
Implementing A Global Order Entry/Installed Base System Within The Limitations Of The Oracle Multi-Org Environment

Trevor Cusworth
Aspen Technology, Inc.
Stephen J. Noone
Arthur Andersen
Rebekah A. McKie
Oracle Corporation

Introduction

Aspen Technology, Inc. is a software and services company that provides solutions for plant design and simulation; advanced control and optimization; and information and supply chain management to 90 percent of the top chemical and oil companies in the world. As a company whose products are used by process manufacturers to increase their profitability and competitiveness, AspenTech's expertise has been in great demand over the past few years. In the last three years, AspenTech's employee base has grown from 350 to over 1500, in a large part due to the acquisition of 13 companies. AspenTech has 22 offices located throughout the world.

To manage this rapid growth, AspenTech embarked on a reengineering project in order to define common business processes and systems across the entire company to replace all of the individual business processes of the acquired companies. As a part of this reengineering effort, AspenTech selected Oracle Applications 10.7SC, as its global solution for Financials, Human Resources and Order Management

AspenTech's business is global in nature and a significant part of its revenue is derived from large multinational companies. In an effort to successfully support this business, AspenTech requires its Order Entry and Installed Base tracking systems also be global in nature. AspenTech's system requirements therefore include shared customer and installed base information.

The reason for this is AspenTech could be providing product to its customers from different internal business units in different parts of the world. To accomplish this it is essential that each of the customer sites and systems, to which that software is being sold, be unique in the database to reduce the risk of errors in shipment and future support and upgrades.

In order to allow its users the ability to implement applications for their global operations with a single instance of the database, Oracle introduced the multi-organization (multi-org) functionality. Using Oracle's multi-org functionality, data can be logically partitioned within a single instance of the applications and database. General Ledger data is partitioned by a set of books; Order Entry (OE), Accounts Receivable (AR), Purchasing (PO), Accounts Payable (AP) and Service data is partitioned by operating unit. Inventory information is partitioned by inventory organization.

User access to specific data is secured by defining Oracle responsibilities. Responsibilities in Oracle are "organization" specific. Each responsibility is associated with a specific set of books and operating unit.

Unfortunately, use of Oracle's multi-org functionality is on an enterprise-wide basis. The data partitioning is an all or nothing decision. Data cannot be selectively partitioned within the applications. This functionality limitation created a challenge for AspenTech. AspenTech's operating model and business requirements included global OE, customer and installed base information and AR data partitioned by sales region.

Background

AspenTech's current Oracle organization structure is shown in Figure 1. There are currently 14 sets of books, which are mainly characterized by functional currency. There are also 2 consolidation sets of books, one for the European entities and one for the overall company. There are 16 operating units. The total number of inventory organizations has not yet been defined, but currently most of the software products are shipped from the US; from Houston, Cambridge or Seattle. A small amount of product is shipped from the U.K.

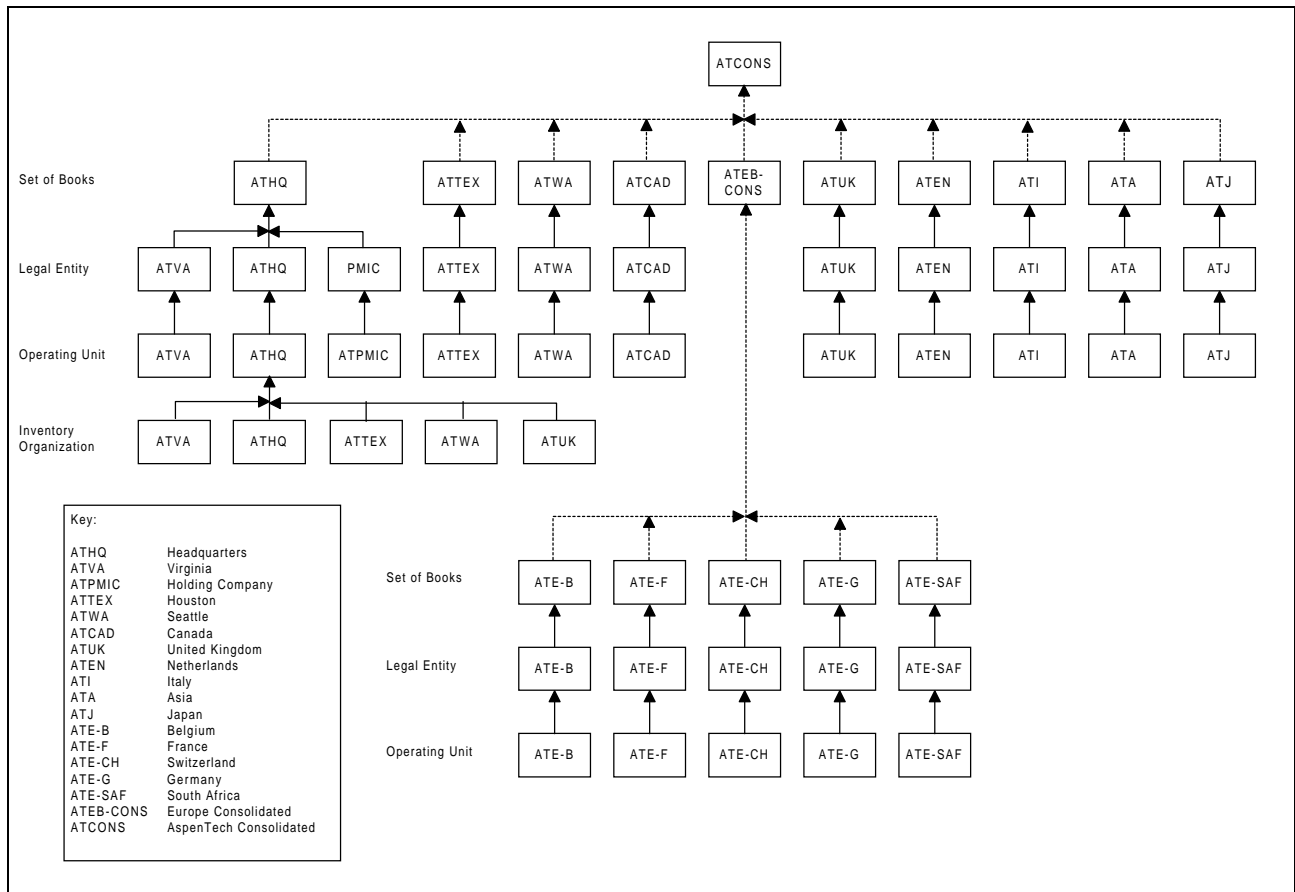


Figure 1 - AspenTech's Current Oracle Structure

AspenTech divides the world into four sales regions:

- NALA North America/Latin America
- EMEA Europe, Middle East and Africa
- ASIA All of Asia except for Japan
- JAPAN Japan

Currently the majority of AspenTech's software sales are in the form of "single point" solutions. Moving forward, the trend is to provide the marketplace with a more integrated suite of software and services. These software products and services are delivered from different business units within the company. These business units are geographically dispersed and do not necessarily align with the Sets of Books structure or the sales regions. Therefore to, accomplish this integration an environment needs to be established to allow the sharing of information.

A major objective of the reengineering project is to achieve greater integration of the AspenTech businesses. Implementing the Oracle business applications repre

sents the first step in achieving this business integration. By establishing a single company-wide transaction system, AspenTech will have a common tool for managing its business.

Key Business Requirements

Based on the previous discussion, AspenTech has identified a number of key business requirements for the Order Management Process (order entry, order fulfillment, accounts receivable and revenue recognition). These business requirements form the basis on which the Oracle system will be designed and implemented.

The key business requirements of AspenTech for the Order Management system are as follows:

- Move the major sales administration functions (order entry, invoicing and collections) out to the four major sales regions, and therefore, closer to the customer

- Allow order entry anywhere in the world for all product(s)
- All cash application related to product sales will be performed in the US
- All order fulfillment (i.e. pick, pack and ship) will be done in the US
- Provide a global/single view of the installed base
- Provide shared customer information including addresses, site codes and contacts
- Provide the ability to handle order entry and accounts receivable in any defined currency
- Decentralized administration of Accounts Receivable

It is also important to understand the rules that are used for the allocation of revenue within AspenTech. There are basically two types of item being sold. The first item is software products and associated maintenance or support agreements. The second item is project consulting services. The revenue recognition rules are very different for each of these.

For product or maintenance sales, the revenue is recognized when the product has shipped. Whereas for consulting services, the revenue is recognized as the services are executed.

The following rules are used for the allocation of revenue within AspenTech:

- All software is licensed with US contracts and is US revenue
- Maintenance contracts (i.e. support services) are US revenue
- Non-US entities “earn” revenue from the sale of software licenses and maintenance contracts through transfer pricing
- Project consulting services revenue are provided by both non-US and US entities and the revenue will be earned in the local region providing the resources to execute the services.

Multi-Org Limitations

Oracle’s multi-org environment was designed for companies to divide and isolate transactions among operating units in their organization. Previous to release 10.6.1 of the software, multiple instances of the database needed to be used in order to manage an entire enterprise and to isolate parts of a company from each other. With multi-org, Oracle has added an organization ID to tables in the database, which has enabled them to partition and give users access to particular transactions depending on their responsibility. In this way, one in-

stance of the Oracle database can be used for an entire company enterprise.

In the Oracle Applications, several of the modules have been made multi-org aware i.e. their data is segmented by organization ID. These modules include Order Entry, Accounts Payable, Accounts Receivable, and Service.

In the multi-org environment, customer and vendor header information is available across the enterprise, but customer detail information such as addresses, contacts, installed base sites (or systems) are unique to an operating unit, which means that the information cannot be shared among operating units. Similarly transactions entered in one operating unit would not be visible in another operating unit.

The implication of this multi-org structure is that different AspenTech locations will not be able to share customer information. It is possible that two locations attempting to ship different products to the same customer, to be installed on the same machine, will not know that the customer address may already be defined in the system, which could cause errors in order fulfillment, and reduce customer service.

Similarly in the installed base, it is possible to have the same site/system information defined more than once on the Service module due to the fact this information is only unique to an operating unit and not globally.

Another important aspect of the multi-org environment is that transactions in a particular operating unit, such as revenue recognition, flow to the set of books associated with the operating unit. Therefore, for example, if an order was placed in the AspenTech Belgium operating unit, then the revenue for that software would flow to the ATEB set of books, even if the software was shipped from the U.S. This is an important point, which will be discussed further in the paper.

Design Options

As mentioned previously, AspenTech would like to be able to enter orders anywhere in the world and share customer and installed base information globally. AspenTech would also like to recognize all revenue for software and support service sales in the U.S. For project consulting services, AspenTech would like to recognize revenue where the project resources are being sourced. Also, the accounts receivable and collections responsibility will belong to the entity which entered the order.

With these basic requirements in mind, several enterprise options were examined to determine the best set-up of the Oracle system in order to meet these needs.

Three options were considered for the enterprise structure. In each of these options, it is assumed that the product fulfillment will be performed out of one or more of the US offices. These options were:

- Custom Solution – Centralized OE and Service/Decentralized AR (option 1)
- Centralized OE/Service and AR (option 2)
- Multiple OE/Service and AR Operating Units (option 3)

Each of these options will be discussed in more detail below:

Option 1 – Centralized OE and Service with Decentralized AR

This is a custom solution which has been implemented by other Oracle customers. Figure 2 shows a diagram of how this option would be configured.

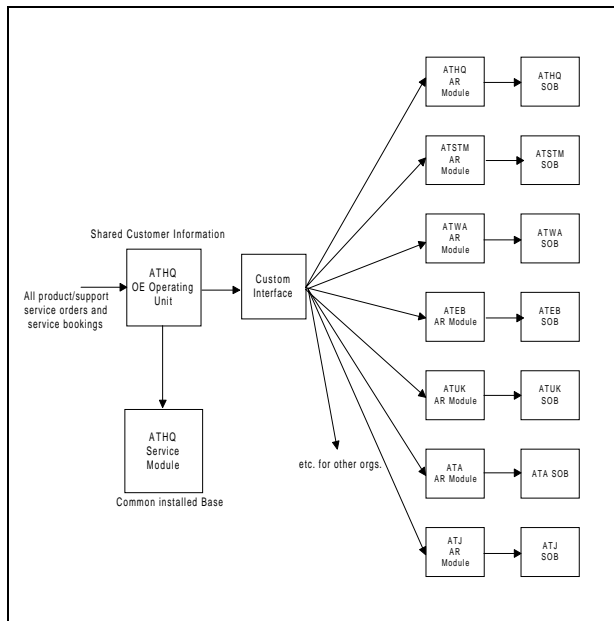


Figure 2 – Centralized OE & Service / Decentralized AR

In this option, all product and support service orders and consulting services bookings are entered into a single operating unit (in this case ATHQ). Similarly all of the installed base information for these products is stored in a single operating unit. Custom coding and programming is used to move transactions out of order entry into

accounts receivable associated with the operating unit where the order was entered.

For example, if an order were entered in the U.K, then the receivables would be tracked in the U.K AR operating unit. The assignment of AR operating unit is achieved based on location information which the user enters at the time of log-on. Once the order information has been passed to AR, the revenue flows into the set of books associated with that AR operating unit. In the example above the revenue from the order entered in the U.K would flow to the U.K set of books. Since Aspen-Tech recognizes software and support service on the US set of books, this option does not satisfy its business needs

For project services, the service booking would be entered in the same single operating unit, but the revenue would be recognized by the project accounting module in the operating unit which supplied the resource. The accounts receivable for the services revenue would be held in this same local operating unit.

Option 2 – Centralized OE, Service and AR

In this case, the current AspenTech Enterprise structure would remain the same. A single operating unit (ATHQ) would be chosen for all order entry and assigned to responsibilities

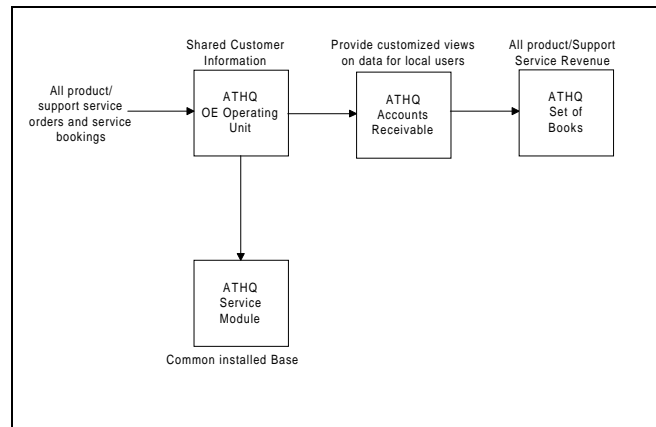


Figure 3 – Centralized OE/Service/AR

Figure 3 shows a diagram of this option. All product and support service orders and project consulting service bookings are entered into this operating unit. The installed base information will therefore all reside in the service module of this same operating unit.

The accounts receivable associated with product and support service orders would reside in the AR module of the ATHQ operating unit and hence all of the products

and support service revenue would flow into the ATHQ (US) set of books. The Oracle standard AutoLockbox functionality will be used for cash application. An example for this option would be for a user in Japan who would log-in to the ATHQ order entry operating unit and enter an order for a product. The product would be shipped from the U.S to the customer in Japan and the revenue associated with the product would flow into the U.S set of books. The invoice could be printed locally in Japan and the collections performed locally. The customer would remit their payment to a US account in a Japanese bank and the cash would be applied in the US.

With this option, the orders could be placed in any supported currency, and the invoice and receivable tracked in that currency. It should be noted that the ability to handle regional sales tax, or VAT, and Multilanguage support were not key business requirements for Aspen-Tech.

As before, consulting services revenue and receivables would be held in the local operating unit providing the project resources for those services.

Since one of the key business requirements is to invoice locally and perform collections locally, it will be necessary to segregate the accounts receivable information by the physical order entry location (where the order was entered). With this option, all of the accounts receivables for product and support service orders would reside in a single operating unit and would be visible to anyone who had access to that operating unit.

It will be necessary, therefore, to customize the application in order to filter the views on the AR data to provide only those people in a particular the ability to view data belonging to their region.

The proposed solution for decentralized invoicing is to use multiple batch sources, database triggers and descriptive flexfields. Based on the origin of the order, the system will determine the batch source site so that AutoInvoice can be run for multiple batch sources. A key user at each order entry site would be responsible for running AutoInvoice based upon the batch source assigned to their order entry location. The invoice can be printed at the point of order origin.

To provide the ability to perform local collections, the proposed solution is to have each Oracle User assigned to one of the four sales regions. This assignment will be populated in a descriptive flexfield in the “Define User” form of the Oracle System Administration. Each customer address will also be assigned to one of the four

regions. A custom program will be designed to match the two descriptive flexfields so that the customer address list of values (LOV) will retrieve a limited set based upon the regional assignment of the user

Option 3 – Multiple OE, Service and AR Operating Units

In this option, shown in Figure 4, the current enterprise structure is not changed. In this case, an operating unit is chosen in each of the sales regions. For example ATHQ for NALA, ATEB for EMEA, ATA for Asia and ATJ for Japan.

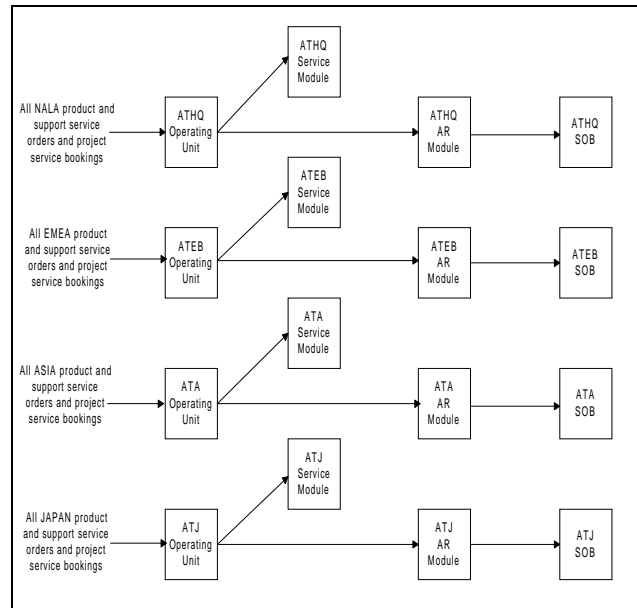


Figure 4 – Multiple OE/Service/AR Operating Units

Product and support service orders and project consulting service bookings are entered in the regional order entry unit in which the customer is located.

The installed base for these products will be in the service module of the operating unit for that region and this is also true of the accounts receivable. The revenue will flow to the set of books that is linked to this regional operating unit.

An example of this option would be for a customer in Germany. The product order would be entered in the EMEA order entry operating unit (in this case ATEB) and shipped from the US to the customer. The accounts receivable for the order would be in the ATEB operating unit AR module and the revenue would flow to the ATEB set of books.

In this case, customer detail information would be unique to each of the 4 operating regions i.e. customer information would not be shared between NALA and EMEA etc. Also, the installed base database would be broken into 4 parts, one for each region. As in the last two options, project services revenue and receivables would be held in the local operating unit providing the project resources.

Solution

With a view to AspenTech's key business requirements for this order management system (as discussed earlier), each of the options were analyzed to see how they matched these requirements. Table 1 shows the advantages and disadvantages of each of these options.

Based on these advantages and disadvantages, option 2 was the path chosen by AspenTech. This option, with the customization of the accounts receivable, meets all of the stated requirements.

With each of the options, custom reporting will be necessary to view global customer revenue and receivable information. Standard Oracle reporting is organization specific and the global customer data will be spread across operating units. This is because revenue and receivables associated with project consulting services will flow into the set of books of the location providing the resources for those services. Therefore, in all cases, revenue and receivables for products and consulting services for a particular customer, may appear in more than one set of books or operating unit. AspenTech has chosen Business Objects for its global reporting requirements.

In each of the options, orders will be entered and invoiced in any defined currency. This is a standard Oracle functionality and therefore none of the options has a particular advantage or disadvantage over the other.

Conclusions

Although Oracle's multi-org functionality was a step forward in allowing companies to manage their entire enterprise in a single instance of the database, it created other problems for some customers. This is the case for companies in the software industry, like Aspen Technology, who need to be able to share customer and installed base data globally while still maintaining individual, regional, sets of books. Therefore while multi-org was able to solve some of Oracle's customer problems, it created new problems for other Oracle customers.

The exercise described in this paper was a good learning experience of the Oracle applications, and the solution chosen achieves most of the desired functionality with minor customization. Due the very nature of the multi-org functionality, some customization will always be required, particularly in reporting, whenever a company is dealing with global customers.

The main conclusion that can be drawn from this paper is that it is critical to understand the key business requirements of your company before embarking on a design of the Oracle Applications. It is essential that you understand the data needs of the company and have a clear picture of how the system is going to be used. Once this has been done, then these needs can be matched to the functionality and capabilities of Oracle and any "gaps" can be identified.

With these known deficiencies, analysis can be done similar to the one described in this paper to identify workarounds or customization in order to match the required functionality. In this way, the outcome will be a system that is well accepted by the users and achieves the business requirements of the company.

About the Authors

Trevor Cusworth is Vice President of Business Operations with Aspen Technology, Inc in Cambridge, MA. Trevor is managing a reengineering project within AspenTech, which includes the implementation of Oracle Financial and Manufacturing Applications.

Stephen Noone is a Manager in Arthur Andersen's New England Business Consulting practice in Boston, MA and has been with the company for 6 years. Stephen specializes in information systems planning, business process improvement and the implementation of Oracle Applications for middle market, high technology companies.

Rebekah McKie is a Managing Principal Consultant for the North Area Practice of Oracle Consulting Services, and has been with the company 4 years. Rebekah is a specialist in the planning, project management, analysis, design, and implementation of Oracle-based single organization and multi-organization structured information systems, and has served roles in both standard and custom implementations of the Oracle Applications family of products.

Table 1 – Advantages and Disadvantages of Order Entry/AR Options

	<i>Advantages</i>	<i>Disadvantages</i>
<p>Option 1 <i>Centralized OE & Service/Decentralized AR</i></p>	<ul style="list-style-type: none"> • Common installed base • Shared customer information • Segregated AR data • Customization has been done before • Ease of use • Easier to maintain data 	<ul style="list-style-type: none"> • Major Customization – increase implementation costs and future upgrade costs • Revenue will flow into set of books in region where customer is located. • Custom reporting for global customer view
<p>Option 2 <i>Centralized OE/SVC/AR</i></p>	<ul style="list-style-type: none"> • Common installed base • Shared customer information • No change to current Oracle structure • All product/support service revenue flows to US set of books • Ease of use • Easier to maintain data 	<ul style="list-style-type: none"> • Some minor customization of views might be required to segregate AR data • Collections personnel may have to log-on to more than one responsibility to perform collections and run AutoLock-box for a particular customers receivable if it is a combination of product order and project services billings. • Custom reporting for global customer view
<p>Option 3 <i>Multiple OE/SVC/AR Operating Units</i></p>	<ul style="list-style-type: none"> • No change to current Oracle structure • No customizations required 	<ul style="list-style-type: none"> • No sharing of customer information • No common installed base • Revenue flows into regional sets of books • Custom reporting required for global customer view • Difficult to maintain data • Difficult to use