for suppliers). However, if a customization were created on the suppliers form, then development and/or gap analysis time needs to be added to the project plan. The reason for this is that suppliers are now entered and maintained through a web-based form rather than the traditional forms structure. In fact, any areas impacted with new functionality (except for the Payables invoice workbench) have been migrated to web-based forms and while these forms provide a variety of available personalizations, they may not fulfill the entirety of the business need.

Other areas to be considered when creating a project timeline are custom interfaces and data conversions into the Financials modules. These modules and the underlying data structures have been significantly updated with the addition of the Subledger Accounting Module and should be reviewed in detail.

Experience with R12

The significant changes to R12 means that even Oracle Applications experts with extensive knowledge of the Oracle E-Business Suite are required to re-learn and learn anew many of the standard features within the applications. For instance, navigation paths and setup paths have changed. This may seem like a small and expected change within any upgrade; however, the changes and new features with R12 are extensive. This lack of familiarity means an increase in time for the creation of test cases because new navigation paths and business processes needed to be learned and documented by the analysts.

New module setups also became a factor in the project timeline. The documentation manuals associated with these setups were in their infancy at the time of this project (as of late 2007) and therefore, they were not extensive enough to complete setups without the creation of a Service Request and then requesting an online web conference (OWC) with the analyst. At the time (this was greatly improved by the end of

the project), the first OWC would be held and it was evident that the frontline analyst would not be able to answer the questions. Thus, a second OWC would be scheduled in order to have the lead analyst on the phone, too. The infancy of the manuals and the duplication of effort on the OWCs lead to an increased timeframe for setups of the new modules.

Preparatory Steps

Needless to say, there are preparatory steps that corporations can take in order to decrease the project timeline and to increase the overall smoothness of the project to upgrade to Release 12. The major areas for preparation are the Functional Team preparation, the Technical Team preparation, and the end user training.

Functional Team Prep

The basis of the Functional Team preparation is to read and practice. The reading needs to occur at multiple levels starting with the Implementation Guide. The Implementation Guide with specific information for the Subledger Accounting and the method for upgrading your data. Next to be read is the Upgrade Guide. Finally, the individual module guides are a must read.

The second area of preparation is to practice, practice, practice. The setup of a Vision environment pre-project and then the setup of an actual R12 environment will increase the navigation comfort level and the knowledge base of the Functional Team. These team members need to learn the new modules and the new navigation paths within the areas that have been updated.

Technical Team Prep

The Technical Team also has areas of preparation including new technology training. The areas of training for the technical team include Forms 10, Fusion Middleware, Application Developer Framework, BI Publisher, and JDeveloper.

Forms Customization

There are still plenty of traditional forms within the E-Business Suite. The forms are now based on Forms 10; however, it is obvious with this upgrade that Oracle is starting to migrate most newly created/revised forms to self-service forms. The self-service forms utilize the OA Framework and thus have a wide range of possibilities for customization including the ability to add fields, hide fields, and to make fields required. Prior to attempting to customize these forms, it is important the OA Framework personalization options be reviewed and analyzed. The reason for this is it is much more difficult to customize a self-service form versus the traditional forms in which everything was contained with the form or database packages. The self-service forms are difficult because it is not just one form, but instead includes java classes, XML files, “metadata fileds”, and JSP files.

Reports Customization

The traditional reports in R12 are based on Reports 10g. However, in a fashion similar to the forms, Oracle has started the migration of new reports and some previously existing reports into BI Publisher (previous XML Publisher) reports that are based on XML or XSL-FO. A large number of rtf templates are provided for reports based in BI Publisher. Most of these templates are easily customized; however, some still require some XSL-FO coding. The benefit of utilizing these templates is that there are several output options including PDF, Excel, HTML, Word, and PowerPoint and, as long as fonts are installed on the servers, then there are several font options.

Summary of Lessons Learned

There are several lessons learned throughout this upgrade project to Release 12. The main areas and aspects in the realm of "lessons learned" include the following: the upgrade, the number of version releases between the current of the upgrade; reports in the Financial modules; training; and the setups for Financials.

The Upgrade

The first lesson learned is in regards to the upgrade itself – the number of version releases "jumped" from current level to R12 level. This project started with the baseline of 11.0.3, contained an interim step at 11.5.10, followed by the upgrade to R12. It is the first of these steps from 11.0.3 to 11.5.10 that was the longest and the most difficult part of the upgrade. Each upgrade dry run generated new issues in this step that needed to be problem-solved and analyzed rather and slowed down the process. The goal was to make this part of the upgrade consistent and streamlined, but that was not possible. On the other hand, the upgrade from 11.5.10 to R12 was extremely streamlined with minimal issues. An option that should be considered, but was not used on this project, is the Maintenance Wizard that assists with upgrades and patches. The Maintenance Wizard was not selected for this project since it requires an additional database installation and was not desired by the client.

When upgrading from 11.0.3 (or earlier) to R12 with an interim step at 11.5.10, it is imperative that a working version of the 11.5.10 applications be left intact in order to problem-solve upgrade issues. Issues were discovered in the SLA and other areas that utilize the new accounting event structures such as AP and PO Reconciliation report and the Accounts Payable Trial Balance. It would have been much easier and more effective to have a working 11.5.10 environment available that was the interim step for the R12 environment. Once we had an 11.5.10 environment that correlated with our working R12 environment, the Oracle analysts and developers were able to assist with a gap analysis of the data and create one-off patches to resolve the issues.

Reports

The R12 release is primarily a re-write of the Financials modules, and thus it can be expected that most, if not all, of the Financials reports are also affected. The result of the movement to the new data structure is that many reports were moved to BI Publisher format or have a new process in place to reflect the addition of the Subledger Accounting. In fact, there were many financial-related reports that did not work for several months including the AP and PO Reconciliation report, the Purchase and the Invoice Price Variance reports, the Accounts Payable Trial Balance, and the 1099 report. These reports are required for monthly reconciliation and for validation of the upgrade of the data in the new environment. Thus, the reports became part of the critical path for the success of the project. Reports are rarely in the critical path and are thus often an after thought in end user testing even when the project team encourages thorough testing of reports. Even though the reports listed above have since been fixed, it should be noted that testing time should be increased to insure that all reports are thoroughly tested and are working as expected.

Training

There will be a need for increased training time, especially for the Financial end-users since these modules are so greatly affected by the upgrade to R12. The users need hands-on activity prior to UAT so that they are comfortable with their new process. Perhaps the biggest impact and training and working with super users is that there are several points of new navigation. There are also areas of new navigation that have movement and toggling between traditional forms to self-service forms and back to traditional forms. For instance, the Subledger Accounting provides the opportunity for complete drilldown capabilities from a summarized journal entry all the way down to the individual transaction line. These data movement involve the movement from traditional form to self-service and back to tradition forms (see Figure 3). There also needs to be allotted time so that the end user can learn their new modules.

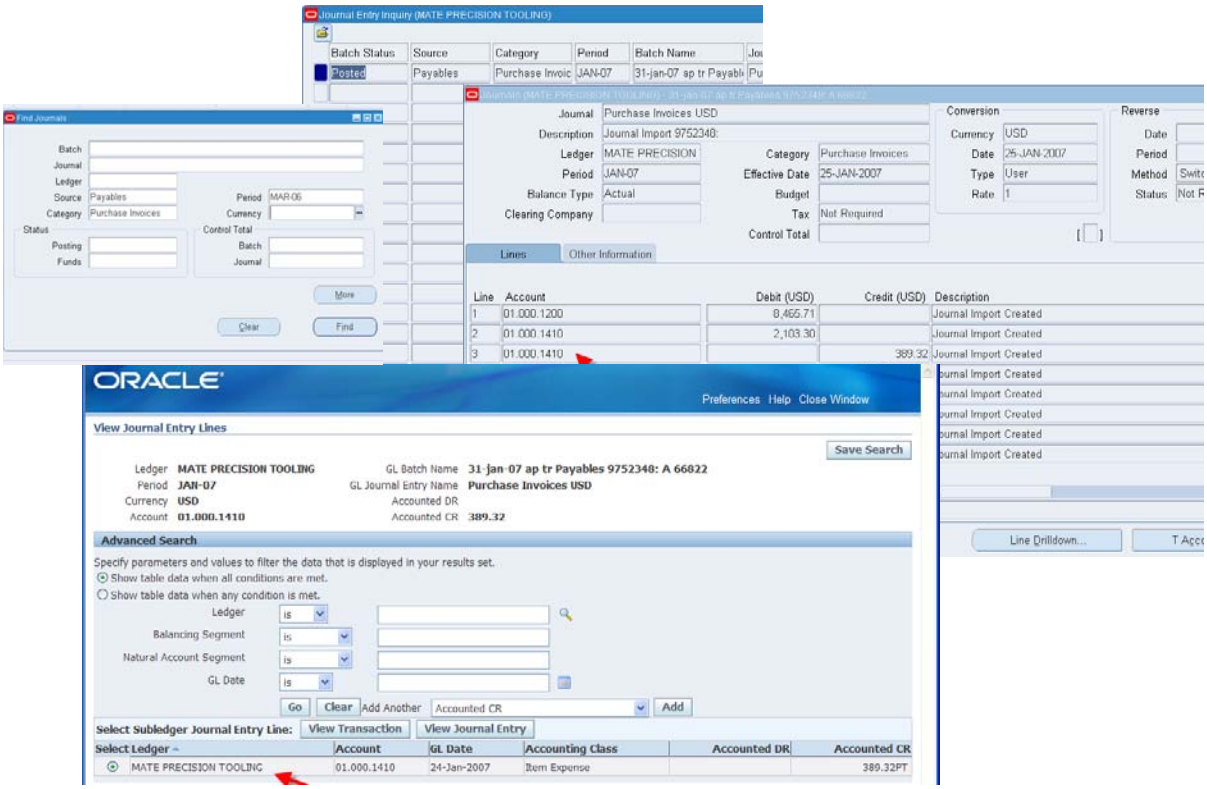

Figure 3. Navigation between traditional and web-based forms

Summary

In Summary, the Oracle Applications Release 12 truly is a “Global Release”. The new features and designs such as Subledger Accounting, Ledger Sets, and Multi-Organization Access Control provide optionis for a global perspective of an organization with streamlined, efficient methodologies. However, as with any release that includes substantial redesigns and modifications comes issues and the need for new approaches and business processes. Some of these new approaches, from an upgrade project perspective, include training for end users, functional analysts, and the technical team. Also, increased project time should be taken into consideration for training of end users; validation and testing of reports; and the increase in size of the module footprint that is driven by the R12 restructuring. All in all, the Release 12 application is a powerful, newly designed application that continues to improve the efficiency and the options for globalization.