

A Giant Step For ERP Web Applications

March 22, 2002

By Jerry Fireman, NDS Systems

There's little disagreement that ERP systems to date have not fulfilled their initial promise. Instead, the trade press is full of stories about ERP implementations that have cost far more than expected, taken years and years to implement, and when they are finally put into operation the applications aren't suitable because the business process has changed.

The core problem seems to be closely related to the primary advantage of ERP systems -- their all encompassing nature -- which means that companies that adopt them must face either a long and expensive process of changing the software to match their business processes or else face the equally daunting alternative of changing their operations to match the software.

Larger companies that have taken on the task of adapting a packaged system to match their operations have found that the cost is usually measured in millions of dollars and the length of time required to complete the process is measured in years. And the initial implementation is just the beginning -- once the software is heavily customized, the customizations themselves must be modified or even completely recreated every time that the company decides to implement a new version of the base software.

Is it any wonder that Larry Ellison, Chief Executive Officer of Oracle, one of the larger ERP software developers, recommends to companies that they should change their businesses rather than trying to change their ERP software?

Some of the software companies in the ERP market have introduced vertical industry-specific solutions. But the cost and implementation challenges involved in these products typically have the effect of limiting their use to the largest companies. What about the possibility of developing cost-effective, industry-specific ERP solutions and providing them to small and medium-sized companies as well as large corporations?

The challenge comes from the fact that the government with its standard industrial classification system has defined well over 10,000 vertical markets. Even within many of these markets there are numerous niches and variations. Developing vertical ERP applications for these industries is a formidable task because it requires both industry knowledge as well as software development expertise.

Traditionally, the high cost of development, high cost of sales, and high cost of ownership has excluded industry-specific ERP products from the small and medium enterprise marketplace. Such vertical solutions have been very expensive to develop because the business logic is tightly linked to the user interface. This means that even relatively small changes, such as changing the terms presented on the screen to match those used in a particular industry, requires significant effort.

The programming effort required for changing the software to reflect changes in business processes can be nearly as great as what was required to create the initial ERP application. Beyond that, even if prepackaged industry vertical solutions were to appear, there are certain proprietary processes that are crucial to nearly every company's competitive strengths that prepackaged solutions cannot effectively model.

A solution emerges

The solution to this problem is appearing in the form of new Web application architectures that separate the core functionality from the user interface. The basic principle of this architecture is that ERP applications as well as customer relationship management (CRM) applications are for the most part transaction-based. The new approach involves the separation of transaction functionality or business logic from the presentation logic.

This segregation is achieved by migrating the business logic into database-stored procedures, triggers and views and using an engine to store the presentation logic consisting of rules, layouts and screens in the database as metadata. NDS Systems began taking this approach in the mid 1990s after 10 years of experience in ongoing ERP application software development.

This ongoing development has produced a powerful and extensive core business logic code base that contains virtually every type of transaction that can occur in an enterprise. NDS Systems has since developed a multi-tier architecture that allows this existing core functionality to be sliced, diced and re-used again and again for an infinite variety of application presentations.

This approach takes advantage of the fact that business logic or functionality is neutral or indifferent to the purpose for which it is being used. In other words, a transaction is a transaction is a transaction. What differentiates industries and businesses is determined in the presentation of functionality, processes and terminology.

The segregation of business logic from presentation logic and the NDS application-generating engine enable the rapid and cost-effective development of vertical industry and company-specific application software. On the average, the amount of effort required to produce a customized or vertical application is reduced from 100% of original code development to only 5%.

The NDS Systems Extensible Virtual Application Engine (XEPHR) is a cross platform, pure Java system built on top of industry standard Java 2 Enterprise Edition components including Servlets, EJB, JMS, JNDI, XML, XSLT and clustering technologies delivering end-user applications through a scalable, fault tolerant, distributed computing environment.

The NDS Systems ERP application is resident entirely within an Oracle 9i database. All transaction handlers, processes and job streams are fully resident within the database and are exposed for use through a series of "smart" views. Although the application is currently available for deployment with an Oracle 9i database, it could be deployed with any ANSI SQL and Java 2 compliant database such as IBM's DB2 universal database. The application code can be translated from Oracle's PL/SQL to Java by any of a number of widely available tools, and IBM provides schema migration assistance.