

# VANILLA IMPLEMENTATION OF ORACLE

## Is a "vanilla" implementation even possible?

Henry Ford once said, "You can have any color car you want, as long as it is black." We've come a long way since then... today's automobile consumers demand the ability to select their cars from a long list of models with a variety of colors, features, and options, all to just get from point A to point B. Switching gears to the ERP software marketplace, it's no surprise then that Oracle's E-Business Suite has the robust functionality and world-class flexibility necessary to meet the needs of best-practice business processes in almost every industry. With such power and flexibility available to Oracle E-Business Suite users, it seems interesting to me today that everyone seems to want only one "flavor" of Oracle... vanilla!

I believe that most people say they want "vanilla" because they want to minimize the cost, complexity, and time required to implement the Oracle E-Business Suite applications. These are fundamental goals for any systems implementation project. But what exactly *is* vanilla in terms of the Oracle E-Business Suite? What do people really mean when they say they want a "vanilla" implementation? Most importantly, is a vanilla implementation really possible, or even desirable?



The Oracle Experts  
Since 1988

Unlike a new car that can be purchased and driven right off of the lot, all of the Oracle E-Business Suite Applications require varying degrees of configuration prior to implementation. At a minimum, there are several required setup steps that have to be completed for each application. Some of these setup steps come "out of the box" with defaulted values, while others do not. The determination of how to configure the applications during this setup process is based on information gathered during the critical "requirements analysis" phase of an implementation project.

Some people may think that if they pursue a "vanilla" implementation of Oracle, then the requirements analysis can be minimized, or skipped altogether. Most people would agree that short-changing the requirements analysis effort would be inviting failure early in the implementation process. Yet, there are implementation approaches heralded by some IT consulting firms that propose to accelerate the configuration of the Oracle E-Business Suite applications by providing "pre-configured" applications or utilizing a "rapid" implementation approach. Choosing these "vanilla" implementation approaches will ensure that you get what *someone else* considers basic, but what will be missed by pursuing these methodologies? Because there are so many ways in which an Oracle application can be configured, and so many variations of best practice business processes, there is usually not just one way to get things done. There really is no substitute for a thorough requirements analysis at the beginning of every project. It is the only way to determine if "vanilla" is the flavor someone really wants!

Occasionally, the requirements analysis identifies "to-be" business processes that cannot be supported by "standard" Oracle application functionality. These "gaps" between the standard Oracle application functionality and a company's documented, unique business requirements introduce

common challenges to the implementation project. A "vanilla" implementation would require all of the proposed business processes to be changed to fit the standard functionality supported by a particular application. Many companies view this approach as a strategic opportunity to lower support costs (by eliminating or avoiding customizations) and to improve business processes (by implementing "best practices" supported by standard Oracle functionality). Some companies, however, have highly automated and refined processes, or cannot easily introduce change into their existing organizations. These companies have no option but develop workarounds, customizations, or extensions to the standard Oracle applications to meet their unique business process needs. So much for vanilla in these cases!

But what if there are no "gaps" identified, and all proposed business processes can be supported by the standard Oracle E-Business Suite applications without customization. Is this the ideal "vanilla" implementation that everyone seeks? Well, remember those required and optional setup steps? The documented requirements still have to be mapped to those setup steps, and this is where in-depth knowledge of the Oracle applications, and years of experience in configuring those applications to support the infinite variations of business processes really becomes important. This *configuration process* is where the investment in the Oracle E-Business Suite really pays off, if it is done correctly... appropriately taking advantage of *all* of the standard and advanced functionality provided by the applications. Cutting corners here to pursue a "vanilla" implementation can significantly lessen the positive impact that the Oracle E-Business Suite applications can have on a company's future business processes.

Let's take Oracle Receivables for example. To implement this application, there are 43 setup steps, 16 of which are required, and 27 are optional. Although several of the required steps have defaulted values that are pre-

seeded with the installation of the software, many do not, and must be specified during the configuration of the application. To complete all of the 16 required setup steps for even a simple implementation, almost 200 separate specifications need to be made to define critical system behavior such as payment terms, invoice numbering rules, auto-accounting, and other Receivables-related functions, to name a few. Merely accepting all of the defaults, or pursuing a "pre-configured" application approach to achieve a "vanilla" implementation can sub-optimize the available standard functionality, and in some cases, even limit your future options.

Additionally, there are dozens of descriptive flexfields (DFF) that can be utilized within the many Receivables application menus. Each DFF allows the application to be natively extended to support the tracking of unique transactional information, but even DFFs can get complicated. There sure seems to be a lot of ingredients that go into vanilla these days!

Some applications lend themselves to a more "vanilla" approach than others. Oracle Project Accounting can rarely be implemented using a "vanilla" approach, as the application is highly dependent on unique, client-specific configuration requirements. Payables, however, can be implemented with simple supplier definitions, basic accounting distributions, and straightforward approval hierarchies and payment processes. But does a "vanilla" implementation of Payables include setting up "positive pay" processes with a bank, the use of the Supplier Invoice API for automated entry of supplier invoices, third party integration for scanning of invoice documents, or interfaces to a supplier master "owned" by a legacy system? These standard business processes, utilized by many companies to improve efficiencies, increase accuracy, and enhance internal controls, require a more advanced configuration of the Payables application... well beyond what would be considered "vanilla" in most cases.

Is it too late to salvage the idea of a "vanilla" implementation? If not, here are some other common aspects of an Oracle E-Business Suite implementation to consider: data conversions and interfaces.

Virtually every implementation requires initial data conversions to populate the Oracle applications with both reference (customers, suppliers, etc.) and transactional ("open" invoices, G/L account balances, etc.) data. How do these data conversions impact a "vanilla" implementation? Often, it is the data conversions that have the greatest impact on the complexity of an implementation. For example, it is common for a company to change its Chart of Accounts when implementing Oracle General Ledger. This change can impact not only the complexity of converting and transforming legacy general ledger system account detail and summary balances, but also impacts the accounting distributions of transactions in other sub-ledgers such as Purchasing, Payables, and Receivables. It is also common for a company to "clean up" its Customer or Supplier master during an implementation. Oracle provides some tools to assist in this data cleansing activity, but is this effort part of a "vanilla" implementation?

Unless a company is implementing all application modules within the E-Business Suite, there will likely be interfaces to legacy systems or other third party data providers. Although Oracle provides dozens of standard APIs and interface tables to make integration with non-Oracle external systems as simple as possible, the implementation of interfaces generally requires effort beyond the elusive "vanilla" approach. Most likely there will need to be data transformation and validation in the interface process; if not on the Oracle side, then on the external system's side. To shield Oracle from data integrity issues, inbound interfaces may need to handle data validation using business rules incorporated into PL/SQL code, or perform data translation using mapping tables. Certainly, the use of external

system interfaces can easily transform a "vanilla" implementation into significantly more complex flavors like cherry cheesecake or chocolate peanut butter swirl.

So, is a "vanilla" implementation even possible? Of course it is! It is just extremely rare. Even a seemingly simple implementation, one that uses only standard Oracle functionality, can get complicated quickly when you add on dozens of interfaces, tricky data conversions, several descriptive flexfields, a complex Chart of Accounts, or integration with external systems. However, if the purpose of *pursuing* the elusive "vanilla" implementation is to adopt standard business processes and avoid the need to customize the applications, then it is a worthy *goal*. Pursuing a "vanilla" implementation may not necessarily equate to a simple implementation, won't negate the need for a thorough requirements analysis, and should not involve ignoring all of the advanced functionality provided by the Oracle E-Business Suite. Even pursuing "vanilla" can require a significant investment of effort!

And that is the most important point to remember. All implementation projects require a significant investment of time, money, and effort. The Oracle E-Business Suite is designed to let people get the greatest return on this investment. Implementing such a world-class suite of software *without* taking the time to understand, evaluate, and take advantage of the many standard and advanced features of the software can lessen the potential return on this strategically critical investment. With the proper analysis, planning, and execution, and TUSC as their implementation partner, they can achieve a cost-effective, efficient, successful implementation of the Oracle E-Business Suite, with whatever flavor they really want to have, even if it is vanilla!

#### **ABOUT TUSC**

Specialists in helping companies optimize their investment in Oracle Technology. Providing integrated functional and technical solutions since 1988 in the areas of Oracle's E-Business Suite, Business Intelligence / Data Warehousing, Custom Development, Managed Services / Remote DBA, Database Services, Training & Mentoring and Oracle Licensing.

Oracle Certified Partner  
Technology & Applications

We Deliver Value

Toll-Free  
800.755.TUSC

On the Web  
[www.tusc.com](http://www.tusc.com)