SOFTWARE ACQUISITION AND DEVELOPMENT ALTERNATIVES: 
AN INVESTIGATION OF PREFERENCES

Roger L. Hayen, Central Michigan University, roger.hayen@cmich.edu

ABSTRACT

The early initial applications of computers to business processes required the custom development of application software. Since those beginnings, the alternatives for acquiring application software have expanded greatly with numerous alternatives now available. Acquisition approaches by business organizations were investigated to determine current preference for acquiring their application software. A survey was conducted and analyzed among companies representing a variety of different industry sectors. Respondents from these companies generally agree that the preferences for acquiring software is to first purchase and configure the software and is to last engage in the custom development of software. The results apply across three major software categories delineated in the investigation.

Keywords: Software acquisition, software development, software alternatives

INTRODUCTION

A conversation with a senior executive of a major chemical manufacturing company revealed that his preferences, for acquiring software for his company, were first to buy it, install it and use it; and last to build or custom develop it. Is this preference similar to that held by other business organizations? When computers were initial applied to business processing, all application software was custom developed. Since those beginnings, the alternatives for acquiring software have expanded. Many of those earlier applications are now referenced as legacy systems, whereas many of the current purchased or packaged application suites are known as enterprise software (ES) or enterprise resource planning (ERP) software. ES or ERP software integrates transaction-oriented, core business processes. For example, the SAP R/3 System is the most widely used ES that links an entire organization together with one comprehensive system. ERP encompasses a completely integrated collection of software programs, which covers all of the major functions of an enterprise [1, 2]. SAP AG competes with other ES vendors including PeopleSoft, Oracle, J.D. Edwards, and Baan in this integrated software segment of purchased software. The focus of this research investigation is to examine the software acquisition preferences held by business organizations. These preferences can then be considered by other organizations in their ongoing management of a software portfolio. The findings are presented here in the order of software acquisition methods, purchased software design concepts, and an acquisition and development evaluation.

SOFTWARE ACQUISITION METHODS

An important question every business organization must address is how software capabilities should be acquired. Ragowsky and Stern [3] delineate three main alternatives for acquiring application software, which are (1) developing a custom application, (2) installing a purchased package, and (3) modifying a package to meet an organization’s specific requirements. They
point out that when an organization purchases a package and modifies it, it is no longer a “package.” Unless those modifications are included in the next release of the package, it is necessary to replicate those modifications. This is an on-going maintenance issue that may easily be as costly as the initial modifications to the software. In today’s software environment, the installation of a purchased package can be expanded from two to five methods. Three additional ways, in which purchased applications may be tweaked, are (1) by its configuration, (2) by the development of customer routines that provide additional functionality, or (3) by the development of a front end or back end to the package to include customer routines. These three alternatives are different from modifying a package in that the code provided by a package’s vendor is not modified. Rather, any customer developed routines are external to the vendor supplied code. This leads to the six acquisition alternatives delineated in Table 1.

Table 1: Acquisition Alternatives

<table>
<thead>
<tr>
<th>Alternative</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a Purchase</td>
<td>Software is bought, installed, and used based on the standard functionality of the software package</td>
</tr>
<tr>
<td>b Purchase and configure</td>
<td>Software is configured by making table settings that specify the functionality selected for a company.</td>
</tr>
<tr>
<td>c Purchase and develop customer routines</td>
<td>Software is provided additional functionality through customer developed routines that are readily integrated using an interface provided in the purchased software.</td>
</tr>
<tr>
<td>d Purchase and develop front end and/or back end</td>
<td>Software requirements are met by customer created processes that precede or follow the processing by the packaged software without any processing interfaces provided in the packaged software.</td>
</tr>
<tr>
<td>e Purchase and modify</td>
<td>Purchased software is modified or altered to a company’s specific processing requirements by direct changes to the code supplied by the software’s vendor.</td>
</tr>
<tr>
<td>f Custom develop</td>
<td>Software is custom developed by the using company.</td>
</tr>
</tbody>
</table>

This set of alternatives can be applied to several different areas or dimensions of software acquisition. For the purposes of this investigation, the dimensions are organized as (1) core business software, (2) desktop software, and (3) system software. Core business software is defined as the main transaction processing and related software, which includes applications such as financials, account receivable, accounts payable, order processing, inventory, payroll, and human resources. Desktop software is personal productivity software that is available on most, if not all, of the desktop personal computers in a company. This software would include word processing, spreadsheet, presentation, and database software, which are commonly included in an office suite of software. System software is operating system software and would include web-server and email software. Each of these categories may follow a different strategy for their acquisition. It is these three categories, which are examined using the six acquisition alternatives in this investigation.

Purchased software design concepts

Purchased software is a computer solution that is typically bought from a vendor, installed on a customer’s computer, and used to process the customer’s transactions (Figure 1). Under this situation, all the functionality of the software is provided by the vendor with little or no flexibility in the processing performed by the purchased software. However, customers often need more flexibility than the processing provided by vendors. Enhance flexibility is achieved frequently using several design concepts in its development. Four of those concepts, most
relevant to this investigation, are (1) table-driven applications, (2) configurable software, (3) user exit processes, and (4) front-end and back-end processes. Each of these concepts is described to provide a context for the software acquisition alternatives examined in this investigation.

A **table-driven application** makes use of relatively static data that is stored in a file or database table and is used in controlling processing within the application software. A simple example of this is an income tax table used to determine the appropriate tax rates used in payroll processing. Any changes to the tax rates are made to the externally stored table, and not to the internal program code itself. Another example is an *.INI file, this is used frequently with operating system software and desktop software. A somewhat more complex example is that of a rollup table, which is used regularly in the consolidation of business unit application data. A rollup table specifies the relationship among business units in an organization hierarchy. An organization structure is stored externally to the program code as a relatively static or persistent specification. Changes to an organization structure are made by revising the table entries. Table-driven design greatly enhances the flexibility of an application by storing relatively persistent data external to the program code.

**Configuration** is a method by which user selections and persistent data are specified for purchased software. This approach is used with ES, such as the SAP R/3 System and the Peoplesoft software, to setup specific processing needs. Configuration includes the specification of the persistent data that delineates an organization structure and a chart of accounts. That is, the persistent data for table-driven software. With ES, configuration also includes the selection of the processes themselves that will be used in its implementation. The details of the configuration may also include the specification of the available fields displayed on an interactive screen. Configurable purchased software makes extensive use of table-driven design with external tables holding persistent configuration specifications.

**User exits** are functions and/or subroutines that provide an interface to a user-defined process. A user exit provides the ability to use a standard interface provided in the purchased software to call an external function that is executed and then to return to the purchased software for continued processing. This feature permits an organization to develop a customized processing routine and to interface it to the purchased software package without the need to make any direct changes or modifications in the purchased software.

**Front-end or back-end processing** are customer developed routines that process transaction data by accessing the database of the purchased software and directly manipulating that data. The results from this processing are then available to the standard processing provided by the purchased software. Because the purchased software does not support the ready integration of customer developed routines into its program processes, such as that available with the interfaces provide by user exits, the customer developed routines access the user transaction data and...
process it directly. This processing method avoids the need to directly modify the program code of purchased software.

The various methods presented here are not mutually exclusive. Clearly, they can be used in any combination. A particular installation of purchased software could be configured, employ user exits for customer unique processing, and utilize front-end and back-end processing for customer unique processing requirements.

ACQUISITION AND DEVELOPMENT EVALUATION

A research survey was completed by 26 respondents for this assessment of software acquisition and development preferences. These respondents were from member companies of a management information systems (MIS) advisory board at the author’s university. They were primarily from larger organizations. The industry sectors represented include chemical manufacturing, insurance, health care, automotive manufacturing, and other manufacturing. Participants were asked to respond to questions in which they ranked the desirability of alternative methods for obtaining software and provided their perceptions of several overall issues in the acquisition of software. Software was organized into the three categories of (1) core business software, (2) desktop software, and (3) system software. Within each category, the six specific acquisition alternatives (Table 1) were designated with an “Other” alternative as a seventh possible designation. Respondents ranked the desirability of each method of obtaining software with 1 as the highest priority or most desirable through 7 as the lowest priority or least desirable. When respondents did not specify the “Other” alternative, then they used only a 1 through 6 ranking of the alternatives. No respondents indicated an “Other” alternative, which they ranked, so it is disregarded in this analysis. On the survey instrument, each of the three categories is arranged as a separate question with each question containing the acquisition alternatives (Table 1) arranged for the ranking processes. Thus, question 1 is for core business software; question 2, for desktop software; and question 3, for system software. Within each question, each of the acquisition alternatives is identified by a letter. This reference method is applied in the presentation of the results.

Figure 2 shows the mean result for the software acquisition alternatives for the core business software category. Purchase & Configure with a mean of 1.5 is the first preference, whereas Custom Develop with a mean of 5.3 is the last preference for this software.
acquisition. For this category, the two alternatives of Purchase and Purchase & Develop Custom Routines have an identical preference with a mean of 2.7. Basically, this indicates the participant’s companies are indifferent to the two alternatives. Or, another view is that they desire to purchase software and use it as is, but if they cannot do that, then they are equally satisfied with purchasing the software and developing custom routines that are readily interfaced with the purchased software to provide additional functionality required to meet their needs.

The chart in Figure 3 displays the mean results of the acquisition alternatives for desktop software. Purchase & Configure with a mean of 1.5 is the first preference, whereas Custom Develop with a mean of 5.8 is the last preference for this software category. The Purchase alternative with a mean of 1.7 indicates this alternative has about the same preference as the Purchase & Configure alternative. The configuration of desktop software frequently involves the use of settings and/or templates. These results indicate that some companies have a very strong desire just to use this software as delivered.

Figure 4 portrays the mean result for the alternatives for operating systems software. Once again, Purchase & Configure with a mean of 1.5 is the first preference, whereas Custom Develop with a mean of 5.7 is the last preference for this software category. The Purchase alternative with a mean of 1.7 indicates this alternative has about the same preference as the Purchase & Configure alternative. The configuration of system software frequently involves the use of initialization settings. These results suggest that some companies
have a very strong desire just to use this software as delivered. The comparison, of the chart in Figure 3 for desktop software to that in Figure 4 for system software, suggests the preferences for the acquisition of these two categories of software are very similar. Essentially, the preference is to use the software as provided by a vendor with little or no need for make adjustments in order to match the business requirements for this software.

Participants were asked to respond to several overall acquisition perception questions using a 7-point Likert scale, where 7 indicates “strongly agree,” 4 “don’t know/no opinion” and 1 “strongly disagree.” To help minimize response bias, question 7 was negatively stated. For analysis purposes, this question was reverse-coded to facilitate an easier comparison to the other responses. The item is stated in reverse-coded form in Table 2, which shows a list of other software acquisition issues included in this survey.

Table 2: Other Software Acquisition Issues

<table>
<thead>
<tr>
<th>Item</th>
<th>Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>The trend in my company is to purchase more of the core business software and to write less custom developed code.</td>
</tr>
<tr>
<td>5</td>
<td>The core business processes of my company are unique and require the use of custom developed software.</td>
</tr>
<tr>
<td>6</td>
<td>Purchased core business software reduces the total cost of software ownership including acquisition and maintenance.</td>
</tr>
<tr>
<td>7</td>
<td>Purchased core business processing software increases the competitiveness of my company. *</td>
</tr>
<tr>
<td>8</td>
<td>Desktop software is now a commodity that is purchased, because there is no competitive advantage from custom developed desktop software.</td>
</tr>
<tr>
<td>9</td>
<td>For desktop software, unique application requirements are met by using templates as the primary means of customizing these applications in my company.</td>
</tr>
<tr>
<td>10</td>
<td>System software is usually purchased, because of the specialized nature of this software.</td>
</tr>
</tbody>
</table>

* Indicates a reverse-coded question

Figure 5 highlights the average responses for questions 4 through 10. Overall, these responses reveal agreement with these acquisition issues. Question 4, with the highest level of agreement for all the questions, indicates there is a very strong trend in the respondent’s companies to purchase more of the core business software and to write less custom developed applications. Respondents were mostly neutral to the issues in questions 5 and 6, concerning the issue of unique business processes and the need for custom development. This neutral position may reflect the situation, that for some of the respondent companies, their information
system is essentially their production system, such as in the case of insurance companies. For other companies, their information systems support manufacturing or other operations, in which the information system is not their primary production system. In general, the respondents suggest the perception of their company’s total cost of ownership is about the same for purchased core business software as for custom developed applications. As portrayed in Figure 5, questions 7 through 10 indicate respondents generally agree that purchased software increased their company’s competitiveness, that desktop and system software are essentially considered as commodities that are purchased, and that templates are used to satisfy unique requirements with desktop software.

Additional feedback was elicited from respondents in an open-ended question. These comments indicate purchased software is an essential consideration of their companies. For example, one respondent stated, “Purchase software that is able to meet 80% of my requirements, and has an open architecture to allow for the completion of the additional 20% of requirements.” Another respondent commented, “Configuring a purchased application is preferred over having to modify the application code. Building an application should only occur when a purchased solution does not exist to support your business need.” Generally, these comments support the ranking of the software acquisition alternative, and they support the viewpoint that a combination of those alternatives is preferred to a single, particular alternative.

**SUMMARY AND CONCLUSION**

Based on these results, the study found considerable evidence to indicate that the first choice in software acquisition is to purchase and configure it and that the last choice is to custom develop it. The application of configuration, user exits, and front-end and back-end processing is preferred to the custom development of software. This seems to stem from the condition that these three alternatives avoid the direct modification of the purchased application’s program code. In general, the same acquisition alternatives are preferred across the three categories of core business software, desktop software, and system software. The implications of this investigation are that business organizations should specifically consider these alternatives in planning and evaluating their organization’s software portfolio. The academic community should consider the implications of these acquisition alternatives in planning course curriculums for individuals pursuing an education that would involve them in the acquisition and implementation of purchased software. Further investigation of acquisition alternatives should be considered to provide added guidance in these processes.

**REFERENCES**