Prototyping: An Alternative Approach To Systems Development Work

Stephen A. Moscove, (Email: smoscove@charger.newhaven.edu), University of New Haven

Abstract

The traditional approach to performing systems development work on a company's accounting information system is initially discussed in this article. It encompasses analysis, design, implementation, and follow-up work. Following the discussion of the traditional approach, an alternative, and less used, approach to systems development work is then examined. This approach is called prototyping, whereby a simplified working software model of a company's accounting information system is developed. After discussing how prototyping works, both the advantages and disadvantages of this approach to systems development work are analyzed. To test empirically the traditional and prototyping approaches, this author worked with a consulting firm on three different companies systems problems. Both of these systems development approaches were used in each company to help solve its problems. The results from the systems development work on the three companies appear to indicate that using either of the two approaches will lead to the same major characteristics in companies' newly designed and implemented information systems.

Introduction

The traditional approach to performing systems development work for revising a company's accounting information system (AIS) is as follows: (1) Analysis--reviewing current system to recognize strengths and weaknesses, (2) Design--developing a revised system to eliminate the current system's weaknesses, (3) Implementation--putting the new system into operation, and (4) Follow-Up—evaluating the revised system periodically and making further revisions when problems exist.

This traditional approach is normally a very costly process and thus may not be cost effective for certain organizations to use. This paper will examine an alternative approach to developing an AIS for a company. The approach is called prototyping, which involves the development of a simplified working software model of the AIS requested by a company's users. Through an iterative process, company users experiment with the software model until a system is obtained that meets their needs.

The next section will briefly review the traditional approach and the prototyping approach to systems development work. This will be followed by a short discussion of the advantages and disadvantages of using prototyping in systems development work. Following the discussion of advantages and disadvantages of prototyping, an example will be provided of prototyping's potential use in the cost management area of accounting when considering revisions in a company's cost accounting system. Finally, the last section will attempt to answer the following question: In designing an AIS for a company, is it likely that the use of either prototyping or the traditional approach will

Readers with comments or questions are encouraged to contact the authors via e-mail.

lead to the same major characteristics in the newly designed and implemented information system?

Traditional and Prototyping Approaches to Systems Development Work

Two major approaches to performing systems development work are reviewed in this section of the paper.

Traditional Approach

According to Moscove, Simkin, and Bagranoff (2001, p. 349), the traditional approach to systems development work encompasses four major phases, as discussed below.

The Analysis Phase. Under this phase, the problems within a company's current AIS are defined in relationship to the goals established for the company's system. A systems survey is performed to acquire sufficient information relating to the current systems problems. In performing the systems survey, the strengths and weaknesses of the current system are identified. The strengths will indicate which components of the current system should be retained in a revised system, whereas the weaknesses will hopefully indicate what is causing the problems within the current system. In concluding analysis work, possible solutions to solve the systems problems should be suggested to the company's management.

The Design Phase. In this phase, a feasibility evaluation is performed on different possible AISs that can replace the company's current system. A major part of the feasibility evaluation is to attempt to monetarily quantify the anticipated costs and benefits of each possible AIS being considered. The most cost-effective information system is the one whose anticipated benefits are greatest in relation to the system's anticipated costs. In projecting the costs and benefits of each proposed AIS, discussions should take place with computer vendors such as IBM or Digital Equipment.

The Implementation Phase. Upon designing a cost-effective AIS to replace the company's current system, this revised system must then be implemented. This phase of systems development work involves the planning and controlling of the various activities that must be performed to implement successfully the revised system. Examples of these activities are selecting and assigning personnel, training personnel, establishing controls for the system, and acquiring and testing the computer programs. A tool such as a PERT network diagram can be quite useful in helping to logically schedule each systems implementation activity.

The Follow-Up Phase. After the revised AIS has been in operation for a specific time period (e.g., two months), the system should be evaluated to determine its effectiveness in meeting the needs of the company's management. Examples of activities performed in follow-up work include talking with management personnel about their satisfaction with the output reports received from the revised system, evaluating the controls of the system to verify whether they are functioning properly, and evaluating whether computer-processing functions are being performed efficiently and effectively.[1]

It should be emphasized that any weaknesses in the revised AIS discovered through follow-up should lead to further systems development work that hopefully eliminates these weaknesses. This work involves considering additional revisions in the system that cause the weaknesses to disappear.

A study of a few years ago sponsored by the American Accounting Association (AAA) made the following observation (1996, p.139):

As organizations become more and more concerned about the costs of efficient and effective operations, the traditional approach to systems development work may not be the one to use. Other more cost-effective approaches should be considered when revising a company's accounting information system.[2]

Based on the above observation by the AAA, one of the possible alternative approaches for

performing systems development work, prototyping, is now examined.

Prototyping Approach

As pointed out by Teagan and Young, an alternative approach to systems development work that has been minimally employed in the business world is prototyping (1995, pp.53-58)[3]. Using prototyping, a simplified working software model, or prototype, of a company's AIS is developed. This prototype is a scaled-down, experimental version of the information system requested by the company's users. The initially developed prototype is built quickly at minimal expense and is provided to users for testing purposes. By allowing users to experiment with the prototype, they can provide feedback to the developers of this prototype regarding what they like and dislike about their company's AIS. Based on user feedback, the prototype developers will modify the information system model and present it to the users for additional testing and feedback. Thus, an iterative process of trial use and modification continues until the users are satisfied that the information system properly meets their needs.

The four major prototype development steps are briefly analyzed below:

Step 1: Identify Information System Requirements. The first step in developing a prototype is to identify the fundamental requirements of a company's information system (whether an accounting information system, a marketing information system, etc.). To accomplish this task, the system developers (whether management consultants or systems professionals working for the company) meet with the information systems users. The purpose of this initial meeting is to reach an agreement between the developers and the users regarding the size and scope of the information system as well as what should be included and what should be excluded from the proposed information system. By the conclusion of the meeting, the system developers must ascertain from the users what outputs the users need for decision making as well as the data inputs required to generate these outputs. In obtaining this information from users, the focus of attention is on what outputs should be generated rather than on how they should be generated. At this stage of the prototyping approach, the system developers must make sure that the users' expectations of what benefits they can obtain from the information system are realistic and that the users' basic information requirements can be satisfied by the system.

Subsequent to the meeting with the information system users, the system developers will provide the users with cost estimates (and other feasibility estimates, such as technical feasibility) for several alternative information systems that appear to meet the users' needs. Based on these estimates, the users should make a decision concerning whether the systems development project should be continued. Since only fundamental requirements are identified for the initial prototype, the process of determining these requirements for the prototype is less formal and time-consuming than when a traditional systems design approach is used. The system users eventually develop the detailed requirements of the information system as they interact with the prototype.

Step 2: Develop Initial Prototype. In this step, an initial prototype of the information system that meets the users' fundamental requirements (established in step 1) is developed. Examples of tools that can be used for developing prototypes are fourth-generation programming languages, CASE tools, and databases. The system developers, in building this initial prototype, have speed of implementation and low cost as their goals. Thus, for example, the area of computer application controls may be given little, or no, consideration so that characteristics such as simplicity, flexibility, and ease of use can be emphasized in the initial prototype. These characteristics will enable the system users to see and use tentative versions of data-entry display screens, menus, input prompts, and source documents. The users also need to be able to respond to prompts, make inquiries of the information system, judge response times of the system, and issue commands.

Upon finishing the initial prototype of the information system, the system developers provide a demonstration to the users of the prototype's use. Following this demonstration, the prototype is given to the system users for experimentation purposes. From the inception of the users' testing procedures, they should be aware that the prototype is incomplete and will require subsequent modifications. The users should be instructed that during their experimentation with the information system, they should list their likes and dislikes about the system and the resultant changes that they recommend. Finally, the users should be informed that, upon finishing their lists, they should notify the system developers to enable them to introduce the desired changes into the prototype.

Step 3: Iterative Process. After the system developers make the necessary changes in the initial prototype of step 2, the revised prototype of the information system is again turned over to the system users for experimental purposes. This iterative process of system users identifying changes, system developers making changes, and the revised prototype of the system returned to the users for experimentation continues until the users are satisfied with the information system prototype. User satisfaction causes the prototype to be approved. It is common for a prototype to go through four to six iterations.

Step 4: Use of Approved Prototype. Approximately half of the approved prototypes of organizations are turned into fully functional information systems, called operational prototypes. (The approximate other half of approved prototypes are determined to be not practical and are called nonoperational, or throwaway, prototypes.) To make a company's approved prototype of an information system operational, the system developers must normally make the following types of changes in the prototype: incorporate needed control procedures, improve operational efficiency, provide backup and recovery, and integrate the prototype with the company's other systems with which it interfaces. If necessary, changes must also be made in the prototype so that the information system will accept real input, access real data files, process real data, and generate real output.

Advantages and Disadvantages of Using Prototyping

Compared to the traditional approach to systems development work, the use of prototyping has both advantages and disadvantages.

Advantages of Prototyping

One important advantage of prototyping is that its use requires intensive involvement by the system users. Therefore, prototyping typically results in a better definition of these users' needs and requirements than does the traditional systems development approach.

A second advantage is that a very short time period, such as a week, is normally required to develop and start experimenting with a prototype. This short time period allows system users to immediately evaluate proposed system changes. In contrast, it may take a year or longer before system users can evaluate proposed system changes when the traditional systems development approach is used.

Since system users experiment with each version of the prototype through an iterative process, a third advantage of prototyping is that errors are hopefully detected and eliminated early in the developmental process. As a result, the information system ultimately implemented should be more reliable and less costly to develop than when the traditional systems development approach is employed.

Disadvantages of Prototyping

Prototyping can only be successful if the system users are willing to devote significant time in experimenting with the prototype and provide the system developers with change suggestions. This leads to a potential disadvantage of prototyping since the system users may not be able or willing to spend the amount of time required under the prototyping approach.

Another disadvantage relates to the fact that the iterative process of prototyping causes the prototype to be experimented with quite extensively. Because of this, the system developers are frequently tempted to minimize the testing and documentation process of the ultimately approved information system. Inadequate testing can make the approved system error-prone, and inadequate documentation can make this system difficult to maintain.

Finally, prototyping may cause behavioral problems with system users. These problems include dissatisfaction by users if system developers are unable to meet all user demands for improvements as well as dissatisfaction and impatience by users when they have to go through too many iterations of the prototype.

Prototyping and the Cost Management Area of Accounting

The cost management area of accounting encompasses the development of effective cost accounting systems for production environments that provide useful information for decision making by organizations' managers and others. Within a company's production environment, both internal and external users of the company's AIS need cost information. Internally, for example, managers use information about costs to reach decisions regarding product pricing and product mix and to evaluate operating performance. And, externally, costs should be properly matched with revenues in preparing financial statements for users such as potential investors and creditors.

Traditionally, most companies' cost accounting systems have been designed and implemented primarily to meet financial reporting requirements and have given only secondary attention to meeting the needs of managers for decision-making purposes. As a result, traditional cost accounting systems have been criticized in recent years for not providing adequate information to manage production operations in a modern manufacturing environment. Two of the major criticisms of traditional cost accounting systems are: (1) overhead costs are inappropriately allocated to products, and (2) performance measures do not accurately reflect the effects of factory automation.

To overcome criticism (1) above, many accountants feel that both job-order and process cost systems can be refined and significantly improved by adopting activity-based costing (ABC) systems. An underlying objective of ABC systems is to link costs to organizational strategy, which leads to decisions about what goods and services to produce. Activities must be performed to produce these goods and services, which in turn incur costs. Thus, organizational strategy determines costs. Consequently, by measuring the costs of basic activities, such as processing purchase orders, activity-based costing provides information to management for evaluating the consequences of strategic decisions.

Developing ABC Systems Through Prototyping

ABC systems cost more to run than traditional cost systems because these systems require the collection of more production-related data, and in greater detail. In addition, ABC systems are more complex than traditional cost systems. This is due in part because more bases are used to allocate overhead costs. Furthermore, ABC systems dictate several requirements on a company's AIS. First, a company converting to ABC must redesign its general ledger to fit the additional cost categories used by the ABC system. Second, conversion to an ABC system requires extensive use of information technology in order to accumulate more precise data about cost drivers. Third, and possibly most important, conversion to an ABC system requires that both financial and nonfinancial measures of production activity be stored in an integrated manner.

Because of the extensive systems development work involved in designing and implementing an ABC system to replace a company's traditional cost system, the prototyping approach compared to the traditional systems-development approach appears to be a more cost-effective approach to use in developing the ABC system. Through prototyping, a scaled-down, experimental version of

the ABC system could be developed quickly at minimal expense and provided to users of the system for testing purposes. Thus, eventual users of the ABC system would be intensively involved in the entire prototyping process, hopefully leading to a better definition of the users' needs and requirements from the ultimately implemented ABC system.

With the prototyping approach to developing an ABC system, a reasonably short time period would typically be required to develop and start experimenting with the ABC system prototype. As a result, eventual users of the system could evaluate proposed changes in a timely fashion. Finally, since eventual ABC system users will be using an iterative process to experiment with each version of the prototype, errors associated with the proposed ABC system will likely be detected and eliminated early in the systems development work process. Therefore, the ABC system finally implemented should be more reliable and less costly to develop than if the traditional approach to systems development work were utilized.

A Systems Development Practical Experiment

An important question that this article is attempting to answer is the following: In designing an AIS for a company, is it likely that the use of either prototyping or the traditional approach for performing systems development work will lead to the same major characteristics in the newly designed and implemented information system?

During the fall of 2000, this author spent considerable time working with a consulting firm in the Midwest that performs systems development work for its clients. For three different companies (medium-size organizations) having systems problems, the consultants used both the prototyping approach and the traditional approach to help each company solve its problems.

Presented below is a brief summary of the three systems development projects worked on by the consulting firm (and this author). At the request of the companies, their real names have been excluded from the presentations. For Companies A and B, the use of both the prototyping and the traditional approaches to systems development work resulted in the same major characteristics in the newly designed and implemented information systems. For Company C, the utilization of the traditional approach resulted in the company's management failing to accept the systems revision suggestion. However, the use of the prototyping approach led to the same systems revision suggestion (as under the traditional approach) now being accepted by the company's management.

Company A: This company (in the insurance industry) had previously contracted for a new computerized system that was designed to handle all aspects of its business operations. This system, which was developed by members of the company's technical staff who did not understand the company's business requirements, failed to work properly once it was implemented. For example, the system sent hundreds of checks to a nonexistent town, made significant numbers of dollars of overpayments, and caused the loss of many clients.

Recommendation by Consultants: Using both the prototyping and traditional approaches to systems development work, the consultants discovered that one of the major reasons for the new computerized system's failure was that its implementation should have included a restructuring of the organization. When this restructuring was finally performed (as recommended by the consultants), one outcome was that three layers of management personnel were removed.

Company B: This company (in the manufacturing industry) had a decentralized purchasing system that effectively met the needs of its several manufacturing plants. But, the company was unable to take advantage of its extensive buying power to negotiate quantity discounts on purchases.

Recommendation by Consultants: Using both the prototyping and traditional approaches to systems development work, the consultants determined that a reengineering of the company's pur-

chasing system was needed. Through the reengineering process, a corporate wide purchasing department that developed and maintained a shared database of approved vendors was introduced. Each plant continued to meet its inventory needs by making its own purchases from the database of approved vendors. The corporate office tracked the purchases of all the manufacturing plants, negotiated quantity discounts, and handled disputes with vendors. The reengineered system resulted in a significantly lower cost of inventory purchased, a 50 percent reduction in lead times, and a 150 percent improvement in on-time deliveries.

Company C: This company (in the healthcare industry) realized that its various systems that were developed in the 1970s and 1980s only automated its manual processes. The company wanted to utilize new technologies to link its many systems with all internal and external systems and their users.

Recommendation by Consultants: Under the traditional approach to systems development work, the consultants recommended a revised system using image processing. However, they were unable to "sell" the company's middle managers on the positive outcomes associated with image processing. The managers thought of image processing as just replacing file cabinets. But, when the prototyping approach to systems development work was employed and a prototype of image processing was developed by the consultants and observed by the company's middle managers, these managers caught on to the advantages of image processing. They realized the business potential of image processing and it was subsequently implemented by the consultants into the company's revised system.

Concluding Comments and Suggestion for Future Research

The empirical data obtained from the systems development work on the previously discussed companies appear to indicate that using either the prototyping approach or the traditional approach will lead to the same major characteristics in companies' newly designed and implemented information systems. In fact, looking at the outcome described above for Company C, the prototyping approach is possibly the best approach for consultants to use in obtaining approval by a company's managers of their systems revision suggestions.

Based on this author's limited empirical data from the three companies in the study, it does appear that prototyping is appropriate to use in systems development work when one or more of the following conditions exist in a company: system users do not understand their information needs very well, system requirements are difficult to define, the system to be developed is critical and needed quickly, or the risk associated with developing and implementing the wrong system is high.

On the other hand, it appears that prototyping may not be appropriate for use in designing large or complex information systems that serve many users throughout an organization having significantly different needs for planning, control, and decision-making information. Furthermore, prototyping will not likely be used for developing traditional AIS applications such as accounts receivable, accounts payable, payroll, or inventory management.

Because of the small number of companies included in this study, additional empirical data definitely need to be accumulated on the prototyping and traditional approaches to systems development work before the above conclusions can be drawn. The data can be acquired by analyzing these two approaches to systems development work in additional companies beyond those examined in the current study.

Footnotes

1. This summary of systems development work comes from Stephen A. Moscove, Mark G. Simkin, and Nancy A. Bagranoff, *Core Concepts of Accounting Information Systems*, 7th edition, New York: John Wiley & Sons, Inc., 2001.

2. American Accounting Association, "Report of the AAA Committee on Contemporary Approaches to Teaching Accounting Information Systems," *Journal of Information Systems*, p.139, Spring 1996.
3. Mark Teagan and Liz Young, "The Dynamics of Prototyping," *Computerworld*, pp.53-58, August 8, 1995.

References

1. American Accounting Association, "Report of the AAA Committee on Contemporary Approaches to Teaching Accounting Information Systems," *Journal of Information Systems*, p. 139, Spring 1996.
2. Casher, Jonathan D. and Metzger, Robert H., "Leverage Your Vendor Relationships and Enhance Your Bottom Line," Management Accounting, pp. 51-54, March 1998.
3. Duncan, James R. and Nixon, Mary, "From Watchdog to Consultant," *Strategic Finance*, pp. 42-47, April 1999.
4. Hendricks, James A. and Rose, Kenneth J., "Accountants on the Line," *Management Accounting*, pp.45-48, June 1998.
5. Johnson, Scott D., "The ABCs of the Electric Utility Industry," Management Accounting, pp. 25-32, November 1998.
6. McCarthy, Chris, "Using Technology as a Competitive Tool, *Management Accounting*, pp.28-33, January 1998.
7. Morgan, Mark, "Staying Competitive in Applications Software," Management Accounting, pp.20-26, May 1998.
8. Moscove, Stephen A., Simkin, Mark G., and Bagranoff, Nancy A., *Core Concepts of Accounting Information Systems*, 7th edition, New York: John Wiley & Sons, Inc., 2001.
9. Teagan, Mark and Young, Liz, "The Dynamics of Prototyping," *Computerworld*, pp. 53-58, August 8, 1995.