



Whitemarsh  
Information Systems Corporation

Section of  
Make, Buy or Generate

Whitemarsh Information Systems Corporation  
2008 Althea Lane  
Bowie, Maryland 20716  
Tele: 301-249-1142  
Email: [mmgorman@wiscorp.com](mailto:mmgorman@wiscorp.com)  
Web: [www.wiscorp.com](http://www.wiscorp.com)

## 1.0 Introduction

Once the enterprise has completed the mission and a high level of the data model (entities only, without attribution), it is ready to start discrete projects for easily separable areas. An easily separable area is one that is either a significant originator of data or one for which the data is quite self-contained.

Implementing through pragmatic units versus the *big bang* approach allows for the benefits of database to be realized quickly while retaining a good measure of control and flexibility.

The necessary first step in each of these pragmatic projects is to complete the seven models. Completed then will be a mission-area-scoped, detailed data model, database process, information systems, business event, business function, and business organization. Once this is complete, a *make, generate, or buy* decision can be made concerning functionally acceptable application packages.

For the purposes of definition, *make* means the complete software development route, that is, detailed design, coding, and unit testing along with overall system testing and integration. *Generate* means that the metadata from the seven models and security can be readily fed into a sophisticated software package and that the complete software system is generated through vendor-provided software generation facilities. While individual tools may exist for screen design, report specification, and the like, *generation* here means the entire software application: screens, menus, calculations, reports, and so forth. Under this technique, knowledge workers develop, to a large extent, the first seven models, and the generator creates the remaining three models.

*Buy* means that the application system is ready to use once it is installed. The PC has made this market. Simply, the package represents all 10 of the models. PC based packages are available for almost every imaginable purpose: finance, human resources, word processing, spreadsheets, telecommunications, scheduling, project management, and so on. There are two big differences among all these packages:

- ! Single user versus multiple user
- ! The breadth, scope, and extent of tuning

The older and more complicated the business is, the greater are the requirements in these two areas. For the PC, single-user, simple accounting systems (G/L, A/P, A/R, Inventory, and Cash) can be purchased for \$150. For the client/server environment, multiple-user, very sophisticated packages can cost well over \$250,000.

Variations on the make, generate, or buy alternatives are becoming available. For example, rather than just being able to buy a generator, and then creating and feeding it with the organization's developed seven models, some software vendors are creating fully developed, functionally rich application packages with generators, and then marketing not the finished package but the generator and the installed metadata. In this form, the organization merely has to modify the generator-installed metadata and regenerate the application systems environment. This, for example, is the business strategy of Oracle Corporation with its human resources, financial,



and manufacturing applications. For Oracle, the generators obtain all their metadata from Oracle/CASE.

If the make, buy, or generate decision is made too early, there will not be sufficient knowledge to correctly discriminate among the packages. As evidence, many organizations evaluate application packages solely based on the vendor's ability to meet their functional requirements (e.g., provide reports, promote end user access). Rarely, for example, are the package's data structures examined to determine if compatibility exists with the organization's way of doing business. Because of too-early picks, it is not uncommon to hear of a business buying a package and then spending from two to ten times the package's purchase price to have the package *fit the business*.

Once a package is picked, for example in the financial area, it can take up to 10,000 staff hours to finally install the package. Why so long as compared to a word processing package? After all, are not both commercial-off-the-shelf (COTS) packages? It is because of the quantity of different sets of reference tables (from 20 to 60), the number of specialized reports (100 to 200), the number of pair-wise validations (on a four-part validation it can be thousands and thousands, which all have to be examined, determined, and set into place within the package).

It is important to understand the role of a COTS package. Simply put, if the business has no real position on how a business function is accomplished, then the COTS package becomes the defining strategy and implementation vehicle for the business function. This would be the case if a business was new, and if COTS packages were procured right from the beginning.

For most large businesses, however, procuring a COTS package is either an alternative to large-scale systems development or an alternative for a large-scale systems upgrade, for example, when migrating from mainframe to client/server. In this latter case, it is important to understand how few work steps are actually avoided. In terms of time and funds, a great deal is avoided. A COTS finance package could take as much as five years to develop with a team of 15 to 20. That would be a cost of well over \$15 million. A COTS package changeover alternatively could be accomplished in two years or less with about 10 staff. That would be a cost of about \$3 million. The savings is *potentially* substantive, and the time saving is 60 percent. However, if all the work steps are not accomplished, the whole effort could well be wasted. Appendix 2 contains a listing—at a very high level—of the critical tasks involved in a complete database project. *Italicized* are those steps that can be avoided by taking the COTS route. While the quantity of italicized steps seems few, they are time-consuming and costly indeed.

